Panduan Pengguna

# AWS Cloud9

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS Cloud9: Panduan Pengguna

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

# Table of Contents

..... xxi
Apa itu AWS Cloud9? ..... 1
Bagaimana cara AWS Cloud9 kerjanya? ..... 1
AWS Cloud9 lingkungan ..... 1
Lingkungan dan sumber daya komputasi ..... 2
Apa yang bisa saya lakukan dengan AWS Cloud9? ..... 2
Bagaimana saya memulainya? ..... 3
Topik Tambahan ..... 3
Apa yang bisa saya lakukan dengan itu? ..... 3
Informasi tambahan ..... 5
Video terkait ..... 6
Topik terkait di AWS Situs ..... 6
Harga ..... 6
Saya memiliki pertanyaan tambahan atau memerlukan bantuan ..... 7
Menyiapkan AWS Cloud9 ..... 8
Persiapan pengguna individu ..... 8
Prasyarat ..... 9
Cara lain untuk mengautentikasi ..... 11
Langkah selanjutnya ..... 12
Persiapan tim ..... 13
Prasyarat ..... 9
Langkah 1: Buat grup dan pengguna IAM, dan tambahkan pengguna ke grup ..... 16
Langkah 2: Tambahkan izin AWS Cloud9 akses ke grup ..... 22
Langkah 3: Masuk ke AWS Cloud9 konsol ..... 26
Langkah selanjutnya ..... 27
Pengaturan Korporasi ..... 28
Langkah 1: Buat akun pengelolaan untuk organisasi ..... 32
Langkah 2: Buat organisasi untuk akun pengelolaan ..... 32
Langkah 3: Tambahkan akun anggota ke organisasi ..... 33
Langkah 4: Aktifkan Pusat Identitas IAM di seluruh organisasi ..... 34
Langkah 5. Menyiapkan grup dan pengguna dalam organisasi ..... 34
Langkah 6. Memungkinkan grup dan pengguna dalam organisasi untuk digunakan AWS Cloud9 ..... 36
Langkah 7: Mulai gunakan AWS Cloud9 ..... 38

Langkah selanjutnya ..... 39
Pilihan persediaan tambahan (tim dan perusahaan) ..... 39
Langkah 1: Buat kebijakan terkelola pelanggan ..... 40
Langkah 2: Menambahkan kebijakan terkelola pelanggan ke grup ..... 42
Langkah selanjutnya ..... 43
Contoh kebijakan terkelola pelanggan untuk tim, menggunakan AWS Cloud9 ..... 43
Memulai: tutorial basic ..... 50
Prasyarat ..... 50
Langkah 1: Buat lingkungan ..... 50
..... 58
Langkah 2: Tur Basic ..... 58
Siapkan lingkungan Anda ..... 58
Tulis kode ..... 59
Jalankan kode Anda ..... 59
Debug kode Anda ..... 61
..... 63
Langkah 3: Bersihkan ..... 63
..... 65
Informasi terkait ..... 65
Bekerja dengan lingkungan ..... 68
Pembuatan lingkungan ..... 68
Menciptakan EC2 Lingkungan ..... 69
Membuat Lingkungan SSH ..... 87
Mengakses EC2 instans no-ingress dengan Systems Manager ..... 92
Manfaat menggunakan Systems Manager untuk EC2 lingkungan ..... 93
Mengelola izin Systems Manager ..... 95
Memberi pengguna akses ke instans yang dikelola oleh Pengelola Sesi ..... 97
Menggunakan AWS CloudFormation untuk membuat lingkungan tanpa masuknya EC2 ..... 100
Mengonfigurasi VPC endpoint untuk Amazon S3 guna mengunduh dependensi ..... 103
Mengonfigurasi VPC endpoint untuk konektivitas privat ..... 106
Membuka lingkungan ..... 107
Panggilan Layanan AWS dari Lingkungan ..... 109
Membuat dan menggunakan profil instans untuk mengelola kredensial sementara ..... 111
Membuat dan menyimpan kredensial akses permanen di Lingkungan ..... 117
Mengubah Pengaturan Lingkungan ..... 122
Mengubah preferensi lingkungan ..... 122

Mengubah pengaturan lingkungan dengan konsol ..... 123
Mengubah pengaturan lingkungan dengan kode ..... 124
Bekerja dengan Lingkungan Bersama ..... 125
Kasus Penggunaan Lingkungan Bersama ..... 126
Tentang peran akses anggota lingkungan ..... 126
Undang pengguna di akun yang sama dengan Lingkungan ..... 129
Minta AWS Cloud9 administrator di akun yang sama dengan Lingkungan mengundang dirinya sendiri atau orang lain ..... 131
Membuka Lingkungan bersama ..... 133
Lihat daftar anggota lingkungan ..... 134
Membuka file aktif anggota lingkungan ..... 135
Membuka file terbuka anggota lingkungan ..... 135
Pergi ke kursor aktif anggota lingkungan ..... 136
Mengelola obrolan di Lingkungan bersama ..... 136
Mengubah peran akses anggota lingkungan ..... 137
Hapus pengguna Anda dari Lingkungan bersama ..... 139
Hapus anggota lingkungan lain ..... 140
Praktik terbaik berbagi lingkungan ..... 141
Memindahkan lingkungan Volume Amazon EBS ..... 142
Memindahkan sebuah lingkungan ..... 143
Memindahkan AWS Cloud9 EC2 lingkungan ke Amazon Machine Image (AMI) yang berbeda ..... 146
Ubah ukuran volume Amazon EBS ..... 150
Enkripsi volume Amazon EBS ..... 153
Menghapus Lingkungan ..... 156
Menghapus Lingkungan dengan konsol ..... 157
Menghapus Lingkungan dengan kode ..... 159
Bekerja dengan IDE ..... 161
Tur IDE ..... 162
Prasyarat ..... 163
Langkah 1: Bilah menu ..... 163
Langkah 2: Dasbor ..... 165
Langkah 3: Jendela lingkungan ..... 166
Langkah 4: Editor, tab, dan panel ..... 167
Langkah 5: Konsol ..... 168
Langkah 6: Buka bagian file ..... 169

Langkah 7: Selokan ..... 170
Langkah 8: Bilah status ..... 170
Langkah 9: jendela Garis Besar ..... 172
Langkah 10: jendela Pergi ..... 173
Langkah 11: tab Segera ..... 175
Langkah 12: Daftar proses ..... 176
Langkah 13: Preferensi ..... 177
Langkah 14: Terminal ..... 178
Langkah 15: Jendela Debugger ..... 179
Pikiran akhir ..... 185
Dukungan bahasa ..... 186
Versi bahasa pemrograman yang didukung di AWS Cloud9 IDE ..... 188
Dukungan bahasa yang ditingkatkan ..... 189
Dukungan Java yang disempurnakan ..... 189
TypeScript Dukungan yang ditingkatkan ..... 197
Referensi perintah menu ..... 202
AWS Cloud9 menu ..... 202
Menu berkas ..... 203
Edit menu ..... 205
Temukan menu ..... 208
Lihat menu ..... 209
Buka menu ..... 211
Jalankan menu ..... 212
Menu alat ..... 213
Menu jendela ..... 214
Menu Support ..... 216
Menu pratinjau ..... 217
Perintah bilah menu lainnya ..... 217
Menemukan dan Mengganti Teks ..... 218
Menemukan Teks di File Tunggal ..... 218
Mengganti Teks di File Tunggal ..... 218
Menemukan Teks di Beberapa File ..... 219
Mengganti Teks di Beberapa File ..... 220
Opsi Menemukan dan Mengganti ..... 221
Mempratinjau file ..... 222
Membuka file untuk pratinjau ..... 223

Memuat ulang pratinjau file ..... 224
Mengubah jenis pratinjau file ..... 224
Membuka pratinjau file di tab peramban web terpisah ..... 225
Beralih ke pratinjau file berbeda ..... 225
Pratinjau aplikasi yang berjalan ..... 225
Menjalankan aplikasi ..... 225
Mempratinjau aplikasi yang sedang berjalan ..... 227
Memuat ulang pratinjau aplikasi ..... 229
Mengubah jenis pratinjau aplikasi ..... 229
Buka pratinjau aplikasi ..... 229
Beralih ke URL pratinjau yang berbeda ..... 230
Bagikan aplikasi yang sedang berjalan ..... 230
Bekerja dengan Revisi File ..... 236
Bekerja dengan File Gambar ..... 237
Melihat atau Mengedit Gambar ..... 238
Mengubah Ukuran Gambar ..... 238
Memangkas Gambar ..... 238
Memutar Gambar ..... 239
Membalik Gambar ..... 239
Memperbesar Gambar ..... 239
Menghaluskan Gambar ..... 240
Bekerja dengan Builder, Runner, dan Debugger ..... 240
Dukungan Bangun, Jalankan, dan Debug Bawaan ..... 240
Bangun File Proyek Anda ..... 241
Jalankan Kode Anda ..... 241
Debug Kode Anda ..... 242
Ubah Runner Bawaan ..... 243
Buat Konfigurasi Penjalanan ..... 243
Buat Builder atau Runner ..... 244
Tentukan Builder atau Runner ..... 245
Bekerja dengan Variabel Lingkungan Kustom ..... 249
Atur Variabel Lingkungan Kustom Tingkat Perintah ..... 249
Atur Variabel Lingkungan Pengguna Kustom di ~/.bash_profile ..... 250
Atur Variabel Lingkungan Kustom Lokal ..... 250
Atur Variabel Lingkungan Pengguna Kustom di ~/.bashrc ..... 250
Atur Variabel Lingkungan Kustom dalam Daftar ENV ..... 251

Bekerja dengan pengaturan proyek ..... 251
Melihat atau mengubah pengaturan proyek ..... 252
Menerapkan pengaturan proyek saat ini untuk lingkungan ke lingkungan lain ..... 252
Sesuaikan setelan proyek Anda ..... 253
Menghentikan EC2 instans lingkungan Anda secara manual ..... 261
Bekerja dengan pengaturan pengguna ..... 261
Melihat atau mengubah setelan pengguna ..... 262
Bagikan pengaturan pengguna Anda dengan pengguna lain ..... 262
Sesuaikan setelan pengguna Anda ..... 263
Bekerja dengan Pengaturan AWS Proyek dan Pengguna ..... 273
Pengaturan tingkat proyek ..... 273
Pengaturan tingkat pengguna ..... 274
Bekerja dengan Keybinding ..... 274
Melihat atau mengubah Keybindings Anda ..... 274
Bagikan Keybindings Anda dengan pengguna lain ..... 275
Ubah mode Keyboard Anda ..... 275
Ubah Keybindings sistem operasi Anda ..... 276
Ubah Keybindings tertentu ..... 276
Hapus semua Keybindings kustom Anda ..... 277
Cara Menggunakan Tema ..... 278
Melihat atau mengubah tema Anda ..... 278
Pengaturan tema secara keseluruhan yang dapat Anda ubah ..... 278
Penggantian tema ..... 279
Mengelola tulisan inisialisasi ..... 279
Membuka tulisan inisialisasi Anda ..... 280
Referensi Keybinding Default MacOS ..... 280
Umum ..... 281
Tab ..... 284
Panel ..... 286
Editor Kode ..... 287
Emmet ..... 295
Terminal ..... 296
Jalankan dan Debug ..... 297
Referensi Keybinding Vim MacOS ..... 297
Umum ..... 298
Tab ..... 302

Panel ..... 304
Editor Kode ..... 305
Emmet ..... 313
Terminal ..... 314
Jalankan dan Debug ..... 314
Referensi Keybinding Emacs MacOS ..... 315
Umum ..... 316
Tab ..... 319
Panel ..... 321
Editor Kode ..... 322
Emmet ..... 330
Terminal ..... 331
Jalankan dan Debug ..... 331
Referensi Keybinding Sublime MacOS ..... 332
Umum ..... 333
Tab ..... 338
Panel ..... 340
Editor Kode ..... 341
Emmet ..... 349
Terminal ..... 350
Jalankan dan Debug ..... 350
Referensi Keybinding Default Windows/Linux ..... 351
Umum ..... 352
Tab ..... 355
Panel ..... 358
Editor Kode ..... 358
Emmet ..... 367
Terminal ..... 367
Jalankan dan Debug ..... 368
Referensi Keybinding Vim Windows/Linux ..... 369
Umum ..... 369
Tab ..... 373
Panel ..... 375
Editor Kode ..... 376
Emmet ..... 384
Terminal ..... 384

Jalankan dan Debug ..... 385
Referensi Keybinding Emacs Windows/Linux ..... 386
Umum ..... 386
Tab ..... 390
Panel ..... 392
Editor Kode ..... 393
Emmet ..... 401
Terminal ..... 402
Jalankan dan Debug ..... 402
Referensi Keybinding Sublime Windows/Linux ..... 403
Umum ..... 404
Tab ..... 408
Panel ..... 410
Editor Kode ..... 411
Emmet ..... 419
Terminal ..... 420
Jalankan dan Debug ..... 420
Referensi perintah ..... 421
Bekerja dengan AWS layanan lain ..... 423
Bekerja dengan instans Amazon Lightsail ..... 423
Langkah 1: Membuat instans Lightsail berbasis Linux ..... 424
Langkah 2: Siapkan instance untuk menggunakannya AWS Cloud9 ..... 427
Langkah 3: Membuat dan menghubungkan ke Lingkungan Pengembangan AWS Cloud9 SSH ..... 429
Langkah 4: Gunakan AWS Cloud9 IDE untuk mengubah kode pada instance ..... 432
Bekerja dengan AWS CodePipeline ..... 433
Langkah 1: Buat atau identifikasi repositori kode sumber Anda ..... 434
Langkah 2: Buat Lingkungan AWS Cloud9 Pengembangan, hubungkan ke repositori kode, dan unggah kode Anda ..... 434
Langkah 3: Bersiaplah untuk bekerja dengan AWS CodePipeline ..... 436
Langkah 4: Buat alur di AWS CodePipeline ..... 436
Bekerja dengan CodeCatalyst ..... 436
Pelajari cara menggunakan AWS Cloud9 CodeCatalyst dan mereplikasi AWS Cloud9 lingkungan Anda di CodeCatalyst. ..... 437
Bermigrasi dari AWS Cloud9 ke Amazon CodeCatalyst ..... 438
Menggunakan alat replikasi ..... 452

FAQs tentang proses replikasi ..... 456
Lingkungan Pengembang di CodeCatalyst ..... 459
Bekerja dengan AWS CDK ..... 464
AWS CDK aplikasi ..... 465
Kontrol sumber visual dengan panel Git ..... 468
Mengelola kontrol sumber dengan panel Git ..... 469
Menginisialisasi atau klon repositori Git ..... 472
Staging dan commiting file ..... 474
Melihat versi file yang berbeda ..... 477
Bekerja dengan cabang ..... 477
Bekerja dengan repositori jauh ..... 481
Menyimpan dan mengambil file ..... 483
Referensi: Perintah Git tersedia di panel Git ..... 484
Referensi untuk perintah Git tersedia dari menu panel Git ..... 486
Perintah Git tersedia dari bidang pencarian panel Git ..... 488
Bekerja dengan AWS Toolkit ..... 491
Mengapa menggunakan AWS Toolkit? ..... 491
Mengaktifkan Toolkit AWS ..... 493
Mengelola kredensi akses untuk Toolkit AWS ..... 494
Menggunakan peran IAM untuk memberikan izin ke aplikasi pada instance EC2 ..... 495
Mengidentifikasi AWS komponen Toolkit ..... 496
Menonaktifkan Toolkit AWS ..... 496
AWS Topik Toolkit ..... 497
Menjelajahi dan mengkonfigurasi ..... 497
Menggunakan AWS Explorer untuk bekerja dengan layanan dan sumber daya di beberapa Wilayah ..... 498
Mengakses dan menggunakan menu AWS Toolkit ..... 499
Memodifikasi pengaturan AWS Toolkit menggunakan panel Konfigurasi AWS ..... 501
API Gateway ..... 504
Memohon REST APIs ..... 505
AWS App Runner ..... 506
Prasyarat ..... 506
Harga ..... 510
Membuat layanan App Runner ..... 510
Mengelola layanan App Runner ..... 513
AWS CloudFormation tumpukan ..... 515

Menghapus tumpukan AWS CloudFormation ..... 516
CloudWatch Log Amazon ..... 516
Melihat grup CloudWatch log dan aliran log ..... 517
Bekerja dengan peristiwa CloudWatch log ..... 517
AWS Lambda fungsi ..... 519
Memanggil fungsi Lambda jarak jauh ..... 520
Mengunduh, mengunggah, dan menghapus fungsi Lambda ..... 521
Sumber daya ..... 526
Izin IAM untuk mengakses sumber daya ..... 527
Berinteraksi dengan sumber daya yang ada ..... 527
Amazon S3 ..... 528
Bekerja dengan bucket Amazon S3 ..... 528
Cara menggunakan objek Amazon S3 ..... 531
AWS Aplikasi Tanpa Server ..... 533
Membuat aplikasi nirserver ..... 534
Menjalankan dan men-debug aplikasi nirserver ..... 535
Menyinkronkan aplikasi tanpa server ..... 543
Menghapus aplikasi nirserver ..... 545
Mengaktifkan lensa kode AWS Toolkit ..... 545
Opsi konfigurasi untuk men-debug aplikasi nirserver ..... 545
AWS Step Functions ..... 549
Prasyarat ..... 550
Membuat dan mempublikasikan mesin negara ..... 550
Jalankan mesin status di AWS Toolkit ..... 552
Unduh file definisi mesin negara dan visualisasikan alur kerjanya ..... 553
AWS Systems Manager ..... 554
Asumsi dan prasyarat ..... 555
Izin IAM untuk dokumen Otomasi Systems Manager ..... 555
Membuat dokumen otomatisasi Systems Manager baru ..... 556
Menerbitkan dokumen otomatisasi Systems Manager ..... 556
Mengedit dokumen otomatisasi Systems Manager yang ada ..... 557
Bekerja dengan versi ..... 558
Menghapus dokumen otomatisasi Systems Manager ..... 558
Menjalankan dokumen otomatisasi Systems Manager ..... 559
Pemecahan Masalah ..... 560
Amazon ECR ..... 560

Prasyarat ..... 561
Menggunakan Amazon ECR dengan IDE AWS Cloud9 ..... 561
AWS IoT ..... 571
AWS IoT prasyarat ..... 571
AWS IoT Hal-hal ..... 571
AWS IoT sertifikat ..... 573
AWS IoT kebijakan ..... 576
Amazon ECS ..... 579
Amazon ECS Exec ..... 580
Amazon EventBridge ..... 582
Bekerja dengan Amazon EventBridge Schemas ..... 582
Tutorial untuk AWS Cloud9 ..... 585
AWS CLI dan tutorial aws-shell ..... 585
Prasyarat ..... 586
Langkah 1: Instal AWS CLI, aws-shell, atau keduanya di lingkungan Anda ..... 587
Langkah 2: Mengatur pengelolaan kredensial di lingkungan Anda ..... 588
Langkah 3: Jalankan perintah dasar dengan AWS CLI atau aws-shell di lingkungan Anda ... 589
Langkah 4: Membersihkan ..... 590
AWS CodeCommit tutorial ..... 590
Prasyarat ..... 591
Langkah 1: Mengatur grup IAM Anda dengan izin akses yang diperlukan ..... 591
Langkah 2: Buat repositori di CodeCommit ..... 593
Langkah 3: Hubungkan lingkungan Anda ke repositori jarak jauh ..... 594
Langkah 4: Klon repositori jarak jauh ke lingkungan Anda ..... 596
Langkah 5: Tambahkan file ke repositori ..... 597
Langkah 6: Bersihkan ..... 599
Amazon DynamoDB tutorial ..... 599
Prasyarat ..... 600
Langkah 1: Instal dan konfigurasikan AWS CLI, yang AWS CloudShell, atau keduanya di lingkungan Anda ..... 601
Langkah 2: Buat Tabel ..... 602
Langkah 3: Tambahkan item ke tabel ..... 603
Langkah 4: Tambahkan beberapa item ke tabel ..... 604
Langkah 5: Buat indeks sekunder global ..... 608
Langkah 6: Dapatkan item dari tabel ..... 611
Langkah 7: Bersihkan ..... 616

AWS CDK tutorial ..... 616
Prasyarat ..... 617
Langkah 1: Pasang alat yang diperlukan ..... 617
Langkah 2: Menambahkan kode ..... 621
Langkah 3: Jalankan kode ..... 623
Langkah 4: Membersihkan ..... 626
Tutorial LAMP ..... 626
Prasyarat ..... 627
Langkah 1: Pasang alatnya ..... 627
Langkah 2: Mengatur MySQL ..... 629
Langkah 3: Mengatur situs web ..... 631
Langkah 4: Membersihkan ..... 635
WordPress tutorial ..... 637
Prasyarat ..... 637
Gambaran umum instalasi ..... 638
Langkah 1: Menginstal dan mengkonfigurasi MariaDB Server ..... 638
Langkah 2: Menginstal dan mengkonfigurasi WordPress ..... 639
Langkah 3: Mengkonfigurasi Apache HTTP Server ..... 640
Langkah 4: Mempratinjau konten WordPress web ..... 641
Mengelola kesalahan konten campuran ..... 642
Tutorial Java ..... 642
Prasyarat ..... 643
Langkah 1: Pasang alat yang diperlukan ..... 644
Langkah 2: Menambahkan kode ..... 646
Langkah 3: Bangun dan jalankan kode ..... 646
Langkah 4: Siapkan untuk menggunakan AWS SDK untuk Java ..... 647
Langkah 5: Siapkan manajemen AWS kredensil di lingkungan Anda ..... 653
Langkah 6: Tambahkan kode AWS SDK ..... 654
Langkah 7: Bangun dan jalankan kode AWS SDK ..... 656
Langkah 8: Membersihkan ..... 657
C ++ tutorial ..... 657
Prasyarat ..... 657
Langkah 1: Pasang g++ dan paket dev yang diperlukan ..... 658
Langkah 2: Instal CMake ..... 659
Langkah 3: Dapatkan dan bangun SDK for C++ ..... 659
Langkah 4: Buat file C ++ dan CMake Daftar ..... 661

Langkah 5: Membangun dan menjalankan kode C++ ..... 665
Langkah 6: Bersihkan ..... 666
Tutorial Python ..... 666
Prasyarat ..... 667
Langkah 1: Pasang Python ..... 667
Langkah 2: Menambahkan kode ..... 668
Langkah 3: Jalankan kode ..... 668
Langkah 4: Instal dan konfigurasikan AWS SDK untuk Python (Boto3) ..... 669
Langkah 5: Tambahkan kode AWS SDK ..... 670
Langkah 6: Jalankan kode AWS SDK ..... 672
Langkah 7: Bersihkan ..... 673
Tutorial.NET ..... 673
Prasyarat ..... 673
Langkah 1: Pasang alat yang diperlukan ..... 674
Langkah 2 (Opsional): Pasang ekstensi .NET CLI untuk fungsi Lambda ..... 676
Langkah 3: Buat proyek aplikasi konsol.NET ..... 677
Langkah 4: Tambahkan kode ..... 677
Langkah 5: Bangun dan jalankan kode ..... 678
Langkah 6: Buat dan siapkan proyek aplikasi konsol.NET yang menggunakan AWS SDK untuk .NET ..... 680
Langkah 7: Tambahkan kode AWS SDK ..... 681
Langkah 8: Bangun dan jalankan kode AWS SDK ..... 683
Langkah 9: Membersihkan ..... 684
Tutorial Node.js ..... 684
Prasyarat ..... 685
Langkah 1: Pasang alat yang diperlukan ..... 685
Langkah 2: Menambahkan kode ..... 687
Langkah 3: Jalankan kode ..... 687
Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js ..... 688
Langkah 5: Tambahkan kode AWS SDK ..... 690
Langkah 6: Jalankan kode AWS SDK ..... 693
Langkah 7: Bersihkan ..... 694
PHP tutorial ..... 695
Prasyarat ..... 695
Langkah 1: Pasang alat yang diperlukan ..... 696
Langkah 2: Menambahkan kode ..... 697

Langkah 3: Jalankan kode ... 698
Langkah 4: Instal dan konfigurasikan AWS SDK untuk PHP ... 698
Langkah 5: Tambahkan kode AWS SDK ... 700
Langkah 6: Jalankan kode AWS SDK ... 702
Langkah 7: Bersihkan ... 703
Ruby ... 703
Pergi tutorial ... 703
Prasyarat ... 704
Langkah 1: Pasang alat yang diperlukan ... 704
Langkah 2: Menambahkan kode ... 705
Langkah 3: Jalankan kode ... 706
Langkah 4: Instal dan konfigurasikan AWS SDK untuk Go ... 708
Langkah 5: Tambahkan kode AWS SDK ... 709
Langkah 6: Jalankan kode AWS SDK ... 711
Langkah 7: Bersihkan ... 712
TypeScript tutorial ... 712
Prasyarat ... 713
Langkah 1: Pasang alat yang diperlukan ... 713
Langkah 2: Menambahkan kode ... 715
Langkah 3: Jalankan kode ... 716
Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js ... 717
Langkah 5: Tambahkan kode AWS SDK ... 718
Langkah 6: Jalankan kode AWS SDK ... 720
Langkah 7: Bersihkan ... 721
Tutorial Docker ... 722
Prasyarat ... 722
Langkah 1: Pasang dan jalankan Docker ... 723
Langkah 2: Bangun citra ... 724
Langkah 3: Jalankan kontainer ... 727
Langkah 4: Buat lingkungan ... 729
Langkah 5: Jalankan kode ... 734
Langkah 6: Membersihkan ... 735
Tutorial Terkait ... 736
Topik lanjutan untuk AWS Cloud9 ... 737
EC2 Lingkungan dibandingkan dengan lingkungan SSH ... 737
Pengaturan Amazon VPC ... 739

Persyaratan Amazon VPC untuk AWS Cloud9 ..... 739
Buat VPC plus sumber daya VPC lainnya ..... 754
Buat VPC saja ..... 756
Buat subnet untuk AWS Cloud9 ..... 757
Mengonfigurasi subnet sebagai publik atau subnet privat ..... 759
Persyaratan host lingkungan SSH ..... 761
Waktu dan cara membuat Lingkungan SSH ..... 762
Persyaratan hosting SSH ..... 763
AWS Cloud9 Pemasang ..... 766
Unduh dan Jalankan AWS Cloud9 Installer ..... 766
Memecahkan Masalah Installer AWS Cloud9 ..... 767
Rentang alamat IP SSH inbound ..... 769
Alamat IP tidak ada di `ip-ranges.json` ..... 770
Konten AMI ..... 771
Amazon Linux 2023/Amazon Linux 2 ..... 772
Ubuntu Server ..... 773
Peran terkait layanan ..... 774
Izin peran terkait layanan untuk AWS Cloud9 ..... 775
Membuat peran terkait layanan untuk AWS Cloud9 ..... 779
Mengedit peran terkait layanan untuk AWS Cloud9 ..... 779
Menghapus peran terkait layanan untuk AWS Cloud9 ..... 779
Wilayah yang Didukung untuk AWS Cloud9 peran terkait layanan ..... 779
Pencatatan panggilan API dengan CloudTrail ..... 780
AWS Cloud9 informasi di CloudTrail ..... 780
Memahami entri file AWS Cloud9 log ..... 782
Tanda ..... 798
Menyebarkan pembaruan tanda ke sumber daya yang mendasari ..... 799
Keamanan untuk AWS Cloud9 ..... 801
Perlindungan data ..... 802
Enkripsi data ..... 803
Identity and Access Management ..... 805
Audiens ..... 806
Mengautentikasi dengan identitas ..... 806
Mengelola akses menggunakan kebijakan ..... 810
Bagaimana AWS Cloud9 bekerja dengan IAM ..... 813
Contoh kebijakan berbasis identitas ..... 820

Pemecahan Masalah ..... 823
Bagaimana AWS Cloud9 bekerja dengan sumber daya dan operasi IAM ..... 825
AWS kebijakan terkelola ..... 829
Membuat kebijakan terkelola pelanggan untuk AWS Cloud9 ..... 840
AWS Cloud9 referensi izin ..... 855
AWS kredensi sementara yang dikelola ..... 862
Pencatatan log dan pemantauan ..... 868
Memantau aktivitas dengan CloudTrail ..... 868
Memantau kinerja EC2 lingkungan ..... 869
Validasi kepatuhan ..... 869
Ketahanan ..... 874
Keamanan infrastruktur ..... 875
Pembaruan perangkat lunak dan patch ..... 875
Praktik terbaik keamanan ..... 876
Pemecahan masalah AWS Cloud9 ..... 877
Penginstal ..... 877
AWS Cloud9 Pemasang hang atau gagal ..... 877
AWS Cloud9 installer tidak selesai setelah menampilkan: “Package Cloud9 IDE 1" ..... 878
Gagal menginstal dependensi ..... 878
Kesalahan lingkungan SSH: “Python versi 3 diperlukan untuk menginstal pty.js” ..... 879
AWS Cloud9 Lingkungan ..... 879
Kesalahan pembuatan lingkungan: “Kami tidak dapat membuat EC2 instance...” ..... 880
Kesalahan pembuatan lingkungan: “Tidak diizinkan untuk melakukan sts:AssumeRole” ..... 880
Identitas federasi tidak dapat menciptakan lingkungan ..... 881
Kesalahan konsol: "Pengguna tidak berwenang untuk melakukan tindakan pada sumber daya" ..... 881
Tidak dapat terhubung ke lingkungan ..... 882
Tidak dapat membuka lingkungan ..... 882
Tidak dapat membuka AWS Cloud9 lingkungan: “Lingkungan ini saat ini tidak dapat diakses oleh kolaborator. Harap tunggu sampai penghapusan kredensial sementara terkelola selesai, atau kontak pemilik lingkungan ini." ..... 884
Kesalahan penghapusan lingkungan: "Satu atau lebih lingkungan gagal untuk menghapus" ..... 885
Mengubah waktu tunggu untuk lingkungan di IDE AWS Cloud9 ..... 886
Kesalahan menjalankan aplikasi SAM secara lokal di AWS Toolkit karena AWS Cloud9 lingkungan tidak memiliki cukup ruang disk ..... 886

Tidak dapat memuat IDE menggunakan versi sebelumnya Microsoft Edge penjelajah ..... 887
Tidak dapat membuat struktur sub-folder/ home/ec2-user/environment/home/ec2-user/ environment di AWS Cloud9 IDE File Explorer. ..... 888
Tidak dapat membuat struktur sub-folder /projects/projects dalam File Explorer IDE untuk. AWS Cloud9 CodeCatalyst ..... 888
Tidak dapat berinteraksi dengan jendela terminal AWS Cloud9 karena kesalahan `tmux` sesi ..... 889
Amazon EC2 ..... 890
EC2 Instans Amazon tidak diperbarui secara otomatis ..... 890
AWS CLI atau AWS-shell error: “Token keamanan yang disertakan dalam permintaan tidak valid” di lingkungan EC2 ..... 891
Tidak dapat terhubung ke EC2 lingkungan karena alamat IP VPC digunakan oleh Docker ... 892
Tidak dapat membuat struktur sub-folder/ home/ec2-user/environment/home/ec2-user/ environment di AWS Cloud9 IDE File Explorer. ..... 888
Tidak dapat diluncurkan AWS Cloud9 dari konsol saat konfigurasi AWS License Manager lisensi dikaitkan dengan EC2 instans Amazon ..... 893
Tidak dapat menjalankan beberapa perintah atau skrip di lingkungan EC2 ..... 893
Pelaporan pesan kesalahan “ AWSCloud9SSMInstanceProfil Instans Profil tidak ada di akun” saat membuat EC2 lingkungan menggunakan AWS CloudFormation ..... 894
Pelaporan pesan kesalahan “tidak diizinkan untuk `perform: ssm:StartSession` pada sumber daya” saat membuat EC2 lingkungan menggunakan AWS CloudFormation ..... 894
Pesan kesalahan melaporkan tidak ada otorisasi “untuk melakukan: `iam:GetInstanceProfile` pada sumber daya: profil instance`AWSCloud9SSMInstanceProfile`" saat membuat EC2 lingkungan menggunakan AWS CLI ..... 895
Kegagalan untuk membuat lingkungan ketika default enkripsi diterapkan ke Amazon EBS volume ..... 895
Kesalahan VPC untuk akun EC2 -Classic: “Tidak dapat mengakses lingkungan Anda” ..... 896
AWS Layanan lainnya ..... 897
Tidak dapat membuat struktur sub-folder /projects/projects dalam File Explorer IDE untuk. AWS Cloud9 CodeCatalyst ..... 888
Tidak dapat menampilkan aplikasi yang sedang berjalan di luar IDE ..... 897
Kesalahan saat menjalankan AWS Toolkit: “Lingkungan Anda kehabisan inode, harap tingkatkan batas 'fs.inotify.max_user_watches'.” ..... 900
Kesalahan menjalankan fungsi lokal Lambda: Tidak dapat menginstal SAM Local ..... 900

AWS Control Tower kesalahan saat mencoba membuat EC2 lingkungan Amazon menggunakan AWS Cloud9: “Pembuatan lingkungan gagal dengan kesalahan: Kait berikut gagal: [: :GuardControlTower: :Hook].” ..... 901
Kegagalan untuk membuat lingkungan ketika default enkripsi diterapkan ke Amazon EBS volume ..... 895
Tidak dapat diluncurkan AWS Cloud9 dari konsol saat konfigurasi AWS License Manager lisensi dikaitkan dengan EC2 instans Amazon ..... 893
Pratinjau aplikasi ..... 902
Setelah memuat ulang lingkungan, Anda harus me-refresh pratinjau aplikasi ..... 902
Pemberitahuan pratinjau aplikasi atau pratinjau file: "Cookie pihak ke tiga dinonaktifkan" .... 903
Tab pratinjau aplikasi menampilkan kesalahan atau kosong ..... 907
Tidak dapat melihat pratinjau konten web di IDE karena koneksi ke situs tidak aman ..... 908
Pratinjau file mengembalikan kesalahan 499 ..... 909
Kinerja ..... 909
AWS Cloud9 Pembekuan IDE untuk waktu yang signifikan ..... 909
Peringatan konsol: "Beralih ke mesin penyelesaian kode minimal..." ..... 910
Peringatan IDE: "Lingkungan ini kehabisan memori" atau "Lingkungan ini memiliki beban CPU yang tinggi" ..... 910
Tidak dapat mengunggah file di AWS Cloud9 IDE ..... 911
Kecepatan unduh lambat di AWS Cloud9 IDE ..... 912
Tidak dapat melihat pratinjau konten web di IDE karena koneksi ke situs tidak aman ..... 908
Aplikasi dan layanan pihak ketiga ..... 912
Tidak dapat berinteraksi dengan jendela terminal AWS Cloud9 karena kesalahan `tmux` sesi ..... 889
Tidak dapat memuat IDE menggunakan versi sebelumnya Microsoft Edge penjelajah ..... 887
Kesalahan `gdb` saat men-debug C++ memproyeksikan ..... 915
Masalah dengan PHP runner di AWS Cloud9 ..... 916
Kesalahan GLIBC terkait dengan Node.js ..... 916
Peramban yang didukung ..... 917
Kuota layanan ..... 919
AWS Cloud9 kuota ..... 919
AWS Cloud9 Kuota Unduh IDE ..... 920
Kuota AWS Layanan Terkait ..... 920
Riwayat dokumen ..... 922

AWS Cloud9 tidak lagi tersedia untuk pelanggan baru. Pelanggan yang sudah ada AWS Cloud9 dapat terus menggunakan layanan seperti biasa. Pelajari selengkapnya

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.

# Apa itu AWS Cloud9?

AWS Cloud9 adalah lingkungan pengembangan terintegrasi, atau IDE.

AWS Cloud9 IDE menawarkan pengalaman pengeditan kode yang kaya dengan dukungan untuk beberapa bahasa pemrograman dan debugger runtime, dan terminal bawaan. Ini berisi koleksi alat yang Anda gunakan untuk kode, membangun, menjalankan, menguji, debug perangkat lunak, dan membantu Anda melepaskan perangkat lunak di cloud.

Anda mengakses AWS Cloud9 IDE melalui browser web. Anda dapat mengonfigurasi IDE untuk preferensi Anda. Anda dapat beralih tema warna, mengikat kunci pintas, mengaktifkan pewarnaan sintaks bahasa khusus pemrograman dan pemformatan kode, dan banyak lagi.

(Punya! Saya siap untuk mencoba AWS Cloud9. Bagaimana saya memulainya?)

# Bagaimana cara AWS Cloud9 kerjanya?

Diagram berikut menunjukkan gambaran tingkat tinggi tentang cara AWS Cloud9 kerja.

Dari diagram (mulai dari bawah), Anda menggunakan AWS Cloud9 IDE, berjalan di peramban web di komputer lokal Anda, untuk berinteraksi dengan lingkungan AWS Cloud9 . Sumber daya komputasi (misalnya,  EC2 instans Amazon atau server Anda sendiri) terhubung ke lingkungan itu. Akhirnya, pekerjaan Anda disimpan dalam repositori AWS CodeCommit  atau jenis repositori jarak jauh lainnya.

## AWS Cloud9 lingkungan

lingkungan AWS Cloud9  adalah tempat di mana Anda menyimpan file proyek pengembangan Anda dan tempat Anda menjalankan alat untuk mengembangkan aplikasi Anda.

Menggunakan AWS Cloud9 IDE, Anda dapat:

- Menyimpan file proyek Anda secara lokal pada instans atau server.
- Kloning repositori kode jarak jauh—seperti repo di —ke lingkungan Anda. AWS CodeCommit
- Bekerja dengan kombinasi file lokal dan klon di lingkungan.

Anda dapat membuat dan beralih di antara beberapa lingkungan, dengan setiap lingkungan diatur untuk proyek pengembangan tertentu. Dengan menyimpan lingkungan di cloud, proyek Anda tidak perlu lagi dikaitkan dengan satu komputer atau pengaturan server. Hal ini mengaktifkan Anda untuk melakukan hal-hal seperti dengan mudah beralih antara komputer dan pengembang onboard lebih cepat ke tim Anda.

## Lingkungan dan sumber daya komputasi

Di balik layar, ada beberapa cara untuk connect lingkungan Anda ke sumber daya komputasi:

- Anda dapat menginstruksikan AWS Cloud9 untuk membuat EC2 instance Amazon, dan kemudian menghubungkan lingkungan ke EC2 instance yang baru dibuat. Jenis pengaturan ini disebut EC2 lingkungan.
- Anda dapat menginstruksikan AWS Cloud9 untuk menghubungkan lingkungan ke instance komputasi cloud yang ada atau ke server Anda sendiri. Jenis pengaturan ini disebut Lingkungan SSH.

EC2 lingkungan dan lingkungan SSH memiliki beberapa kesamaan dan beberapa perbedaan. Jika Anda baru AWS Cloud9, kami sarankan Anda menggunakan EC2 lingkungan karena AWS Cloud9 mengurus banyak konfigurasi untuk Anda. Saat Anda mempelajari lebih lanjut tentang AWS Cloud9, dan ingin memahami persamaan dan perbedaan ini dengan lebih baik, lihatEC2 lingkungan dibandingkan dengan lingkungan SSH di AWS Cloud9.

Untuk informasi selengkapnya tentang cara AWS Cloud9 kerja, lihat video dan halaman web terkait ini.

# Apa yang bisa saya lakukan dengan AWS Cloud9?

Dengan AWS Cloud9, Anda dapat membuat kode, membangun, menjalankan, menguji, men-debug, dan merilis perangkat lunak dalam banyak skenario dan variasi menarik. Ini termasuk (namun tidak terbatas pada):

- Bekerja dengan kode dalam beberapa bahasa pemrograman dan AWS Cloud Development Kit (AWS CDK).
- Bekerja dengan kode dalam kontainer Docker berjalan.
- Menggunakan repositori kode online.
- Bekerja sama dengan orang lain secara waktu nyata.

- Berinteraksi dengan berbagai teknologi basis data dan situs web.
- Penargetan AWS Lambda, Amazon API Gateway, dan Aplikasi AWS Tanpa Server.
- Mengambil keuntungan dari AWS produk lain seperti Amazon Lightsail AWS CodeStar,, dan. AWS CodePipeline

Untuk daftar yang lebih detail, lihat Apa yang bisa saya lakukan dengan AWS Cloud9?

## Bagaimana saya memulainya?

Untuk mulai menggunakan AWS Cloud9, ikuti langkah-langkah diMenyiapkan AWS Cloud9, dan kemudian pergi melalui tutorial dasar.

## Topik Tambahan

- Apa yang bisa saya lakukan dengan AWS Cloud9?
- Informasi tambahan tentang AWS Cloud9

## Apa yang bisa saya lakukan dengan AWS Cloud9?

Jelajahi sumber daya berikut untuk mempelajari penggunaan AWS Cloud9 untuk beberapa skenario umum.

Skenario kunci

| Skenario | Sumber Daya |
| --- | --- |
| Buat, jalankan, dan debug kode dalam AWS Lambda fungsi dan aplikasi tanpa server menggunakan Toolkit. AWS | Bekerja dengan AWS Lambda fungsi menggunakan AWS Toolkit |
| Bekerja dengan instans Amazon Lightsail yang telah dikonfigurasi sebelumnya dengan aplikasi dan kerangka kerja populer WordPress seperti, LAMP (Linux, Apache, MySQL, dan PHP), Node.js, Nginx, Drupal, dan Joomla, dan | Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9 |

| Skenario | Sumber Daya |
| --- | --- |
| distribusi Linux seperti Amazon Linux, Ubuntu, Debian, FreeBSD, dan openSUSE. | |
| Bekerja dengan kode dalam solusi pengiriman berkelanjutan di AWS CodePipeline. | Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE |
| Mengotomatiskan AWS layanan dengan menggunakan AWS CLI dan. AWS CloudShell | AWS CLI dan tutorial aws-shell untuk AWS Cloud9 |
| Bekerja dengan repositori kode sumber di. AWS CodeCommit | AWS CodeCommit tutorial untuk AWS Cloud9 |
| Bekerja dengan repositori kode sumber dalam GitHub menggunakan antarmuka panel Git. | Kontrol sumber visual dengan panel Git |
| Bekerja dengan basis data NoSQL di Amazon DynamoDB. | Amazon DynamoDB tutorial untuk AWS Cloud9 |
| Bekerja dengan tumpukan LAMP (Linux, Apache HTTP Server, MySQL, dan PHP). | Tutorial LAMP untuk AWS Cloud9 |
| Bekerja dengan WordPress situs web. | WordPress tutorial untuk AWS Cloud9 |
| Bekerja dengan kode untuk Java dan AWS SDK untuk Java. | Tutorial Java untuk AWS Cloud9 |
| Bekerja dengan kode untuk C++ dan AWS SDK untuk C++. | Tutorial C++ untuk AWS Cloud9 |
| Bekerja dengan kode untuk Python dan AWS SDK for Python (Boto). | Tutorial Python untuk AWS Cloud9 |
| Bekerja dengan kode untuk .NET Core dan AWS SDK untuk .NET. | Tutorial.NET untuk AWS Cloud9 |
| Bekerja dengan kode untuk Node.js dan AWS SDK untuk JavaScript. | Tutorial Node.js untuk AWS Cloud9 |

| Skenario | Sumber Daya |
| --- | --- |
| Bekerja dengan kode untuk PHP dan AWS SDK untuk PHP. | PHP tutorial untuk AWS Cloud9 |
| Bekerja dengan kode untuk Ruby dan AWS SDK untuk Ruby. | AWS SDK for Ruby di AWS Cloud9 |
| Bekerja dengan kode untuk Go dan AWS SDK untuk Go. | Pergi tutorial untuk AWS Cloud9 |
| Bekerja dengan kode untuk TypeScript dan AWS SDK untuk JavaScript. | TypeScript tutorial untuk AWS Cloud9 |
| Bekerja dengan kode untuk AWS Cloud Development Kit (AWS CDK). | AWS CDK tutorial untuk AWS Cloud9 |
| Bekerja dengan kode dalam kontainer Docker berjalan. | Tutorial Docker untuk AWS Cloud9 |
| Undang orang lain untuk menggunakan lingkungan baru bersama Anda, secara langsung dan dengan support obrolan teks. | Bekerja dengan lingkungan bersama di AWS Cloud9 |
| Bekerja dengan kode untuk aplikasi robotika cerdas di AWS RoboMaker. | Berkembang dengan AWS Cloud9 di Panduan AWS RoboMaker Pengembang |

# Informasi tambahan tentang AWS Cloud9

Topik ini memberikan informasi lebih lanjut untuk membantu Anda mempelajarinya AWS Cloud9.

Topik


- Video terkait
- Topik terkait di AWS Situs
- Harga
- Saya memiliki pertanyaan tambahan atau memerlukan bantuan

## Video terkait

- AWS re:invent 2017 - Memperkenalkan AWS Cloud9: Werner Vogels Keynote (9 menit, situs web) YouTube
- AWS re:invent Launchpad 2017 - AWS Cloud9, (15 menit, situs web) YouTube
- Memperkenalkan AWS Cloud9 - AWS Online Tech Talks (33 menit, YouTube situs web)
- AWS Sydney Summit 2018: AWS Cloud9 dan AWS CodeStar (25 menit, YouTube situs web)

## Topik terkait di AWS Situs

- Memperkenalkan AWS Cloud9
- AWS Cloud9 — Lingkungan Pengembang Cloud
- AWS Cloud9 Ikhtisar
- AWS Cloud9 Fitur
- AWS Cloud9 FAQs

## Harga

Tidak ada biaya tambahan untuk AWS Cloud9. Jika Anda menggunakan EC2 instans Amazon untuk lingkungan AWS Cloud9 pengembangan, Anda hanya membayar untuk sumber daya komputasi dan penyimpanan (misalnya, EC2 instans Amazon, volume Amazon EBS) yang digunakan untuk menjalankan dan menyimpan kode Anda. Anda juga dapat meng-connect lingkungan Anda ke server Linux yang ada (misalnya, server on-premise) melalui SSH tanpa biaya tambahan.

Anda hanya membayar untuk apa yang Anda gunakan, saat Anda menggunakannya; tidak ada biaya minimum dan tidak ada komitment di muka. Anda dikenakan AWS tarif normal untuk AWS sumber daya apa pun (misalnya, AWS Lambda fungsi) yang Anda buat atau gunakan di lingkungan Anda.

AWS Pelanggan baru yang memenuhi syarat untuk Tingkat AWS Gratis dapat menggunakannya secara AWS Cloud9 gratis. Jika lingkungan Anda menggunakan sumber daya di luar Tingkat AWS Gratis, Anda akan dikenakan AWS tarif normal untuk sumber daya tersebut.

Untuk informasi selengkapnya, lihat hal berikut.

- AWS Cloud9 harga: Lihat AWS Cloud9 Harga.

- AWS harga layanan: Lihat EC2 Harga Amazon, Harga Amazon EBS, AWS Lambda Harga, dan AWS Harga.
- Tingkat AWS Gratis: Lihat Menggunakan Tingkat AWS Gratis dan Melacak Penggunaan Tingkat Gratis Anda di Panduan AWS Manajemen Penagihan dan Biaya Pengguna.
- Harga pendidikan: Lihat program AWS Mendidik.

## Saya memiliki pertanyaan tambahan atau memerlukan bantuan

Untuk mengajukan pertanyaan atau mencari bantuan dari AWS Cloud9 komunitas, lihat Forum AWS Cloud9 Diskusi. (Saat Anda memasuki forum ini, AWS mungkin mengharuskan Anda untuk masuk.)

Lihat juga pertanyaan yang sering diajukan (FAQs), atau hubungi kami secara langsung.

# Menyiapkan AWS Cloud9

Untuk mulai menggunakan AWS Cloud9, ikuti salah satu dari rangkaian prosedur ini, tergantung pada bagaimana Anda berencana untuk menggunakannya AWS Cloud9.

| Pola penggunaan | Ikuti prosedur ini |
| --- | --- |
| Saya satu-satunya orang yang menggunakan AWS akun saya, dan saya bukan siswa. | Pengaturan Pengguna Individu |
| Saya termasuk dalam tim yang memiliki banyak pengguna dalam satu AWS akun. | Pengaturan Tim |
| Saya milik perusahaan yang memiliki satu atau lebih AWS akun dalam satu organisasi. | Pengaturan Perusahaan |

Untuk informasi umum tentang AWS Cloud9, lihat Apa itu AWS Cloud9? .

Topik

- Pengaturan pengguna individu untuk AWS Cloud9
- Menyiapkan Tim untuk AWS Cloud9
- Pengaturan perusahaan untuk AWS Cloud9
- Opsi pengaturan tambahan untuk AWS Cloud9

# Pengaturan pengguna individu untuk AWS Cloud9

Topik ini menjelaskan cara mengatur dan menggunakan AWS Cloud9 sebagai satu-satunya pengguna di Akun AWS saat Anda bukan pelajar. Anda dapat mengatur AWS Cloud9 pola penggunaan lainnya. Untuk informasi selengkapnya, lihat Menyiapkan AWS Cloud9.

Untuk digunakan AWS Cloud9 sebagai satu-satunya pengguna di Anda Akun AWS, daftar Akun AWS jika Anda belum memilikinya. Selanjutnya, masuk ke AWS Cloud9 konsol.

Topik

- Prasyarat

- Cara lain untuk mengautentikasi
- Langkah selanjutnya

# Prasyarat

## Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

   Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

   Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS mengirimi Anda email konfirmasi setelah proses pendaftaran selesai. Kapan saja, Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan masuk https://aws.amazon.comke/ dan memilih Akun Saya.

## Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke AWS Management Consolesebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

   Untuk bantuan masuk dengan menggunakan pengguna root, lihat Masuk sebagai pengguna root di AWS Sign-In Panduan Pengguna.

2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuk, lihat Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root (konsol) Anda di Panduan Pengguna IAM.

Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center .

2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

   Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

  Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal AWS akses di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

   Untuk petunjuknya, lihat Membuat set izin di Panduan AWS IAM Identity Center Pengguna.

2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

   Untuk petunjuk, lihat Menambahkan grup di Panduan AWS IAM Identity Center Pengguna.

# Cara lain untuk mengautentikasi

> **Warning**
>
> Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti AWS IAM Identity Center.

## Kelola akses di seluruh Akun AWS

Sebagai praktik keamanan terbaik, kami sarankan menggunakan AWS Organizations dengan IAM Identity Center untuk mengelola akses di semua Anda Akun AWS. Untuk informasi selengkapnya tentang administrator, lihat Praktik terbaik keamanan di IAM di dalam Panduan Pengguna IAM.

Anda dapat membuat pengguna di Pusat Identitas IAM, menggunakan Microsoft Active Directory, menggunakan penyedia identitas SAMP 2.0 (iDP), atau menggabungkan IDP Anda secara individual. Akun AWS Dengan menggunakan salah satu pendekatan ini, Anda dapat memberikan pengalaman masuk tunggal untuk pengguna Anda. Anda juga dapat menerapkan otentikasi multi-faktor (MFA) dan menggunakan kredensi sementara untuk akses. Akun AWS Ini berbeda dari pengguna IAM, yang merupakan kredensi jangka panjang yang dapat dibagikan dan yang dapat meningkatkan risiko keamanan terhadap sumber daya Anda AWS .

## Buat pengguna IAM hanya untuk lingkungan kotak pasir

Jika Anda baru AWS, Anda dapat membuat pengguna IAM uji dan kemudian menggunakannya untuk menjalankan tutorial dan menjelajahi apa yang AWS ditawarkan. Tidak apa-apa untuk menggunakan jenis kredensi ini saat Anda belajar, tetapi kami sarankan Anda menghindari menggunakannya di luar lingkungan kotak pasir.

Untuk kasus penggunaan berikut, mungkin masuk akal untuk memulai dengan pengguna IAM di AWS:

- Memulai AWS SDK atau alat Anda dan menjelajah Layanan AWS di lingkungan kotak pasir.
- Menjalankan skrip terjadwal, pekerjaan, dan proses otomatis lainnya yang tidak mendukung proses masuk yang dihadiri manusia sebagai bagian dari pembelajaran Anda.

Jika Anda menggunakan pengguna IAM di luar kasus penggunaan ini, maka transisi ke IAM Identity Center atau federasikan penyedia identitas Anda ke Akun AWS sesegera mungkin. Untuk informasi lebih lanjut, lihat Federasi identitas di AWS.

### Kunci akses pengguna IAM yang aman

Anda harus memutar kunci akses pengguna IAM secara teratur. Ikuti panduan dalam Memutar kunci akses di Panduan Pengguna IAM. Jika Anda yakin bahwa Anda telah secara tidak sengaja membagikan kunci akses pengguna IAM Anda, kemudian putar kunci akses Anda.

Kunci akses pengguna IAM harus disimpan dalam AWS `credentials` file bersama di mesin lokal. Jangan menyimpan kunci akses pengguna IAM dalam kode Anda. Jangan sertakan file konfigurasi yang berisi kunci akses pengguna IAM Anda di dalam perangkat lunak manajemen kode sumber apa pun. Alat eksternal, seperti git-secret proyek open source, dapat membantu Anda dari secara tidak sengaja melakukan informasi sensitif ke repositori Git. Untuk informasi selengkapnya, lihat Identitas IAM (pengguna, grup pengguna, dan peran) di Panduan Pengguna IAM.

## Langkah selanjutnya

| Tugas untuk belajar | Topik |
| --- | --- |
| Pelajari cara menggunakan AWS Cloud9 IDE. | Memulai: tutorial basic dan Bekerja dengan IDE |

| Tugas yang lebih maju | Topik |
| --- | --- |
| Buat lingkungan AWS Cloud9 pengembangan, lalu gunakan AWS Cloud9 IDE untuk bekerja dengan kode di lingkungan baru Anda. | Menciptakan Lingkungan |
| Undang orang lain untuk menggunakan lingkungan baru Anda bersama Anda secara real time dan dengan dukungan obrolan teks. | Bekerja dengan Lingkungan Bersama |

# Menyiapkan Tim untuk AWS Cloud9

Topik ini menjelaskan cara menggunakan AWS IAM Identity Centeruntuk mengaktifkan beberapa pengguna dalam satu Akun AWS untuk digunakan AWS Cloud9. Untuk mengatur penggunaan AWS Cloud9 pola penggunaan lainnya, lihat Menyiapkan AWS Cloud9 petunjuk yang benar.

Instruksi ini mengasumsikan bahwa Anda memiliki atau akan memiliki akses administratif ke satu Akun AWS. Untuk informasi selengkapnya, lihat Pengguna Akun AWS root dan Membuat administrator dan grup pertama Anda di Panduan Pengguna IAM. Jika Anda sudah memiliki Akun AWS tetapi Anda tidak memiliki akses administratif ke akun, lihat Akun AWS administrator Anda.

**Warning**

Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti AWS IAM Identity Center.

**Note**

Anda dapat menggunakan IAM Identity Center alih-alih IAM untuk mengaktifkan beberapa pengguna dalam satu Akun AWS untuk digunakan. AWS Cloud9 Dalam pola penggunaan ini, single Akun AWS berfungsi sebagai akun manajemen untuk sebuah organisasi di AWS Organizations. Selain itu, organisasi itu tidak memiliki akun anggota. Untuk menggunakan IAM Identity Center, lewati topik ini dan ikuti petunjuk di Enterprise Setup. Untuk informasi terkait, lihat sumber daya berikut:

- Apa itu AWS Organizations in the AWS Organizations User Guide (IAM Identity Center membutuhkan penggunaan) AWS Organizations
- Apa itu AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center
- Video Pusat AWS Pengetahuan video 4 menit: Bagaimana cara memulai AWS Organizations YouTube
- Video 7 menit Kelola akses pengguna ke beberapa AWS akun menggunakan IAM Identity Center di YouTube
- Video 9 menit Cara mengatur Pusat Identitas IAM untuk pengguna Active Directory di lokasi Anda YouTube

Untuk mengaktifkan beberapa pengguna dalam satu Akun AWS untuk mulai menggunakan AWS Cloud9, mulai langkah-langkah yang untuk AWS sumber daya yang Anda miliki.

| Apakah Anda memiliki AWS akun? | Apakah Anda memiliki setidaknya satu grup IAM dan pengguna di akun itu? | Mulailah dengan langkah ini |
|---|---|---|
| Tidak | — | Langkah 1: Mendaftar untuk Akun AWS |
| Ya | Tidak | Langkah 2: Buat grup dan pengguna IAM, dan tambahkan pengguna ke grup |
| Ya | Ya | Langkah 3: Tambahkan izin AWS Cloud9 akses ke grup |

Topik


- Prasyarat
- Langkah 1: Buat grup dan pengguna IAM, dan tambahkan pengguna ke grup
- Langkah 2: Tambahkan izin AWS Cloud9 akses ke grup
- Langkah 3: Masuk ke AWS Cloud9 konsol
- Langkah selanjutnya


# Prasyarat

## Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS mengirimi Anda email konfirmasi setelah proses pendaftaran selesai. Kapan saja, Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan masuk https://aws.amazon.comke/ dan memilih Akun Saya.

## Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke AWS Management Consolesebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

   Untuk bantuan masuk dengan menggunakan pengguna root, lihat Masuk sebagai pengguna root di AWS Sign-In Panduan Pengguna.

2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuk, lihat Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root (konsol) Anda di Panduan Pengguna IAM.

Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center .

2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

  Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal AWS akses di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

   Untuk petunjuknya, lihat Membuat set izin di Panduan AWS IAM Identity Center Pengguna.
2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

   Untuk petunjuk, lihat Menambahkan grup di Panduan AWS IAM Identity Center Pengguna.

## Langkah 1: Buat grup dan pengguna IAM, dan tambahkan pengguna ke grup

Pada langkah ini, Anda membuat grup dan pengguna di AWS Identity and Access Management (IAM), menambahkan pengguna ke grup, dan kemudian menggunakan pengguna untuk mengakses AWS Cloud9. Ini adalah praktik terbaik AWS keamanan. Untuk informasi selengkapnya, lihat Praktik Terbaik IAM dalam Panduan Pengguna IAM.

Jika Anda sudah memiliki semua grup IAM dan pengguna yang Anda butuhkan, lanjutkan ke Langkah 3: Tambahkan izin AWS Cloud9 akses ke grup.

> **Note**
>
> Organisasi Anda mungkin telah memiliki grup IAM dan pengguna yang disiapkan untuk Anda. Jika organisasi Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut.

Anda dapat menyelesaikan tugas ini menggunakan AWS Management Console atau Antarmuka Baris Perintah AWS (AWS CLI).

Untuk menonton video berdurasi 9 menit yang terkait dengan prosedur konsol berikut, lihat Bagaimana cara mengatur pengguna IAM dan masuk ke kredensi AWS Management Console menggunakan IAM aktif. YouTube

### Langkah 1.1: Buat grup IAM dengan konsol

1. Masuk ke AWS Management Console, jika Anda belum masuk, di https://console.aws.amazon.com/codecommit.

   > **Note**
   >
   > Anda dapat masuk ke AWS Management Console dengan alamat email dan kata sandi yang diberikan saat Akun AWS dibuat. Ini disebut masuk sebagai pengguna root). Namun, ini bukan praktik AWS keamanan terbaik. Di masa mendatang, kami sarankan Anda masuk menggunakan kredensil untuk pengguna administrator di AWS akun. Pengguna administrator memiliki izin AWS akses yang serupa dengan pengguna Akun AWS root dan menghindari beberapa risiko keamanan terkait. Jika Anda tidak dapat masuk sebagai pengguna administrator, periksa dengan Akun AWS administrator Anda. Untuk informasi selengkapnya, lihat Membuat pengguna dan grup IAM pertama Anda di Panduan Pengguna IAM.

2. Buka konsol IAM. Untuk melakukan ini, di bilah AWS navigasi, pilih Layanan. Lalu pilih IAM.
3. Di panel navigasi konsol IAM, pilih Grup.
4. Pilih Buat Grup Baru.
5. Pada halaman Atur Nama Grup, untuk Nama Grupmasukkan nama untuk grup baru.
6. Pilih Langkah Selanjutnya.

7. Pada halaman Lampirkan Kebijakan, pilih Langkah Berikutnya tanpa melampirkan kebijakan apa pun. Anda akan melampirkan kebijakan di Langkah 3: Tambahkan izin AWS Cloud9 akses ke grup.
8. Pilih Buat group.

> **Note**
>
> Kami menyarankan Anda mengulangi prosedur ini untuk membuat setidaknya dua grup: satu grup untuk AWS Cloud9 pengguna, dan grup lain untuk AWS Cloud9 administrator. Praktik terbaik AWS keamanan ini dapat membantu Anda mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

Lewati ke depan ke Langkah 2.2: Buat pengguna IAM dan tambahkan pengguna ke grup pada konsol.

## Langkah 1.2: Buat grup IAM dengan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensil sementara yang AWS dikelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

1. Instal dan konfigurasikan AWS CLI di komputer Anda, jika Anda belum melakukannya. Untuk melakukannya, lihat petunjuk berikut di AWS Command Line Interface Panduan Pengguna:
   - Menginstal Antarmuka Baris AWS Perintah
   - Konfigurasi cepat

> **Note**
>
> Anda dapat mengonfigurasi AWS CLI menggunakan kredensi yang terkait dengan alamat email dan kata sandi yang diberikan saat Akun AWS dibuat. Ini disebut masuk sebagai pengguna root. Namun, ini bukan praktik AWS keamanan terbaik. Sebagai

gantinya, kami sarankan Anda mengonfigurasi kredensi AWS CLI penggunaan untuk pengguna administrator IAM di akun. AWS Pengguna administrator IAM memiliki izin AWS akses yang serupa dengan pengguna Akun AWS root dan menghindari beberapa risiko keamanan terkait. Jika Anda tidak dapat mengonfigurasi AWS CLI sebagai pengguna administrator IAM, periksa dengan Akun AWS administrator Anda. Untuk informasi lebih lanjut, lihat Membuat Grup dan Pengguna Admin IAM Pertama Anda dalam Panduan Pengguna IAM.

2. Jalankan perintah IAM `create-group`, menentukan nama grup baru (misalnya,`MyCloud9Group`).

```
aws iam create-group --group-name MyCloud9Group
```

> **Note**
>
> Kami menyarankan Anda mengulangi prosedur ini untuk membuat setidaknya dua grup: satu grup untuk AWS Cloud9 pengguna, dan grup lain untuk AWS Cloud9 administrator. Praktik terbaik AWS keamanan ini dapat membantu Anda mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

Lewati ke Langkah 2.2: Buat pengguna IAM dan tambahkan pengguna ke grup dengan AWS CLI.

## Langkah 1.3: Buat pengguna IAM dan tambahkan pengguna ke grup dengan konsol

1. Dengan konsol IAM terbuka dari prosedur sebelumnya, di panel navigasi, pilih Pengguna.
2. Pilih Tambahkan pengguna.
3. Untuk Nama pengguna, masukkan nama untuk pengguna baru.

> **Note**
>
> Anda dapat membuat beberapa pengguna secara bersamaan dengan memilih Tambahkan pengguna lain. Pengaturan lain dalam prosedur ini berlaku untuk setiap pengguna baru ini.

4. Pilih kotak centang Akses terprogram dan akses AWS Management Console . Hal ini memungkinkan pengguna baru untuk menggunakan berbagai alat developer AWS dan konsol layanan.

5. Tinggalkan pilihan default dari Kata sandi yang dihasilkan otomatis. Ini menciptakan kata sandi acak bagi pengguna baru untuk masuk ke konsol. Atau, pilih Kata sandi khusus dan masukkan kata sandi khusus untuk pengguna baru.
6. Jangan ubah pilihan default pada Memerlukan pengaturan ulang kata sandi. Tindakan ini akan meminta pengguna baru untuk mengubah sandi mereka setelah masuk ke konsol untuk pertama kalinya.
7. Pilih Berikutnya: Izin.
8. Jangan ubah pilihan default pada Menambahkan pengguna ke grup (atau Menambahkan pengguna ke grup untuk beberapa pengguna).
9. Di daftar grup, pilih kotak centang (bukan nama) di samping grup tempat Anda ingin menambahkan pengguna.
10. Pilih Berikutnya: Tinjauan.
11. Pilih Create user (Buat pengguna). Atau, Buat pengguna untuk banyak pengguna.
12. Di halaman terakhir wizard, lakukan salah satu hal berikut:
    - Di samping setiap pengguna baru, pilih Kirim email, dan ikuti petunjuk di layar untuk mengirim email kepada pengguna baru URL masuk konsol dan nama pengguna mereka. Kemudian, komunikasikan kepada setiap pengguna baru kata sandi masuk konsol mereka, ID kunci AWS akses, dan kunci akses AWS rahasia mereka secara terpisah.
    - Pilih Unduh .csv. Kemudian, komunikasikan ke setiap pengguna baru URL login konsol mereka, kata sandi masuk konsol, ID kunci AWS akses, dan kunci akses AWS rahasia yang ada di file yang diunduh.
    - Di samping setiap pengguna baru, pilih Tampilkan untuk Secret access key dan Kata Sandi. Kemudian komunikasikan kepada setiap pengguna baru URL login konsol mereka, kata sandi masuk konsol, ID kunci AWS akses, dan kunci akses AWS rahasia.

> **Note**
>
> Jika Anda tidak memilih Unduh.csv, ini adalah satu-satunya waktu Anda dapat melihat kunci akses AWS rahasia pengguna baru dan kata sandi masuk konsol. Untuk membuat kunci akses AWS rahasia baru atau kata sandi masuk konsol untuk pengguna baru, lihat yang berikut ini di Panduan Pengguna IAM.
>
> - Membuat, memodifikasi, dan melihat tombol akses (konsol)
> - Membuat, mengubah, atau menghapus kata sandi pengguna IAM (konsol)

13. Ulangi prosedur ini untuk setiap pengguna IAM tambahan yang ingin Anda buat, dan kemudian lewati ke Langkah 3: Tambahkan AWS Cloud9 izin akses ke grup.

## Langkah 1.4: Buat pengguna IAM dan tambahkan pengguna ke grup dengan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensil sementara yang AWS dikelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

1. Jalankan perintah `create-user` IAM untuk membuat pengguna, dengan menentukan nama pengguna baru (misalnya, `MyCloud9User`).

   ```
   aws iam create-user --user-name MyCloud9User
   ```

2. Jalankan perintah IAM `create-login-profile` untuk membuat kata sandi masuk konsol baru untuk pengguna, menentukan nama pengguna dan kata sandi masuk awal (misalnya, `MyC10ud9Us3r!`). Setelah pengguna masuk, AWS meminta pengguna untuk mengubah kata sandi masuk mereka.

   ```
   aws iam create-login-profile --user-name MyCloud9User --password MyC10ud9Us3r! --
   password-reset-required
   ```

   Jika Anda perlu membuat kata sandi masuk konsol pengganti untuk pengguna nanti, lihat Membuat, mengubah, atau menghapus kata sandi pengguna IAM (API, CLI, PowerShell) di Panduan Pengguna IAM.

3. Jalankan `create-access-key` perintah IAM untuk membuat kunci AWS akses baru dan kunci akses AWS rahasia yang sesuai untuk pengguna.

   ```
   aws iam create-access-key --user-name MyCloud9User
   ```

   Buat catatan tentang `AccessKeyId` dan `SecretAccessKey` nilai-nilai yang ditampilkan. Setelah Anda menjalankan `create-access-key` perintah IAM, ini adalah satu-satunya waktu Anda

dapat melihat kunci akses AWS rahasia pengguna. Jika Anda perlu membuat kunci akses AWS rahasia baru untuk pengguna nanti, lihat Membuat, memodifikasi, dan melihat kunci akses (API, CLI PowerShell,) di Panduan Pengguna IAM.

4. Jalankan perintah IAM `add-user-to-group` untuk menambahkan pengguna ke grup, menentukan nama grup dan pengguna.

```
aws iam add-user-to-group --group-name MyCloud9Group --user-name MyCloud9User
```

5. Berkomunikasi dengan pengguna URL login konsol mereka, kata sandi login konsol awal, ID kunci AWS akses, dan kunci akses AWS rahasia.
6. Ulangi prosedur ini untuk setiap pengguna IAM tambahan yang ingin Anda buat.

## Langkah 2: Tambahkan izin AWS Cloud9 akses ke grup

Secara default, sebagian besar grup dan pengguna IAM tidak memiliki akses ke salah satu Layanan AWS, termasuk AWS Cloud9, (pengecualian adalah grup administrator IAM dan pengguna administrator IAM, yang memiliki akses ke semua Layanan AWS dalam mereka secara Akun AWS default). Pada langkah ini, Anda menggunakan IAM untuk menambahkan izin AWS Cloud9 akses langsung ke grup IAM yang dimiliki oleh satu atau beberapa pengguna. Dengan cara ini, Anda dapat memastikan bahwa pengguna tersebut dapat mengakses AWS Cloud9.

> **Note**
>
> Organisasi Anda mungkin telah menyiapkan grup untuk Anda dengan izin akses yang sesuai. Jika organisasi Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut.

Anda dapat menyelesaikan tugas ini menggunakan AWS Management Console atau AWS CLI.

### Langkah 2.1: Tambahkan izin AWS Cloud9 akses ke grup dengan konsol

1. Masuk ke AWS Management Console, jika Anda belum masuk, di https://console.aws.amazon.com/codecommit.

> **Note**
>
> Anda dapat masuk ke AWS Management Console dengan alamat email dan kata sandi yang diberikan saat Akun AWS dibuat. Ini disebut masuk sebagai pengguna root. Namun, ini bukan praktik AWS keamanan terbaik. Di masa mendatang, kami sarankan Anda masuk menggunakan kredensil untuk pengguna administrator IAM di. Akun AWS Pengguna administrator memiliki izin AWS akses yang serupa dengan pengguna Akun AWS root dan menghindari beberapa risiko keamanan terkait. Jika Anda tidak dapat masuk sebagai pengguna administrator, periksa dengan Akun AWS administrator Anda. Untuk informasi lebih lanjut, lihat Membuat Grup dan Pengguna Admin IAM Pertama Anda dalam Panduan Pengguna IAM.

2. Buka konsol IAM. Untuk melakukan ini, di bilah AWS navigasi, pilih Layanan. Kemudian, pilih IAM.
3. Pilih Grup.
4. Pilih nama grup.
5. Putuskan apakah Anda ingin menambahkan izin akses AWS Cloud9 pengguna atau AWS Cloud9 administrator ke grup. Izin ini berlaku untuk setiap pengguna dalam grup.

   AWS Cloud9 izin akses pengguna memungkinkan setiap pengguna dalam grup untuk melakukan hal-hal berikut di dalamnya: Akun AWS

   - Buat lingkungan AWS Cloud9 pengembangan mereka sendiri.
   - Dapatkan informasi tentang lingkungan mereka sendiri.
   - Mengubah pengaturan untuk lingkungan mereka sendiri.

   AWS Cloud9 izin akses administrator memungkinkan setiap pengguna dalam grup untuk melakukan hal-hal tambahan di dalamnya: Akun AWS

   - Membuat lingkungan untuk diri mereka sendiri atau orang lain.
   - Dapatkan informasi tentang lingkungan untuk diri mereka sendiri atau orang lain.
   - Hapus lingkungan untuk diri mereka sendiri atau orang lain.
   - Mengubah pengaturan lingkungan untuk diri mereka sendiri atau orang lain.

   > **Note**
   >
   > Anda sebaiknya menambahkan pengguna dengan jumlah terbatas untuk Grup administrator AWS Cloud9 . Praktik terbaik AWS keamanan ini dapat membantu Anda

mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

6. Pada tab Izin, untuk Kebijakan Terkelola, pilih Lampirkan Kebijakan.
7. Dalam daftar nama kebijakan, pilih kotak di sebelah AWSCloud9Pengguna untuk izin akses AWS Cloud9 pengguna atau AWSCloud9Administrator untuk izin akses AWS Cloud9 administrator. Jika Anda tidak melihat salah satu nama kebijakan ini dalam daftar, masukkan nama kebijakan di kotak Filter untuk menampilkannya.
8. Pilih Lampirkan Kebijakan.

> **Note**
>
> Jika Anda memiliki lebih dari satu grup yang ingin Anda tambahkan izin AWS Cloud9 akses, ulangi prosedur ini untuk masing-masing grup tersebut.

Untuk melihat daftar izin akses yang diberikan kebijakan AWS terkelola ini ke grup, lihat kebijakan AWS terkelola (yang telah ditentukan sebelumnya).

Untuk mempelajari tentang izin AWS akses yang dapat ditambahkan ke grup selain izin akses yang diperlukan oleh AWS Cloud9, lihat Kebijakan terkelola dan kebijakan sebaris serta Memahami izin yang diberikan oleh kebijakan di Panduan Pengguna IAM.

Lewati ke Langkah 4: Masuk ke konsol AWS Cloud9.

### Langkah 2.2: Tambahkan izin AWS Cloud9 akses ke grup dengan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensil sementara yang AWS dikelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

1. Instal dan konfigurasikan AWS CLI di komputer Anda, jika Anda belum melakukannya. Untuk melakukannya, lihat petunjuk berikut di AWS Command Line Interface Panduan Pengguna:

- Menginstal Antarmuka Baris AWS Perintah
- Konfigurasi Cepat

> **Note**
>
> Anda dapat mengonfigurasi AWS CLI menggunakan kredensi yang terkait dengan alamat email dan kata sandi yang diberikan saat Akun AWS dibuat. Ini disebut masuk sebagai pengguna root. Namun, ini bukan praktik AWS keamanan terbaik. Sebagai gantinya, kami sarankan Anda mengonfigurasi kredensi AWS CLI penggunaan untuk pengguna administrator IAM di. Akun AWS Pengguna administrator IAM memiliki izin AWS akses yang serupa dengan pengguna Akun AWS root dan menghindari beberapa risiko keamanan terkait. Jika Anda tidak dapat mengonfigurasi AWS CLI sebagai pengguna administrator, periksa dengan Akun AWS administrator Anda. Untuk informasi selengkapnya, lihat Membuat Pengguna dan Grup Admin IAM Pertama Anda di Panduan Pengguna IAM.

2. Putuskan apakah akan menambahkan izin akses AWS Cloud9 pengguna atau AWS Cloud9 administrator ke grup. Izin ini berlaku untuk setiap pengguna dalam grup.

   AWS Cloud9 izin akses pengguna memungkinkan setiap pengguna dalam grup untuk melakukan hal-hal berikut di dalamnya: Akun AWS

   - Buat lingkungan AWS Cloud9 pengembangan mereka sendiri.
   - Dapatkan informasi tentang lingkungan mereka sendiri.
   - Mengubah pengaturan untuk lingkungan mereka sendiri.

   AWS Cloud9 izin akses administrator memungkinkan setiap pengguna dalam grup untuk melakukan hal-hal tambahan di dalamnya: Akun AWS

   - Membuat lingkungan untuk diri mereka sendiri atau orang lain.
   - Dapatkan informasi tentang lingkungan untuk diri mereka sendiri atau orang lain.
   - Hapus lingkungan untuk diri mereka sendiri atau orang lain.
   - Mengubah pengaturan lingkungan untuk diri mereka sendiri atau orang lain.

   > **Note**
   >
   > Anda sebaiknya menambahkan pengguna dengan jumlah terbatas untuk Grup administrator AWS Cloud9 . Praktik terbaik AWS keamanan ini dapat membantu Anda

mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

3. Jalankan `attach-group-policy` perintah IAM, tentukan nama grup dan Nama Sumber Daya Amazon (ARN) agar kebijakan izin AWS Cloud9 akses ditambahkan.

   Untuk izin akses AWS Cloud9 pengguna, tentukan ARN berikut.

   ```
   aws iam attach-group-policy --group-name MyCloud9Group --policy-arn
    arn:aws:iam::aws:policy/AWSCloud9User
   ```

   Untuk izin akses AWS Cloud9 administrator, tentukan ARN berikut.

   ```
   aws iam attach-group-policy --group-name MyCloud9Group --policy-arn
    arn:aws:iam::aws:policy/AWSCloud9Administrator
   ```

   > **Note**
   >
   > Jika Anda memiliki lebih dari satu grup yang ingin Anda tambahkan izin AWS Cloud9 akses, ulangi prosedur ini untuk masing-masing grup tersebut.

Untuk melihat daftar izin akses yang diberikan oleh kebijakan AWS terkelola ini ke grup, lihat Kebijakan AWS Terkelola (Predefined).

Untuk mempelajari tentang izin AWS akses yang dapat ditambahkan ke grup selain izin akses yang diperlukan oleh AWS Cloud9, lihat Kebijakan Terkelola dan Kebijakan Sebaris dan Memahami Izin yang Diberikan oleh Kebijakan dalam Panduan Pengguna IAM.

## Langkah 3: Masuk ke AWS Cloud9 konsol

Setelah Anda menyelesaikan langkah-langkah sebelumnya dalam topik ini, Anda dan pengguna Anda siap untuk masuk ke AWS Cloud9 konsol.

1. Jika Anda sudah masuk ke AWS Management Console sebagai pengguna Akun AWS root, keluar dari konsol.
2. Buka AWS Cloud9 konsol, di https://console.aws.amazon.com/cloud9/.

3. Masukkan Akun AWS nomor untuk pengguna IAM yang Anda buat atau identifikasi sebelumnya, lalu pilih Berikutnya.

> **Note**
>
> Jika Anda tidak melihat opsi untuk memasukkan nomor AWS akun, pilih Masuk ke akun lain. Masukkan Akun AWS nomor di halaman berikutnya, lalu pilih Berikutnya.

4. Masukkan kredenal masuk pengguna IAM yang Anda buat atau identifikasi sebelumnya, lalu pilih Masuk.
5. Jika diminta, ikuti petunjuk di layar untuk mengubah kata sandi masuk awal pengguna Anda. Simpan kata sandi masuk baru Anda di lokasi yang aman.

AWS Cloud9 Konsol ditampilkan, dan Anda dapat mulai menggunakan AWS Cloud9.

## Langkah selanjutnya

| Tugas | Lihat topik ini |
| --- | --- |
| Batasi AWS Cloud9 penggunaan untuk orang lain di Anda Akun AWS, untuk mengontrol biaya. | Opsi pengaturan tambahan |
| Buat lingkungan AWS Cloud9 pengembangan, lalu gunakan AWS Cloud9 IDE untuk bekerja dengan kode di lingkungan baru Anda. | Menciptakan lingkungan |
| Pelajari cara menggunakan AWS Cloud9 IDE. | Memulai: tutorial basic, dan Bekerja dengan IDE |
| Undang orang lain untuk menggunakan lingkungan baru Anda bersama Anda secara real time dan dengan dukungan obrolan teks. | Bekerja dengan lingkungan bersama |

# Pengaturan perusahaan untuk AWS Cloud9

Topik ini menjelaskan cara menggunakan AWS IAM Identity Centeruntuk memungkinkan satu atau lebih Akun AWS untuk digunakan AWS Cloud9 dalam suatu perusahaan. Untuk mengatur penggunaan AWS Cloud9 pola penggunaan lainnya, lihat Menyiapkan AWS Cloud9 petunjuk yang benar.

> **Warning**
>
> Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata. Sebaliknya, gunakan federasi dengan penyedia identitas seperti AWS IAM Identity Center.

Instruksi ini mengasumsikan bahwa Anda memiliki atau akan memiliki akses administratif ke organisasi di AWS Organizations. Jika Anda belum memiliki akses administratif ke organisasi AWS Organizations, lihat Akun AWS administrator Anda. Untuk informasi selengkapnya, lihat sumber daya berikut:

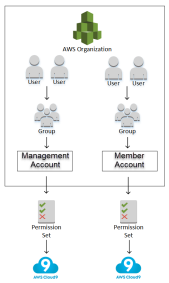
- Mengelola izin akses untuk AWS Organisasi Anda di Panduan Pengguna AWS Organizations (Pusat Identitas IAM memerlukan penggunaan) AWS Organizations
- Ikhtisar mengelola izin akses ke Sumber Daya Pusat Identitas IAM Anda di Panduan Pengguna AWS IAM Identity Center
- MenggunakanAWS Control Tower, yang merupakan layanan yang dapat Anda gunakan untuk mengatur dan mengatur lingkungan AWS multi-akun. AWS Control Tower melibatkan kemampuan lain Layanan AWS, termasuk, AWS Service Catalog dan AWS Organizations AWS IAM Identity Center, untuk membangun landing zone dalam waktu kurang dari satu jam.

Untuk informasi pengantar yang terkait dengan topik ini, lihat sumber daya berikut:

- Apa itu AWS Organizations in the AWS Organizations User Guide (IAM Identity Center membutuhkan penggunaan) AWS Organizations
- Apa itu AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center
- Memulai AWS Control Tower di Panduan Pengguna AWS Control Tower
- Video Pusat AWS Pengetahuan video 4 menit: Bagaimana cara memulai dengan AWS Organizations on YouTube

- Video 7 menit Kelola akses pengguna ke beberapa AWS akun yang digunakan AWS IAM Identity Center di YouTube
- Video 9 menit Cara mengatur AWS Single Sign On untuk pengguna Active Directory di lokasi Anda YouTube

Diagram konseptual berikut menunjukkan apa yang Anda dapatkan.




Untuk mengaktifkan satu atau lebih Akun AWS untuk mulai menggunakan AWS Cloud9 dalam suatu perusahaan, ikuti langkah-langkah sesuai dengan AWS sumber daya yang sudah Anda miliki.

| Apakah Anda memiliki Akun AWS yang dapat atau tidak berfungsi sebagai akun manajemen untuk organisasi di AWS Organizations? | Apakah Anda memiliki organisasi AWS Organizations untuk akun manajemen itu? | Apakah semua Akun AWS anggota yang dicari dari organisasi itu? | Apakah organisasi itu diatur untuk menggunakan IAM Identity Center? | Apakah organisasi itu diatur dengan semua grup yang diinginkan dan pengguna yang ingin menggunakan AWS Cloud9? | Mulailah dengan langkah ini |
|---|---|---|---|---|---|
| Tidak | — | — | — | — | Langkah 1: Buat akun manajemen untuk organisasi |

| Apakah Anda memiliki Akun AWS yang dapat atau tidak berfungsi sebagai akun manajemen untuk organisasi di AWS Organizations? | Apakah Anda memiliki organisasi AWS Organizations untuk akun manajemen itu? | Apakah semua Akun AWS anggota yang dicari dari organisasi itu? | Apakah organisasi itu diatur untuk menggunakan IAM Identity Center? | Apakah organisasi itu diatur dengan semua grup yang diinginkan dan pengguna yang ingin menggunakan AWS Cloud9? | Mulailah dengan langkah ini |
|---|---|---|---|---|---|
| Ya | Tidak | — | — | — | Langkah 2: Buat organisasi untuk akun manajemen |
| Ya | Ya | Tidak | — | — | Langkah 3: Tambahkan akun anggota ke organisasi |
| Ya | Ya | Ya | Tidak | — | Langkah 4: Aktifkan Pusat Identitas IAM di seluruh organisasi |

| Apakah Anda memiliki Akun AWS yang dapat atau tidak berfungsi sebagai akun manajemen untuk organisasi di AWS Organizations? | Apakah Anda memiliki organisasi AWS Organizations untuk akun manajemen itu? | Apakah semua Akun AWS anggota yang dicari dari organisasi itu? | Apakah organisasi itu diatur untuk menggunakan IAM Identity Center? | Apakah organisasi itu diatur dengan semua grup yang diinginkan dan pengguna yang ingin menggunakan AWS Cloud9? | Mulailah dengan langkah ini |
|---|---|---|---|---|---|
| Ya | Ya | Ya | Ya | Tidak | Langkah 5. Menyiapkan grup dan pengguna dalam organisasi |
| Ya | Ya | Ya | Ya | Ya | Langkah 6. Memungkinkan grup dan pengguna dalam organisasi untuk digunakan AWS Cloud9 |

## Langkah 1: Buat akun pengelolaan untuk organisasi

> **Note**
>
> Perusahaan Anda mungkin sudah memiliki akun pengelolaan yang disiapkan untuk Anda. Jika perusahaan Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut. Jika Anda sudah memiliki akun pengelolaan, lewati Langkah 2: Buat organisasi untuk akun pengelolaan.

Untuk menggunakan AWS IAM Identity Center (IAM Identity Center), Anda harus memiliki Akun AWS file. Anda Akun AWS berfungsi sebagai akun manajemen untuk organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat diskusi tentang akun pengelolaan dalam Terminologi organisasi dan konsep AWS di AWS Organizations Panduan Pengguna.

Untuk menonton video berdurasi 4 menit yang terkait dengan prosedur berikut, lihat Membuat akun Amazon YouTube Web Services aktif.

Untuk membuat akun pengelolaan:

1. Pergi ke https://aws.amazon.com/.
2. Pilih Masuklah ke konsol IAM.
3. Pilih Buat yang baruAkun AWS.
4. Selesaikan prosesnya dengan mengikuti petunjuk di layar. Ini termasuk memberikan AWS alamat email dan informasi kartu kredit Anda. Anda juga harus menggunakan ponsel Anda untuk memasukkan kode yang AWS memberi Anda.

Setelah Anda selesai membuat akun, AWS akan mengirimkan email konfirmasi. Jangan lanjut ke langkah berikutnya sampai Anda mendapatkan konfirmasi ini.

## Langkah 2: Buat organisasi untuk akun pengelolaan

> **Note**
>
> Perusahaan Anda mungkin sudah AWS Organizations menyiapkan untuk menggunakan akun manajemen. Jika perusahaan Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut. Jika Anda sudah AWS Organizations

menyiapkan untuk menggunakan akun manajemen, lanjutkan ke Langkah 3: Tambahkan akun anggota ke organisasi.

Untuk menggunakan IAM Identity Center, Anda harus memiliki organisasi AWS Organizations yang menggunakan akun manajemen. Untuk informasi selengkapnya, lihat diskusi tentang organisasi dalam Terminologi organisasi dan konsep AWS di AWS Panduan Pengguna Organisasi.

Untuk membuat organisasi AWS Organizations untuk manajemen Akun AWS, ikuti petunjuk ini di Panduan AWS Organizations Pengguna:

1. Menciptakan organisasi
2. Mengaktifkan semua fitur di organisasi Anda

Untuk menonton video 4 menit yang terkait dengan prosedur ini, lihat Video Pusat AWS Pengetahuan: Bagaimana cara memulai dengan AWS Organizations on YouTube.

## Langkah 3: Tambahkan akun anggota ke organisasi

**Note**

Bisnis Anda mungkin sudah AWS Organizations menyiapkan akun anggota yang diinginkan. Jika perusahaan Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut. Jika Anda sudah AWS Organizations menyiapkan akun anggota yang diinginkan, lanjutkan ke Langkah 4: Aktifkan Pusat Identitas IAM di seluruh organisasi.

Pada langkah ini, Anda menambahkan apa pun Akun AWS yang akan berfungsi sebagai akun anggota untuk organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat diskusi tentang akun anggota dalam Terminologi organisasi dan konsep AWS di AWS Organizations Panduan Pengguna.

**Note**

Anda tidak perlu menambahkan akun anggota ke organisasi. Anda dapat menggunakan Pusat Identitas IAM hanya dengan satu akun manajemen di organisasi. Nantinya, Anda dapat menambahkan akun anggota ke organisasi, jika Anda mau. Jika Anda tidak ingin

menambahkan akun anggota apa pun sekarang, lanjutkan ke Langkah 4: Aktifkan Pusat Identitas IAM di seluruh organisasi.

Untuk menambahkan akun anggota ke organisasi di AWS Organizations, ikuti salah satu atau kedua set instruksi berikut dalam Panduan AWS Organizations Pengguna. Ulangi instruksi ini sebanyak yang diperlukan sampai Anda memiliki semua Akun AWS yang Anda inginkan sebagai anggota organisasi:

- Membuat sebuah Akun AWS di organisasi Anda
- Mengundang Akun AWS untuk bergabung dengan organisasi Anda

## Langkah 4: Aktifkan Pusat Identitas IAM di seluruh organisasi

**Note**

Perusahaan Anda mungkin sudah AWS Organizations menyiapkan untuk menggunakan IAM Identity Center. Jika perusahaan Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut. Jika Anda sudah AWS Organizations menyiapkan untuk menggunakan IAM Identity Center, lewati ke Langkah 5. Menyiapkan grup dan pengguna dalam organisasi.

Pada langkah ini, Anda mengaktifkan organisasi AWS Organizations untuk menggunakan IAM Identity Center. Untuk melakukan ini, ikuti rangkaian instruksi ini di Panduan AWS IAM Identity Center Pengguna:

1. Prasyarat Pusat Identitas IAM
2. Aktifkan Pusat Identitas IAM

## Langkah 5. Menyiapkan grup dan pengguna dalam organisasi

**Note**

Perusahaan Anda mungkin sudah AWS Organizations menyiapkan grup dan pengguna dari direktori Pusat Identitas IAM atau direktori AD Connector yang dikelola. AWS

Managed Microsoft AD AWS Directory Service Jika perusahaan Anda memiliki Akun AWS administrator, tanyakan kepada orang tersebut sebelum memulai prosedur berikut. Jika Anda sudah AWS Organizations menyiapkan grup dan pengguna dari direktori Pusat Identitas IAM atau AWS Directory Service, lewati ke Langkah 6. Aktifkan grup dan pengguna dalam organisasi untuk menggunakan AWS Cloud9.

Pada langkah ini, Anda membuat grup dan pengguna di direktori Pusat Identitas IAM untuk organisasi. Atau, Anda terhubung ke direktori AD Connector AWS Managed Microsoft AD atau yang dikelola AWS Directory Service untuk organisasi. Pada langkah selanjutnya, Anda memberikan izin akses yang diperlukan kepada kelompok-kelompok untuk menggunakan AWS Cloud9.

- Jika Anda menggunakan direktori Pusat Identitas IAM untuk organisasi, ikuti kumpulan petunjuk ini di Panduan AWS IAM Identity Center Pengguna. Ulangi langkah-langkah ini sebanyak yang diperlukan sampai Anda memiliki semua grup dan pengguna yang Anda inginkan:
  1. Tambahkan grup. Sebaiknya buat setidaknya satu grup untuk AWS Cloud9 administrator mana pun di seluruh organisasi. Kemudian, ulangi langkah ini untuk membuat grup lain untuk semua AWS Cloud9 pengguna di seluruh organisasi. Secara opsional, Anda juga dapat mengulangi langkah ini untuk membuat grup ketiga untuk semua pengguna di seluruh organisasi tempat Anda ingin berbagi lingkungan AWS Cloud9 pengembangan yang ada. Tapi, jangan biarkan mereka menciptakan lingkungan sendiri. Untuk kemudahan penggunaan, sebaiknya beri nama grup ini `AWSCloud9Administrators`, `AWSCloud9Users`, dan `AWSCloud9EnvironmentMembers`, sesuai urutannya. Untuk informasi selengkapnya, lihat AWS Kebijakan terkelola (ditentukan sebelumnya) untuk AWS Cloud9.
  2. Tambah pengguna.
  3. Menambahkan pengguna ke grup. Tambahkan AWS Cloud9 administrator apa pun ke `AWSCloud9Administrators` grup, ulangi langkah ini untuk menambahkan AWS Cloud9 pengguna ke `AWSCloud9Users` grup. Secara opsional, ulangi langkah ini untuk menambahkan pengguna yang tersisa ke `AWSCloud9EnvironmentMembers` grup. Menambahkan pengguna ke grup adalah praktik terbaik AWS keamanan yang dapat membantu Anda mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.
- Jika Anda menggunakan direktori AD Connector AWS Managed Microsoft AD atau AD Connector yang Anda kelola AWS Directory Service untuk organisasi, lihat Connect to your Microsoft AD directory in the AWS IAM Identity Center User Guide.

# Langkah 6. Memungkinkan grup dan pengguna dalam organisasi untuk digunakan AWS Cloud9

Secara default, sebagian besar pengguna dan grup dalam organisasi AWS Organizations tidak memiliki akses ke salah satu Layanan AWS, termasuk AWS Cloud9. Pada langkah ini, Anda menggunakan Pusat Identitas IAM untuk memungkinkan grup dan pengguna di seluruh organisasi AWS Organizations untuk digunakan AWS Cloud9 dalam kombinasi akun yang berpartisipasi.

1. Di konsol Pusat Identitas IAM, pilih Akun AWSdi panel navigasi layanan.
2. Pilih tab Rangkaian Izin.
3. Pilih Buat rangkaian izin yang ditentukan.
4. Pilih Membuat rangkaian izin kustom.
5. Masukkan Nama untuk rangkaian izin ini. Sebaiknya buat setidaknya satu set izin untuk AWS Cloud9 administrator mana pun di seluruh organisasi. Kemudian, ulangi langkah 3 hingga 10 dalam prosedur ini untuk membuat set izin lain untuk semua AWS Cloud9 pengguna di seluruh organisasi. Secara opsional, Anda juga dapat mengulangi langkah 3 hingga 10 dalam prosedur ini untuk membuat set izin ketiga untuk semua pengguna di seluruh organisasi yang ingin Anda bagikan dengan lingkungan AWS Cloud9 pengembangan yang ada. Tapi, jangan biarkan mereka menciptakan lingkungan sendiri. Untuk kemudahan penggunaan, sebaiknya beri nama rangkaian izin ini `AWSCloud9AdministratorsPerms`, `AWSCloud9UsersPerms`, dan `AWSCloud9EnvironmentMembersPerms`, sesuai urutannya. Untuk informasi selengkapnya, lihat AWS Kebijakan terkelola (ditentukan sebelumnya) untuk AWS Cloud9.
6. Masukkan pilihan Deskripsi untuk rangkaian izin.
7. Pilih Durasi sesi untuk rangkaian izin, atau biarkan durasi sesi default 1 jam.
8. Pilih Lampirkan kebijakan AWS terkelola.
9. Dalam daftar kebijakan, pilih salah satu kotak berikut di samping entri Nama kebijakan yang benar. (Jangan pilih nama kebijakan itu sendiri. Jika Anda tidak melihat nama kebijakan dalam daftar, masukkan nama kebijakan di kotak Pencarian untuk menampilkannya.)
   - Untuk set `AWSCloud9AdministratorsPerms` izin, pilih AWSCloud9Administrator.
   - Untuk set `AWSCloud9UsersPerms` izin, pilih AWSCloud9Pengguna.
   - Secara opsional, untuk set `AWSCloud9EnvironmentMembersPerms` izin, pilih AWSCloud9EnvironmentMember.

> **Note**
>
> Untuk mempelajari kebijakan yang dapat Anda tambahkan selain kebijakan yang diperlukan AWS Cloud9, lihat Kebijakan terkelola dan kebijakan sebaris serta Memahami izin yang diberikan oleh kebijakan di Panduan Pengguna IAM.

10. Pilih Buat.
11. Setelah Anda selesai membuat semua set izin yang Anda inginkan, pada tab AWS organisasi, pilih Akun AWS yang ingin Anda tetapkan izin AWS Cloud9 akses. Jika tab AWS organisasi tidak terlihat, maka di panel navigasi layanan, pilih Akun AWS. Ini menampilkan tab AWS organisasi.
12. Pilih Tetapkan pengguna.
13. Pada tab Grup, pilih kotak yang berada di sebelah nama grup yang ingin Anda tetapkan izin AWS Cloud9 akses. Jangan memilih nama grup itu sendiri.
    - Jika Anda menggunakan direktori Pusat Identitas IAM untuk organisasi, Anda mungkin telah membuat grup yang diberi nama AWSCloud9Administrator untuk AWS Cloud9 administrator.
    - Jika Anda menggunakan direktori AD Connector AWS Managed Microsoft AD atau yang Anda kelola AWS Directory Service untuk organisasi, pilih ID direktori. Selanjutnya, masukkan sebagian atau seluruh nama grup dan pilih Cari direktori yang terhubung. Terakhir, pilih kotak di sebelah nama grup yang ingin Anda tetapkan izin AWS Cloud9 akses.

> **Note**
>
> Sebaiknya tetapkan izin AWS Cloud9 akses ke grup, bukan untuk pengguna individu. Praktik terbaik AWS keamanan ini dapat membantu Anda mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

14. Pilih Selanjutnya: Rangkaian izin.
15. Pilih kotak di samping nama set izin yang ingin Anda tetapkan ke grup ini (misalnya, AWSCloud9AdministratorsPermsuntuk grup AWS Cloud9 administrator). Jangan memilih nama set izin itu sendiri.
16. Pilih Selesai.
17. Pilih Lanjutkan ke Akun AWS.
18. Ulangi langkah 11 hingga 17 dalam prosedur ini untuk AWS Cloud9 mendapatkan izin akses tambahan yang ingin Anda tetapkan Akun AWS di seluruh organisasi.

# Langkah 7: Mulai gunakan AWS Cloud9

Setelah Anda menyelesaikan langkah-langkah sebelumnya dalam topik ini, Anda dan pengguna Anda siap untuk masuk ke IAM Identity Center dan mulai menggunakan AWS Cloud9.

1. Jika Anda sudah masuk ke AWS akun atau Pusat Identitas IAM, keluar. Untuk melakukannya, lihat Bagaimana cara keluar dari AWS akun saya di situs web AWS Support atau Cara keluar dari portal pengguna di Panduan AWS IAM Identity Center Pengguna.
2. Untuk masuk ke Pusat Identitas IAM, ikuti petunjuk di Cara menerima undangan untuk bergabung dengan Pusat Identitas IAM di AWS IAM Identity Center Panduan Pengguna. Ini termasuk membuka URL masuk yang unik dan masuk dengan kredenal masuk yang unik. Akun AWS Administrator Anda akan mengirim email kepada Anda informasi ini atau memberikannya kepada Anda.

   > **Note**
   >
   > Pastikan untuk menandai URL masuk unik yang Anda berikan. Dengan cara ini, Anda dapat dengan mudah kembali ke sana nanti. Selain itu, pastikan untuk menyimpan kredenal masuk unik untuk URL ini di lokasi yang aman.
   > Kombinasi URL, nama pengguna, dan kata sandi ini mungkin berubah tergantung pada berbagai tingkat izin AWS Cloud9 akses yang diberikan Akun AWS administrator kepada Anda. Misalnya, Anda dapat menggunakan satu URL, nama pengguna, dan kata sandi untuk mendapatkan akses AWS Cloud9 administrator ke satu akun. Anda dapat menggunakan URL, nama pengguna, dan kata sandi yang berbeda yang hanya memungkinkan akses AWS Cloud9 pengguna ke akun yang berbeda.
3. Setelah Anda masuk ke IAM Identity Center, pilih Akun AWSubin.
4. Pilih nama tampilan pengguna Anda dari daftar drop-down yang ditampilkan. Jika lebih dari satu nama ditampilkan, pilih nama yang ingin Anda mulai gunakan AWS Cloud9. Jika Anda tidak yakin nama mana yang harus dipilih, lihat Akun AWS administrator Anda.
5. Pilih Konsol manajemen di sebelah nama tampilan pengguna Anda. Jika lebih dari satu tautan konsol Manajemen ditampilkan, pilih tautan yang berada di sebelah set izin yang benar. Jika Anda tidak yakin tautan mana yang harus dipilih, lihat Akun AWS administrator Anda.
6. Dari AWS Management Console, lakukan salah satu hal berikut:
   - PilihCloud9, jika sudah ditampilkan.
   - Perluas Semua layanan, lalu pilih Cloud9.

- Di kotak Temukan layanan, jenis Cloud9, lalu tekan `Enter`.
- Di bilah AWS navigasi, pilih Layanan, lalu pilih Cloud9.

AWS Cloud9 Konsol ditampilkan, dan Anda dapat mulai menggunakan AWS Cloud9.

## Langkah selanjutnya

| Tugas | Lihat topik ini |
| --- | --- |
| Buat lingkungan AWS Cloud9 pengembangan, lalu gunakan AWS Cloud9 IDE untuk bekerja dengan kode di lingkungan baru Anda. | Menciptakan lingkungan |
| Pelajari cara menggunakan AWS Cloud9 IDE. | Memulai: tutorial basic dan Bekerja dengan IDE |
| Undang orang lain untuk menggunakan lingkungan baru Anda bersama Anda secara real time dan dengan dukungan obrolan teks. | Bekerja dengan lingkungan bersama |

# Opsi pengaturan tambahan untuk AWS Cloud9

Topik ini mengasumsikan Anda telah menyelesaikan langkah-langkah penyiapan di Pengaturan Tim atau Pengaturan Perusahaan.

Di Pengaturan Tim atau Pengaturan Perusahaan, Anda membuat grup dan menambahkan izin AWS Cloud9 akses langsung ke grup tersebut. Ini untuk memastikan bahwa pengguna dalam grup tersebut dapat mengakses AWS Cloud9. Dalam topik ini, Anda menambahkan lebih banyak izin akses untuk membatasi jenis lingkungan yang dapat dibuat oleh pengguna dalam grup tersebut. Ini dapat membantu mengontrol biaya yang terkait AWS Cloud9 dengan AWS akun dan organisasi.

Untuk menambahkan izin akses ini, Anda membuat serangkaian kebijakan Anda sendiri yang menentukan AWS izin akses yang ingin diterapkan. Kami menyebut masing-masing kebijakan yang dikelola pelanggan. Kemudian, Anda melampirkan kebijakan yang dikelola pelanggan tersebut ke grup yang menjadi milik pengguna. Dalam beberapa skenario, Anda juga harus melepaskan kebijakan AWS terkelola yang ada yang sudah dilampirkan ke grup tersebut. Untuk mengatur hal ini, ikuti prosedur dalam topik ini.

> **Note**
>
> Prosedur berikut mencakup melampirkan dan melepaskan kebijakan hanya untuk AWS Cloud9 pengguna. Prosedur ini mengasumsikan Anda sudah memiliki grup AWS Cloud9 pengguna dan grup AWS Cloud9 administrator yang terpisah. Mereka juga berasumsi bahwa Anda hanya memiliki sejumlah pengguna terbatas di grup AWS Cloud9 administrator. Praktik terbaik AWS keamanan ini dapat membantu Anda mengontrol, melacak, dan memecahkan masalah dengan akses AWS sumber daya dengan lebih baik.

## Langkah 1: Buat kebijakan terkelola pelanggan

Anda dapat membuat kebijakan terkelola pelanggan menggunakan AWS Management Console atau AWS Antarmuka Baris Perintah (AWS CLI).

> **Note**
>
> Langkah ini mencakup pembuatan kebijakan terkelola pelanggan untuk grup IAM saja. Untuk membuat set izin khusus untuk grup AWS IAM Identity Center, lewati langkah ini dan ikuti petunjuk di Buat Set Izin di Panduan AWS IAM Identity Center Pengguna. Dalam topik ini, ikuti petunjuk untuk membuat kumpulan izin khusus. Untuk kebijakan izin kustom terkait, lihat Contoh kebijakan terkelola pelanggan untuk tim yang menggunakan AWS Cloud9 pada bagian selanjutnya dalam topik ini.

### Langkah 1.1: Membuat kebijakan terkelola pelanggan menggunakan konsol

1. Masuk ke AWS Management Console, jika Anda belum masuk.

   Kami menyarankan Anda masuk menggunakan kredensi untuk pengguna administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.
2. Buka konsol IAM. Untuk melakukannya, di bilah navigasi konsol tersebut, pilih Layanan. Lalu pilih IAM.
3. Di panel navigasi layanan, pilih Kebijakan.
4. Pilih Buat kebijakan.
5. Di tab JSON, tempel salah satu contoh kebijakan terkelola pelanggan yang kami sarankan.

> **Note**
>
> Anda juga dapat membuat kebijakan terkelola pelanggan. Untuk informasi selengkapnya, lihat Referensi Kebijakan IAM JSON di Panduan Pengguna IAM dan dokumentasinya. Layanan AWS

6. Pilih Tinjau kebijakan.
7. Pada halaman Tinjau kebijakan, ketik Nama dan Deskripsi opsional untuk kebijakan, lalu pilih Buat kebijakan.

Ulangi langkah ini untuk setiap kebijakan tambahan yang dikelola pelanggan yang ingin Anda buat. Kemudian, lewati ke Tambahkan kebijakan terkelola pelanggan ke grup menggunakan konsol.

### Langkah 1.2: Buat kebijakan yang dikelola pelanggan menggunakan AWS CLI

1. Di komputer tempat Anda menjalankan AWS CLI, buat file untuk menjelaskan kebijakan (misalnya,`policy.json`).

   Jika Anda membuat file dengan nama file yang berbeda, lakukan penggantian di seluruh prosedur ini.
2. Tempelkan salah satu contoh kebijakan terkelola pelanggan yang kami sarankan ke dalam file `policy.json`.

> **Note**
>
> Anda juga dapat membuat kebijakan terkelola pelanggan. Untuk informasi selengkapnya, lihat Referensi Kebijakan JSON IAM di Panduan Pengguna IAM dan AWS dokumentasi layanan.

3. Dari terminal atau command prompt, beralihlah ke direktori yang berisi file `policy.json`.
4. Jalankan perintah IAM `create-policy`, tentukan nama untuk kebijakan dan file `policy.json`.

```
aws iam create-policy --policy-document file://policy.json --policy-name MyPolicy
```

   Pada perintah sebelumnya, ganti `MyPolicy` dengan nama kebijakan.

Loncat ke depan untuk Menambahkan Kebijakan yang dikelola pelanggan ke Grup Menggunakan AWS CLI.

## Langkah 2: Menambahkan kebijakan terkelola pelanggan ke grup

Anda dapat menambahkan kebijakan terkelola pelanggan ke grup dengan menggunakan AWS Management Consoleatau Antarmuka Baris AWS Perintah (AWS CLI). Untuk informasi selengkapnya, lihat Contoh kebijakan terkelola pelanggan untuk tim yang menggunakan AWS Cloud9.

> **Note**
>
> Langkah ini mencakup penambahan kebijakan terkelola pelanggan ke grup IAM saja. Untuk menambahkan set izin khusus ke grup AWS IAM Identity Center, lewati langkah ini dan ikuti petunjuk di Tetapkan Akses Pengguna di Panduan AWS IAM Identity Center Pengguna sebagai gantinya.

### Langkah 2.1: Menambahkan kebijakan yang dikelola pelanggan ke grup menggunakan konsol

1. Dengan konsol IAM terbuka dari prosedur sebelumnya, di panel navigasi pada layanan, pilih Grup.
2. Pilih nama grup.
3. Pada tab Izin, untuk Kebijakan Terkelola, pilih Lampirkan Kebijakan.
4. Dalam daftar nama kebijakan, pilih kotak di samping setiap kebijakan terkelola pelanggan yang ingin dilampirkan ke grup. Jika Anda tidak melihat nama kebijakan tertentu dalam daftar, masukkan nama kebijakan di kotak Filter untuk menampilkannya.
5. Pilih Lampirkan Kebijakan.

### Langkah 2.2: Tambahkan kebijakan yang dikelola pelanggan ke grup menggunakan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola

tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

Jalankan perintah IAM `attach-group-policy`, tentukan nama grup dan Amazon Resource Name (ARN) kebijakan.

```
aws iam attach-group-policy --group-name MyGroup --policy-arn
 arn:aws:iam::123456789012:policy/MyPolicy
```

Dengan perintah sebelumnya, ganti `MyGroup` dengan nama grup. Ganti `123456789012` dengan ID AWS akun. Dan ganti `MyPolicy` dengan nama kebijakan yang dikelola pelanggan.

## Langkah selanjutnya

| Tugas | Lihat topik ini |
| --- | --- |
| Buat lingkungan AWS Cloud9 pengembangan, lalu gunakan AWS Cloud9 IDE untuk bekerja dengan kode di lingkungan baru Anda. | Pembuatan lingkungan |
| Pelajari cara menggunakan AWS Cloud9 IDE. | Memulai: tutorial basic dan Bekerja dengan IDE |
| Undang orang lain untuk menggunakan lingkungan baru Anda bersama Anda secara real time dan dengan dukungan obrolan teks. | Bekerja dengan Lingkungan Bersama |

## Contoh kebijakan terkelola pelanggan untuk tim, menggunakan AWS Cloud9

Berikut ini adalah beberapa contoh kebijakan yang dapat Anda gunakan untuk membatasi lingkungan yang dapat dibuat oleh pengguna dalam grup. Akun AWS

- Mencegah pengguna dalam grup membuat lingkungan
- Mencegah pengguna dalam grup membuat EC2 lingkungan
- Izinkan pengguna dalam grup untuk membuat EC2 lingkungan hanya dengan jenis EC2 instans Amazon tertentu

- Izinkan pengguna dalam grup untuk membuat hanya satu EC2 lingkungan per AWS Wilayah

## Mencegah pengguna dalam grup agar tidak membuat lingkungan

Kebijakan terkelola pelanggan berikut, saat dilampirkan ke grup AWS Cloud9 pengguna, mencegah pengguna tersebut membuat lingkungan dalam file Akun AWS. Ini berguna jika Anda ingin pengguna administrator Akun AWS mengelola lingkungan pembuatan. Jika tidak, pengguna dalam grup AWS Cloud9 pengguna melakukan ini.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "cloud9:CreateEnvironmentEC2",
        "cloud9:CreateEnvironmentSSH"
      ],
      "Resource": "*"
    }
  ]
}
```

Kebijakan terkelola pelanggan sebelumnya secara eksplisit mengesampingkan `"Effect": "Allow"` untuk `"Action": "cloud9:CreateEnvironmentEC2"` dan `"cloud9:CreateEnvironmentSSH"` seterusnya `"Resource": "*"` dalam kebijakan `AWSCloud9User` terkelola yang sudah dilampirkan ke grup pengguna. AWS Cloud9

## Mencegah pengguna dalam grup membuat EC2 lingkungan

Kebijakan terkelola pelanggan berikut, saat dilampirkan ke grup AWS Cloud9 pengguna, mencegah pengguna tersebut membuat EC2 lingkungan dalam file Akun AWS. Ini berguna jika Anda ingin pengguna administrator Akun AWS mengelola EC2 lingkungan pembuatan. Jika tidak, pengguna dalam grup AWS Cloud9 pengguna melakukan ini. Ini mengasumsikan Anda juga tidak melampirkan kebijakan yang mencegah pengguna di grup tersebut membuat lingkungan SSH. Jika tidak, pengguna tersebut tidak dapat membuat lingkungan.

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Effect": "Deny",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*"
    }
  ]
}
```

Kebijakan terkelola pelanggan sebelumnya secara eksplisit menggantikan `"Effect": "Allow"` `"Action": "cloud9:CreateEnvironmentEC2"` on `"Resource": "*"` dalam kebijakan `AWSCloud9User` terkelola yang sudah dilampirkan ke grup pengguna. AWS Cloud9

## Izinkan pengguna dalam grup untuk membuat EC2 lingkungan hanya dengan jenis EC2 instans Amazon tertentu

Kebijakan terkelola pelanggan berikut, ketika dilampirkan ke grup AWS Cloud9 pengguna, memungkinkan pengguna dalam grup pengguna untuk membuat EC2 lingkungan yang hanya menggunakan tipe instance yang dimulai dengan `t2` dalam sebuah Akun AWS. Kebijakan ini mengasumsikan bahwa Anda juga tidak melampirkan kebijakan yang mencegah pengguna di grup tersebut membuat EC2 lingkungan. Jika tidak, pengguna tersebut tidak dapat membuat EC2 lingkungan.

Anda dapat mengganti `"t2.*"` dalam kebijakan berikut dengan kelas instans yang berbeda (misalnya, `"m4.*"`). Atau, Anda dapat membatasinya ke beberapa kelas instance atau tipe instance (misalnya, `[ "t2.*", "m4.*" ]` atau`[ "t2.micro", "m4.large" ]`).

Untuk grup AWS Cloud9 pengguna, lepaskan kebijakan `AWSCloud9User` terkelola dari grup. Kemudian, tambahkan kebijakan yang dikelola pelanggan berikut sebagai gantinya. Jika Anda tidak melepaskan kebijakan `AWSCloud9User` terkelola, kebijakan yang dikelola pelanggan berikut tidak akan berpengaruh.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:CreateEnvironmentSSH",
        "cloud9:ValidateEnvironmentName",
        "cloud9:GetUserPublicKey",
        "cloud9:UpdateUserSettings",
```

```
        "cloud9:GetUserSettings",
        "iam:GetUser",
        "iam:ListUsers",
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "cloud9:InstanceType": "t2.*"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:DescribeEnvironmentMemberships"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "Null": {
          "cloud9:UserArn": "true",
          "cloud9:EnvironmentId": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "cloud9.amazonaws.com"
        }
      }
```

```
    }
  ]
}
```

Kebijakan terkelola pelanggan sebelumnya juga memungkinkan pengguna tersebut untuk membuat lingkungan SSH. Untuk mencegah pengguna tersebut agar tidak membuat lingkungan SSH sama sekali, hapus `"cloud9:CreateEnvironmentSSH",` dari kebijakan yang dikelola pelanggan sebelumnya.

### Izinkan pengguna dalam grup untuk membuat hanya satu EC2 lingkungan di masing-masing Wilayah AWS

Kebijakan yang dikelola pelanggan berikut, ketika dilampirkan ke grup AWS Cloud9 pengguna, memungkinkan setiap pengguna tersebut untuk membuat maksimal satu EC2 lingkungan di setiap lingkungan Wilayah AWS yang AWS Cloud9 tersedia. Ini dilakukan dengan membatasi nama lingkungan ke satu nama tertentu di dalamnya. Wilayah AWS Dalam contoh ini, lingkungan dibatasi untuk`my-demo-environment`.

> **Note**
>
> AWS Cloud9 tidak mengaktifkan pembatasan lingkungan ke spesifik Wilayah AWS agar tidak dibuat. AWS Cloud9 juga tidak memungkinkan membatasi jumlah keseluruhan lingkungan yang dapat dibuat. Satu-satunya pengecualian adalah batas layanan yang diterbitkan.

Untuk grup AWS Cloud9 pengguna, lepaskan kebijakan `AWSCloud9User` terkelola dari grup, lalu tambahkan kebijakan terkelola pelanggan berikut sebagai gantinya. Jika Anda tidak melepaskan kebijakan `AWSCloud9User` terkelola, kebijakan yang dikelola pelanggan berikut tidak akan berpengaruh.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:CreateEnvironmentSSH",
        "cloud9:ValidateEnvironmentName",
        "cloud9:GetUserPublicKey",
        "cloud9:UpdateUserSettings",
```

```
        "cloud9:GetUserSettings",
        "iam:GetUser",
        "iam:ListUsers",
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:CreateEnvironmentEC2"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloud9:EnvironmentName": "my-demo-environment"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:DescribeEnvironmentMemberships"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "Null": {
          "cloud9:UserArn": "true",
          "cloud9:EnvironmentId": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "cloud9.amazonaws.com"
```

```
        }
      }
    }
  ]
}
```

Kebijakan terkelola pelanggan sebelumnya memungkinkan pengguna tersebut untuk membuat lingkungan SSH. Untuk mencegah pengguna tersebut agar tidak membuat lingkungan SSH sama sekali, hapus `"cloud9:CreateEnvironmentSSH",` dari kebijakan yang dikelola pelanggan sebelumnya.

Untuk contoh lainnya, lihat Contoh kebijakan yang dikelola pelanggan.

# Mulai menggunakan AWS Cloud9

Gunakan tutorial ini untuk memulai AWS Cloud9. Anda dapat menggunakan AWS Cloud9 konsol atau AWS Command Line Interface (AWS CLI) untuk menggunakan AWS Cloud9 IDE. Dalam tutorial ini, Anda akan belajar cara mengatur lingkungan AWS Cloud9 pengembangan, dan kemudian menggunakan AWS Cloud9 IDE untuk kode, menjalankan, dan men-debug aplikasi pertama Anda. Untuk informasi lebih lanjut tentang AWS Cloud9, lihat Apa itu AWS Cloud9.

Untuk mempelajari lebih lanjut tentang AWS Cloud9 IDE, lihatTur AWS Cloud9 IDE.

Tutorial ini membutuhkan waktu sekitar satu jam untuk menyelesaikannya.

> **Warning**
>
> Menyelesaikan tutorial ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk Amazon EC2. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

## Prasyarat

Untuk berhasil menyelesaikan tutorial ini, Anda harus terlebih dahulu menyelesaikan langkah-langkah di Menyiapkan AWS Cloud9.

## Langkah 1: Buat lingkungan

Pada langkah ini, Anda dapat menggunakan AWS Cloud9 konsol atau AWS CLI untuk membuat lingkungan AWS Cloud9 pengembangan.

> **Note**
>
> Jika Anda sudah menciptakan lingkungan yang ingin Anda gunakan untuk tutorial ini, buka lingkungan itu dan lompat ke depanLangkah 2: Tur Basic IDE.

Di AWS Cloud9, lingkungan pengembangan, atau lingkungan, adalah suatu tempat di mana Anda menyimpan file proyek pengembangan Anda dan menjalankan alat untuk mengembangkan aplikasi

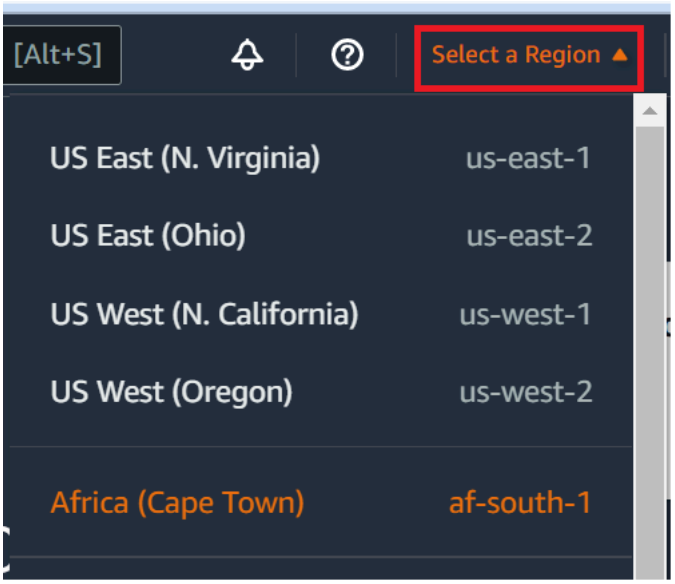
Anda. Dalam tutorial ini, Anda membuat EC2 lingkungan, dan bekerja dengan file dan alat di lingkungan itu.

Create an EC2 Environment with the console

1. Masuk ke AWS Cloud9 konsol:

   - Jika Anda satu-satunya yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, tanyakan Akun AWS administrator Anda untuk instruksi masuk.
   - Jika Anda adalah seorang siswa di kelas, mintalah petunjuk masuk kepada pengajar Anda.

2. Setelah Anda masuk ke AWS Cloud9 konsol, di bilah navigasi atas pilih Wilayah AWS untuk membuat lingkungan. Untuk daftar yang tersedia Wilayah AWS, lihat AWS Cloud9di Referensi Umum AWS.




3. Memilih tombol Menciptakan lingkungan yang besar di salah satu lokasi yang ditampilkan.

   Jika Anda belum memiliki AWS Cloud9 lingkungan, tombol ditampilkan di halaman selamat datang.

Jika Anda sudah memiliki AWS Cloud9 lingkungan, tombol ditampilkan sebagai berikut.

4. Pada halaman Buat lingkungan, untuk Nama, masukkan nama untuk lingkungan Anda.
5. Untuk Deskripsi, memasukkan sesuatu tentang lingkungan Anda. Untuk tutorial ini, gunakan `This environment is for the AWS Cloud9 tutorial.`
6. Untuk jenis Lingkungan, pilih  EC2 Instans baru untuk membuat EC2 lingkungan Amazon:

   - EC2 Instans baru — Meluncurkan EC2 instans Amazon baru yang AWS Cloud9 dapat terhubung langsung melalui SSH. Anda dapat menggunakan Systems Manager untuk berinteraksi dengan EC2 instans Amazon baru, untuk informasi selengkapnya, lihatMengakses instance no-ingress EC2 dengan AWS Systems Manager.

   - Komputasi yang ada — Meluncurkan EC2 instans Amazon yang sudah ada yang memerlukan detail login SSH yang EC2 instans Amazon harus memiliki aturan grup keamanan masuk.
     - Jika Anda memilih opsi Komputasi yang ada, peran layanan akan dibuat secara otomatis. Anda dapat melihat nama peran layanan dalam catatan di bagian bawah layar pengaturan.

> **Note**
>
> Shutdown otomatis tidak akan tersedia untuk AWS Cloud9 lingkungan yang dibuat menggunakan EC2 instans Amazon menggunakan komputasi yang ada.

> **Warning**
>
> Membuat EC2 instans Amazon untuk lingkungan Anda dapat mengakibatkan kemungkinan biaya Akun AWS untuk Amazon Anda EC2. Tidak ada biaya tambahan untuk menggunakan Systems Manager untuk mengelola koneksi ke EC2 instans Anda.

7. Pada panel EC2 instans Baru untuk tipe Instance, pertahankan pilihan default. Opsi ini mungkin memiliki lebih sedikit RAM dan lebih sedikit vCPUs. Namun, jumlah memori ini cukup untuk tutorial ini.

> **Warning**
>
> Memilih jenis instans dengan lebih banyak RAM dan v CPUs dapat mengakibatkan biaya tambahan Akun AWS untuk Amazon Anda EC2.

8. Untuk Platform, pilih jenis EC2 instans Amazon yang Anda inginkan: Amazon Linux 2023, Amazon Linux 2 atau Ubuntu 22.04 LTS. AWS Cloud9 membuat instance dan kemudian menghubungkan lingkungan ke sana.

> **Important**
>
> Kami menyarankan Anda memilih opsi Amazon Linux 2023 untuk EC2 lingkungan Anda. Selain menyediakan lingkungan runtime yang aman, stabil, dan berkinerja tinggi, Amazon Linux 2023 AMI menyertakan dukungan jangka panjang hingga 2024. Untuk informasi lebih lanjut, lihat halaman AL2 023.

9. Pilih periode waktu untuk Timeout. Opsi ini menentukan berapa lama AWS Cloud9 tidak aktif sebelum hibernasi otomatis. Ketika semua instance browser web yang terhubung ke IDE untuk lingkungan ditutup, AWS Cloud9 menunggu jumlah waktu yang ditentukan dan kemudian mematikan EC2 instance Amazon untuk lingkungan.

> **Warning**
>
> Memilih jangka waktu yang lebih lama dapat menghasilkan lebih banyak biaya untuk Anda Akun AWS.

10. Pada panel Pengaturan jaringan, pilih bagaimana lingkungan Anda diakses dari dua opsi berikut:

    - AWS System Manager (SSM) — Metode ini mengakses lingkungan menggunakan SSM tanpa membuka port masuk.

    - Secure Shell (SSH) — Metode ini mengakses lingkungan menggunakan SSH dan membutuhkan port inbound terbuka.

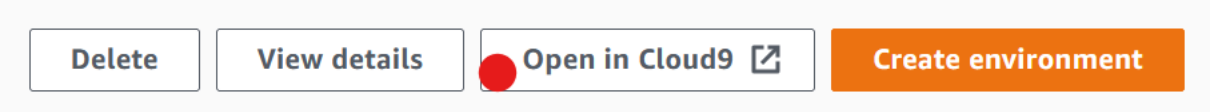
11. Pilih Pengaturan VPC untuk menampilkan Amazon Virtual Private Cloud dan Subnet untuk lingkungan Anda. AWS Cloud9 menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk berkomunikasi dengan instans Amazon EC2 yang baru dibuat. Untuk tutorial ini, kami rekomendasikan agar Anda tidak mengubah pengaturan default yang dipilih sebelumnya. Dengan pengaturan default, AWS Cloud9 upaya untuk secara otomatis menggunakan VPC default dengan subnet tunggal di yang sama Akun AWS dan Wilayah sebagai lingkungan baru.

    Anda dapat menemukan informasi selengkapnya tentang pilihan VPC Amazon di Create an EC2 Environment with the Console, dan di dalamnya. Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9

12. Tambahkan hingga 50 tag dengan menyediakan Kunci dan Nilai untuk setiap tag. Lakukan dengan memilih Tambahkan tag baru. Tag dilampirkan ke AWS Cloud9 lingkungan sebagai tag sumber daya, dan disebarkan ke sumber daya dasar berikut: AWS CloudFormation tumpukan, EC2 instans Amazon, dan grup EC2 keamanan Amazon. Untuk mempelajari lebih lanjut tentang tag, lihat Mengontrol Akses Menggunakan Tag AWS Sumber Daya di Panduan Pengguna IAM dan informasi lanjutan dalam panduan ini.

    > **Warning**
    >
    > Jika Anda memperbarui tag ini setelah Anda membuatnya, perubahan tidak disebarkan ke sumber daya yang mendasarinya. Untuk informasi selengkapnya, lihat Menyebarkan pembaruan tanda ke sumber daya yang mendasari dalam informasi lanjutan tentang tanda.

13. Pilih Buat untuk membuat lingkungan Anda, dan kemudian Anda diarahkan ke halaman beranda. Jika akun berhasil dibuat, bilah flash hijau muncul di bagian atas AWS Cloud9 konsol. Anda dapat memilih lingkungan baru dan memilih Buka di Cloud9 untuk meluncurkan IDE.




Jika akun gagal dibuat, bilah flash merah muncul di bagian atas AWS Cloud9 konsol. Akun Anda mungkin gagal dibuat karena masalah dengan browser web Anda, izin AWS akses Anda, instans, atau jaringan terkait. Anda dapat menemukan informasi tentang kemungkinan perbaikan di bagian AWS Cloud9 Pemecahan Masalah.

> **Note**
>
> AWS Cloud9 mendukung keduanya IMDSv1 dan IMDSv2. Kami merekomendasikan untuk mengadopsi IMDSv2 karena memberikan tingkat keamanan yang ditingkatkan dibandingkan IMDSv1 dengan. Untuk informasi lebih lanjut tentang manfaat IMDSv2, lihat Blog AWS Keamanan. Untuk informasi tentang cara beralih ke IMDSv2 dari IMDSv1, lihat Transisi ke menggunakan Layanan Metadata Instans Versi 2 di Panduan EC2 Pengguna Amazon untuk Instans Linux.

> **Note**
>
> Jika lingkungan Anda menggunakan proxy untuk mengakses internet, Anda harus memberikan detail proxy AWS Cloud9 agar dapat menginstal dependensi. Untuk informasi selengkapnya, lihat Gagal menginstal dependensi.

Create an EC2 environment with the AWS CLI

1. Instal dan konfigurasikan AWS CLI, jika Anda belum melakukannya. Untuk melakukannya, lihat petunjuk berikut di AWS Command Line Interface Panduan Pengguna:

   - Menginstal Antarmuka Baris AWS Perintah
   - Konfigurasi cepat

   Anda dapat mengonfigurasi kredensi AWS CLI menggunakan untuk salah satu dari berikut ini:

   - Pengguna IAM yang Anda buat di Menyiapkan Tim untuk AWS Cloud9.

- Administrator IAM di AWS akun Anda, jika Anda akan bekerja secara teratur dengan AWS Cloud9 sumber daya untuk beberapa pengguna di seluruh akun. Jika Anda tidak dapat mengonfigurasi AWS CLI sebagai administrator IAM, periksa dengan administrator AWS akun Anda. Untuk informasi lebih lanjut, lihat Membuat Grup dan Pengguna Admin IAM Pertama Anda dalam Panduan Pengguna IAM.
- Pengguna root AWS akun, tetapi hanya jika Anda akan selalu menjadi satu-satunya yang menggunakan AWS akun Anda sendiri, dan Anda tidak perlu berbagi lingkungan Anda dengan orang lain. Kami tidak merekomendasikan opsi ini karena ini bukan praktik terbaik AWS keamanan. Untuk informasi selengkapnya, lihat Membuat, Menonaktifkan, dan Menghapus Kunci Akses untuk AWS Akun Anda di. Referensi Umum Amazon Web Services
- Untuk opsi lain, lihat administrator AWS akun atau instruktur kelas Anda.

2. Dalam AWS Cloud9 perintah berikut, berikan nilai untuk `--region` dan`--subnet-id`. Kemudian jalankan perintah dan buat catatan nilai `"environmentId"` untuk pembersihan kemudian.

```
aws cloud9 create-environment-ec2 --name my-demo-environment --description "This
 environment is for the AWS Cloud9 tutorial." --instance-type t2.micro --image-
id resolve:ssm:/aws/service/cloud9/amis/amazonlinux-2-x86_64 --region MY-REGION
 --connection-type CONNECT_SSM --subnet-id subnet-12a3456b
```

Dalam perintah sebelumnya:

- `--name` mewakili nama lingkungan. Dalam tutorial ini, kita menggunakan nama `my-demo-environment`.
- `--description` mewakili deskripsi opsional untuk lingkungan.
- `--instance-type`mewakili jenis EC2 instans Amazon yang AWS Cloud9 akan diluncurkan dan terhubung ke lingkungan baru. Contoh ini menentukan`t2.micro`, yang memiliki RAM dan v relatif rendah CPUs dan cukup untuk tutorial ini. Menentukan jenis instans dengan lebih banyak RAM dan v CPUs dapat mengakibatkan biaya tambahan ke AWS akun Anda untuk Amazon EC2. Untuk daftar jenis instans yang tersedia, lihat wizard buat lingkungan di AWS Cloud9 konsol.
- `--image-id`menentukan pengenal untuk Amazon Machine Image (AMI) yang digunakan untuk membuat instance. EC2 Untuk memilih AMI untuk instance, Anda harus menentukan alias AMI yang valid atau jalur AWS Systems Manager (SSM) yang valid. Dalam contoh di atas, jalur SSM untuk Amazon Linux 2 AMI ditentukan.

Untuk informasi selengkapnya, lihat create-environment-ec2 di Referensi AWS CLI Perintah.

- `--region`mewakili ID AWS Wilayah AWS Cloud9 untuk menciptakan lingkungan di. Untuk daftar AWS Wilayah yang tersedia, lihat AWS Cloud9di Referensi Umum Amazon Web Services.
- `--connection-type CONNECT_SSM`menentukan yang AWS Cloud9 terhubung ke EC2 instans Amazon melalui Systems Manager. Opsi ini memastikan tidak ada lalu lintas masuk ke instans yang diizinkan. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager.

  > **Note**
  >
  > Saat menggunakan opsi ini, Anda perlu membuat peran `AWSCloud9SSMAccessRole` layanan dan `AWSCloud9SSMInstanceProfile` jika belum dibuat. Untuk informasi selengkapnya, lihat Mengelola profil instans untuk Systems Manager dengan AWS CLI.

- `--subnet-id`mewakili subnet yang AWS Cloud9 ingin Anda gunakan. Ganti `subnet-12a3456b` dengan ID subnet Amazon Virtual Private Cloud (VPC), yang harus kompatibel dengan AWS Cloud9. Untuk informasi selengkapnya, lihat Buat VPC plus sumber daya VPC lainnya di Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9.
- AWS Cloud9 mematikan EC2 instance Amazon untuk lingkungan setelah semua instance browser web yang terhubung ke IDE untuk lingkungan telah ditutup. Untuk mengkonfigurasi periode waktu ini, tambahkan `--automatic-stop-time-minutes` dan jumlah menit. Jangka waktu yang lebih singkat dapat mengakibatkan lebih sedikit biaya ke AWS akun Anda. Demikian juga, waktu yang lebih lama mungkin mengakibatkan biaya lebih.
- Secara default, entitas yang memanggil perintah ini memiliki lingkungan. Untuk mengubah ini, menambahkan `--owner-id` dan Amazon Resource Name (ARN) dari entitas yang memiliki.

3. Setelah Anda berhasil menjalankan perintah ini, buka AWS Cloud9 IDE untuk lingkungan yang baru dibuat. Untuk melakukannya, lihat Membuka lingkungan di AWS Cloud9. Kemudian kembali ke topik ini dan lanjutkan dengan Langkah 2: Tur Basic IDE mempelajari cara menggunakan AWS Cloud9 IDE untuk bekerja dengan lingkungan baru Anda.

Jika Anda mencoba membuka lingkungan, tetapi AWS Cloud9 tidak menampilkan IDE setelah setidaknya lima menit, mungkin ada masalah dengan browser web Anda, izin AWS akses Anda, instance, atau VPC terkait. Untuk kemungkinan perbaikan, lihatTidak dapat membuka lingkungan.

# Langkah 2: Tur Basic IDE

Bagian tutorial ini memperkenalkan beberapa cara Anda dapat menggunakan AWS Cloud9 IDE untuk membuat dan menguji aplikasi.

- Anda dapat menggunakan jendela editor untuk membuat dan mengedit kode.
- Anda dapat menggunakan jendela terminal atau jendela Jalankan Konfigurasi untuk menjalankan kode Anda tanpa melakukan debugging.
- Anda dapat menggunakan jendela Debugger untuk men-debug kode Anda.

Lakukan ketiga tugas ini menggunakan JavaScript dan mesin Node.js. Untuk petunjuk tentang penggunaan bahasa pemrograman lain, lihatTutorial untuk AWS Cloud9.

## Siapkan lingkungan Anda

Sebagian besar alat yang Anda butuhkan untuk menjalankan dan men-debug JavaScript kode sudah diinstal untuk Anda. Namun, Anda memerlukan satu paket Node.js tambahan untuk tutorial ini. Instal itu seperti berikut.

1. Pada bilah menu di bagian atas AWS Cloud9 IDE, pilih Jendela, Terminal Baru atau gunakan jendela terminal yang ada.
2. Di jendela terminal, yang merupakan salah satu tab di bagian bawah IDE, masukkan yang berikut ini.

   ```
   npm install readline-sync
   ```

   Verifikasi bahwa hasilnya mirip dengan yang berikut ini. Jika `npm WARN` pesan juga ditampilkan, Anda dapat mengabaikannya.

   ```
   + readline-sync@1.4.10
   ```

```
added 1 package from 1 contributor and audited 5 packages in 0.565s
found 0 vulnerabilities
```

## Tulis kode

Mulailah dengan tulis beberapa kode.

1. Pada bilah menu, memilih File, File Baru.
2. Tambahkan berikut ini JavaScript ke file baru.

```
var readline = require('readline-sync');
var i = 10;
var input;

console.log("Hello Cloud9!");
console.log("i is " + i);

do {
    input = readline.question("Enter a number (or 'q' to quit): ");
    if (input === 'q') {
        console.log('OK, exiting.')
    }
    else{
        i += Number(input);
        console.log("i is now " + i);
    }
} while (input != 'q');

console.log("Goodbye!");
```

3. Memilih File, Simpan, dan kemudian simpan file sebagai `hello-cloud9.js`.

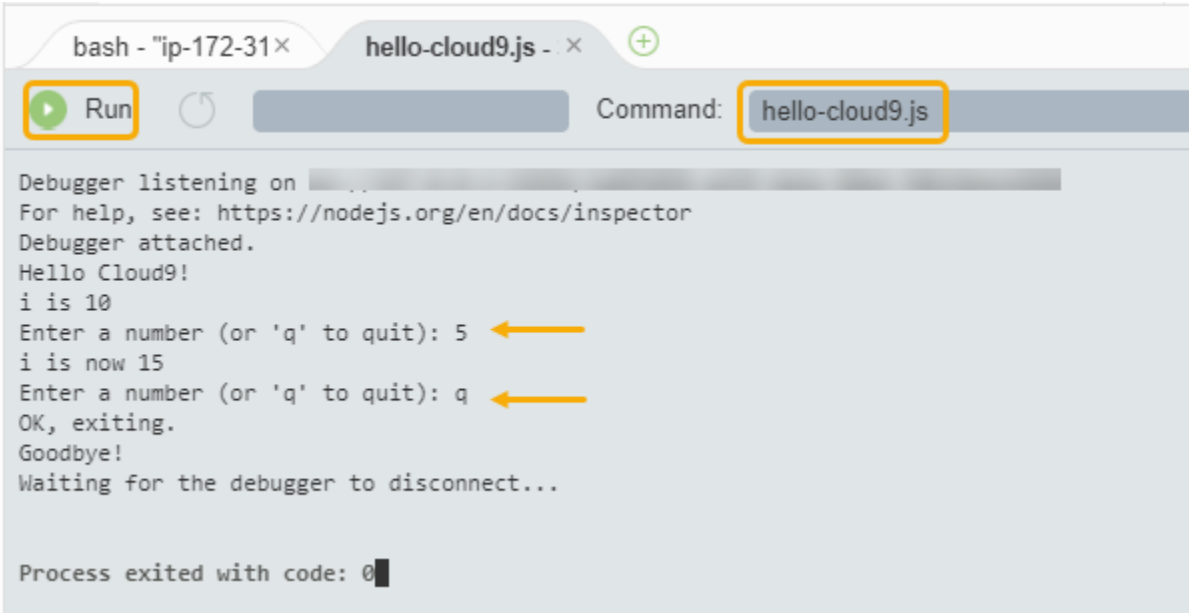
## Jalankan kode Anda

Selanjutnya, Anda dapat menjalankan kode Anda.

Tergantung pada bahasa pemrograman yang Anda gunakan, mungkin ada beberapa cara yang dapat Anda jalankan kode. Tutorial ini menggunakan JavaScript, yang dapat Anda jalankan menggunakan jendela terminal atau jendela Run Configuration.

Untuk menjalankan kode menggunakan jendela Jalankan Konfigurasi

1. Pada bilah menu, memilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.
2. Di jendela Jalankan Konfigurasi yang baru (salah satu tab di bagian bawah IDE), memasukkan `hello-cloud9.js` di bidang Perintah, dan kemudian memilih Jalankan.
3. Pastikan bahwa prompt Run Configuration aktif, dan kemudian berinteraksi dengan aplikasi dengan memasukkan nomor pada prompt.
4. Melihat output dari kode Anda di jendela Jalankan Konfigurasi. Ini mirip dengan yang berikut ini.



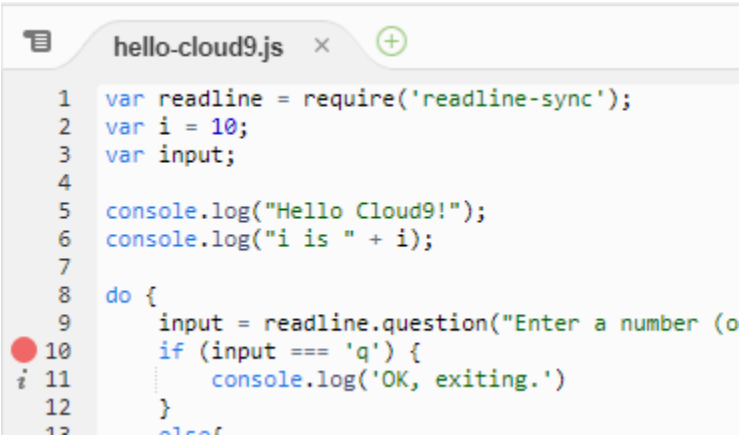

Untuk menjalankan kode menggunakan jendela terminal

1. Buka jendela terminal yang Anda gunakan sebelumnya (atau buka yang baru).
2. Di jendela terminal, memasukkan `ls` pada prompt terminal, dan verifikasi bahwa file kode Anda ada dalam daftar file.
3. Memasukkan `node hello-cloud9.js` pada prompt untuk mulai aplikasi.
4. Berinteraksi dengan aplikasi dengan memasukkan nomor pada prompt.
5. Melihat output dari kode Anda di jendela terminal. Ini mirip dengan yang berikut ini.

## Debug kode Anda

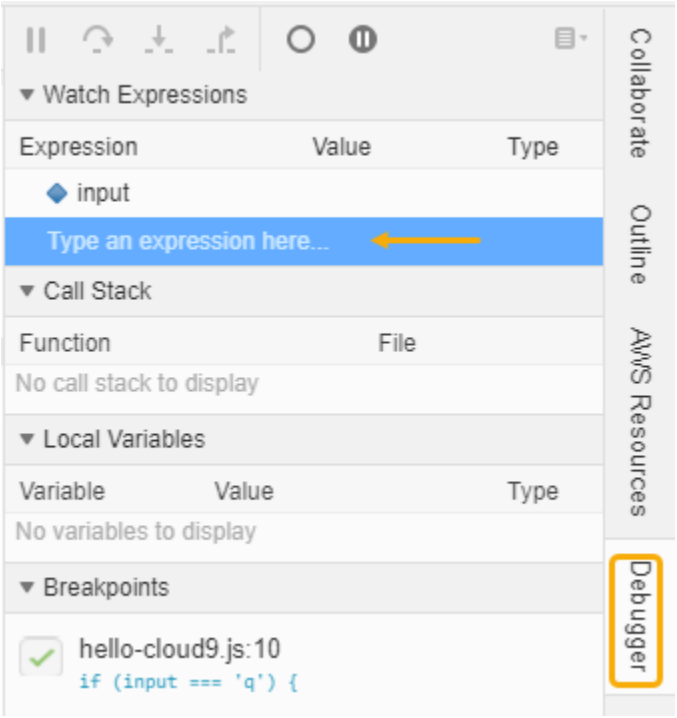
Akhirnya, Anda dapat men-debug kode Anda dengan menggunakan jendela Debugger.

1. Menambahkan breakpoint ke kode Anda di baris 10 (`if (input === 'q')`) dengan memilih margin di sebelah baris 10. Sebuah lingkaran merah ditampilkan di sebelah nomor baris, sebagai berikut.




2. Buka jendela Debugger dengan memilih tombol Debugger di sisi kanan IDE. Atau, memilih Jendela, Debugger pada menu.

   Lalu, pasang jam pada variabel `input` dengan memilih Ketik ekspresi di sini di bagian Watch Expressions dari jendela Debugger.

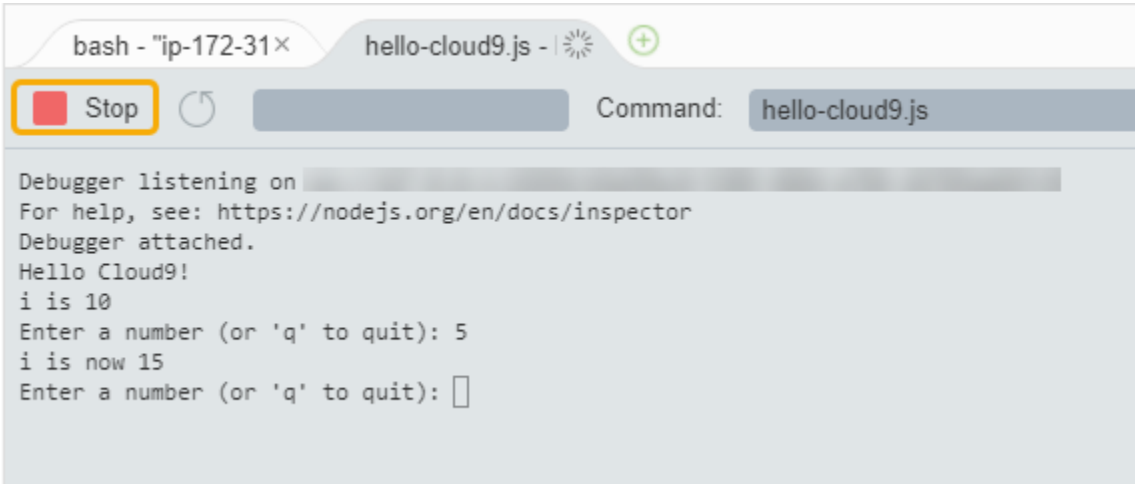



3. Pergi ke jendela Jalankan Konfigurasi yang Anda gunakan sebelumnya untuk menjalankan kode. Memilih Jalankan.

   Sebagai alternatif, Anda dapat membuka jendela Run Configuration baru dan mulai menjalankan kode. Lakukan dengan memilih Run, Run With, Node.js dari menu bar.

4. Memasukkan nomor di prompt Jalankan Konfigurasi dan melihat bahwa kode berhenti di baris 10. Jendela Debugger menunjukkan nilai yang Anda masukkan di Watch Expressions.

5. Di jendela Debugger, pilih Lanjutkan. Ini adalah ikon panah biru yang disorot di tangkapan layar sebelumnya.
6. Pilih Berhenti di jendela Jalankan Konfigurasi untuk menghentikan debugger.




## Langkah 3: Bersihkan

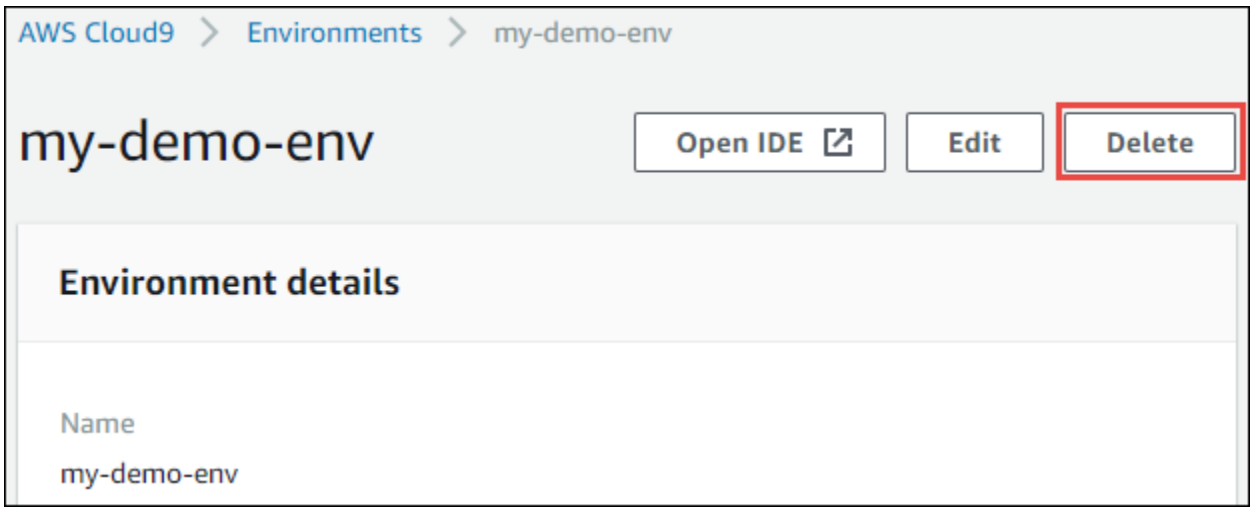
Untuk mencegah tagihan berkelanjutan ke Anda Akun AWS yang terkait dengan tutorial ini, hapus lingkungan.

> **Warning**
>
> Anda tidak dapat memulihkan lingkungan Anda setelah Anda menghapusnya.

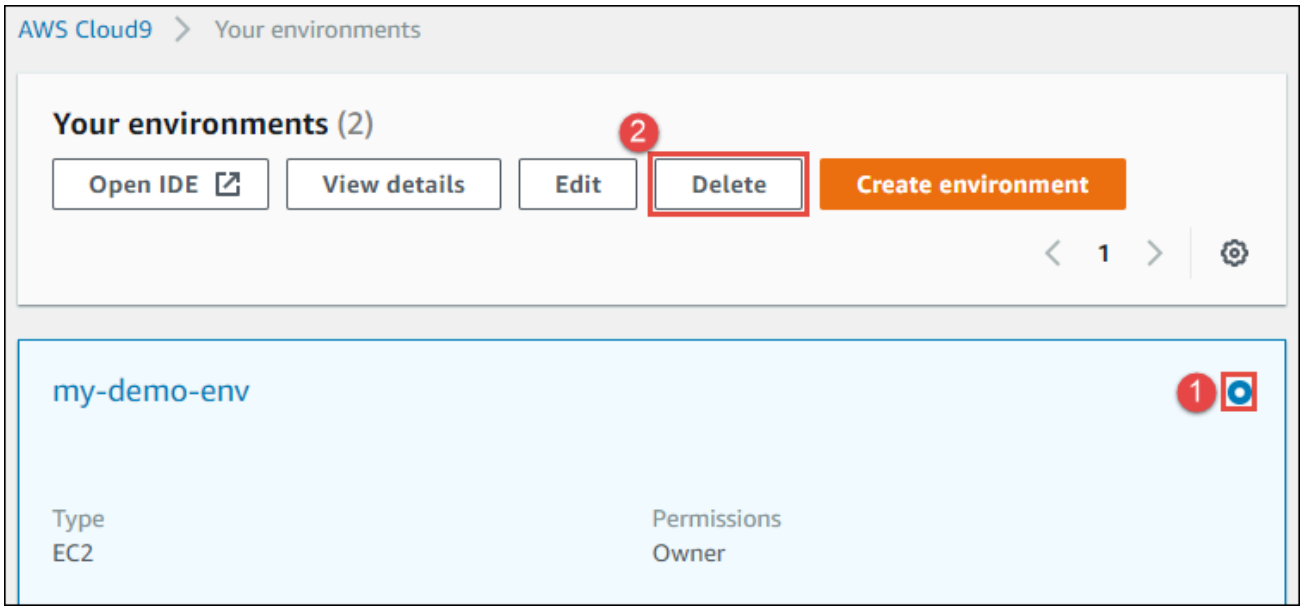
Delete the Environment by using the AWS Cloud9 console

1. Untuk membuka dasbor, pada bilah menu di IDE, memilih AWS Cloud9, Pergi ke Dasbor Anda.
2. Lakukan salah satu hal berikut ini:

   - Pilih judul di dalam my-demo-environmentkartu, lalu pilih Hapus.




- Pilih my-demo-environmentkartu, lalu pilih Hapus.




3. Di kotak dialog Hapus, memasukkan `Delete`, lalu memilih Hapus. Operasi hapus membutuhkan waktu beberapa menit.

> **Note**
>
> Jika Anda mengikuti tutorial ini dengan tepat, maka lingkungan adalah EC2 lingkungan dan AWS Cloud9 juga mengakhiri EC2 instance Amazon yang terhubung ke lingkungan itu.
>
> Namun, jika Anda menggunakan lingkungan SSH alih-alih mengikuti tutorial, dan lingkungan tersebut terhubung ke EC2 instance Amazon, AWS Cloud9 tidak akan menghentikan instance itu. Jika Anda tidak menghentikan instans itu nanti, Anda Akun

AWS mungkin terus memiliki biaya berkelanjutan untuk Amazon EC2 yang terkait dengan instance itu.

Delete the Environment with the AWS CLI

1. Jalankan AWS Cloud9 `delete-environment` perintah, tentukan ID lingkungan yang akan dihapus.

   ```
   aws cloud9 delete-environment --region MY-REGION --environment-id
    12a34567b8cd9012345ef67abcd890e1
   ```

   Pada perintah sebelumnya, ganti `MY-REGION` dengan AWS Wilayah tempat lingkungan dibuat dan `12a34567b8cd9012345ef67abcd890e1` dengan ID lingkungan yang akan dihapus.

   Jika Anda tidak menyimpan ID saat membuat lingkungan, ID dapat ditemukan dengan menggunakan AWS Cloud9 konsol. Pilih nama lingkungan di konsol tersebut, lalu temukan bagian terakhir dari Lingkungan ARN.
2. Jika Anda membuat VPC Amazon untuk tutorial ini dan Anda tidak lagi membutuhkannya, hapus VPC menggunakan konsol Amazon VPC di/vpc. https://console.aws.amazon.com

## Informasi terkait

Berikut ini adalah informasi tambahan untuk Memulai dengan AWS Cloud9 Konsol.

- Saat Anda membuat EC2 lingkungan, lingkungan tidak berisi kode sampel apa pun secara default. Untuk membuat lingkungan dengan kode sampel, lihat topik berikut:
  - Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9
- Saat lingkungan AWS Cloud9 pengembangan sedang dibuat, Anda diarahkan AWS Cloud9 untuk membuat EC2 instance Amazon. AWS Cloud9 membuat instance dan kemudian menghubungkan lingkungan ke sana. Sebagai alternatif, Anda dapat menggunakan instance komputasi awan yang ada atau server Anda sendiri, yang disebut lingkungan SSH. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.

Langkah opsional selanjutnya

Jelajahi salah satu atau semua topik berikut untuk terus membiasakan diri AWS Cloud9.

| Tugas | Lihat topik ini |
| --- | --- |
| Pelajari selengkapnya tentang apa yang dapat Anda lakukan dengan lingkungan. | Bekerja dengan lingkungan di AWS Cloud9 |
| Coba bahasa komputer lainnya. | Tutorial untuk AWS Cloud9 |
| Pelajari lebih lanjut tentang AWS Cloud9 IDE. | Tur AWS Cloud9 IDE di Bekerja dengan IDE |
| Undang orang lain untuk menggunakan lingkungan baru Anda secara real time dan dengan dukungan obrolan teks. | Bekerja dengan lingkungan bersama di AWS Cloud9 |
| Buat lingkungan SSH. Ini adalah lingkunga n yang menggunakan instance komputasi awan atau server yang Anda buat, bukan EC2 instans Amazon yang AWS Cloud9 dibuat untuk Anda. | Menciptakan lingkungan di AWS Cloud9 dan Persyaratan host lingkungan SSH |
| Buat, jalankan, dan debug kode dalam AWS Lambda fungsi dan aplikasi tanpa server menggunakan Toolkit. AWS | Bekerja dengan AWS Lambda fungsi menggunakan AWS Toolkit |
| Gunakan AWS Cloud9 dengan Amazon Lightsail. | Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9 |
| Gunakan AWS Cloud9 dengan AWS CodePipel ine. | Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE |
| Gunakan AWS Cloud9 dengan AWS CLI,, AWS Cloud Development Kit (AWS CDK) AWS CodeCommit,, atau Amazon DynamoDB GitHub, dan Node.js, Python, atau bahasa pemrograman lainnya. AWS CloudShell | Tutorial untuk AWS Cloud9 |
| Bekerja dengan kode untuk aplikasi robotika cerdas di AWS RoboMaker. | Berkembang dengan AWS Cloud9 di Panduan AWS RoboMaker Pengembang |

Untuk mendapatkan bantuan AWS Cloud9 dari komunitas, lihat Forum AWS Cloud9 Diskusi. (Saat Anda memasuki forum ini, AWS mungkin mengharuskan Anda untuk masuk.)

Untuk mendapatkan bantuan AWS Cloud9 langsung dari AWS, lihat opsi dukungan di halaman AWS Support.

# Bekerja dengan lingkungan di AWS Cloud9

Lingkungan pengembangan adalah tempat di AWS Cloud9 mana Anda menyimpan file proyek Anda dan di mana Anda menjalankan alat untuk mengembangkan aplikasi Anda.

AWS Cloud9 menyediakan dua jenis lingkungan pengembangan: EC2 lingkungan dan lingkungan SSH. Untuk memahami persamaan dan perbedaan utama antara lingkungan pengembangan ini, lihat EC2 lingkungan dibandingkan dengan lingkungan SSH di AWS Cloud9.

Pelajari cara bekerja dengan lingkungan AWS Cloud9 dengan membaca satu atau lebih topik ini.

Topik


- Menciptakan lingkungan di AWS Cloud9
- Mengakses instance no-ingress EC2 dengan AWS Systems Manager
- Membuka lingkungan di AWS Cloud9
- Menelepon Layanan AWS dari lingkungan di AWS Cloud9
- Mengubah pengaturan lingkungan di AWS Cloud9
- Bekerja dengan lingkungan bersama di AWS Cloud9
- Memindahkan AWS Cloud9 IDE dari volume Amazon EBS
- Menghapus lingkungan di AWS Cloud9


## Menciptakan lingkungan di AWS Cloud9

Untuk menciptakan lingkungan AWS Cloud9 pengembangan, ikuti salah satu prosedur yang disediakan berdasarkan bagaimana Anda berencana untuk menggunakannya AWS Cloud9.

Jika Anda tidak yakin apa yang harus dipilih, kami merekomendasikan Menciptakan EC2 Lingkungan.

Untuk pengaturan cepat, buat EC2 lingkungan. AWS Cloud9 secara otomatis membuat dan menyiapkan EC2 instans Amazon baru di instans Anda Akun AWS. AWS Cloud9 juga secara otomatis menghubungkan instance baru itu ke lingkungan untuk Anda.

Untuk memahami persamaan dan perbedaan utama antara lingkungan pengembangan, lihat EC2 lingkungan dibandingkan dengan lingkungan SSH di AWS Cloud9.

| Penyedia kode sumber | Penyedia host lingkungan pengembangan | Prosedur yang relevan |
| --- | --- | --- |
| Anda | AWS Cloud9 | Ciptakan EC2 lingkungan |
| Anda | Anda | Buat lingkungan SSH |
| Amazon Lightsail atau Anda | Anda (menggunakan Lightsail) | Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9 |
| Anda (menggunakan AWS CodePipeline) | AWS Cloud9 atau kamu | Membuat lingkungan EC2atau SSH, dan Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE |
| Anda (menggunakan AWS CodeCommit) | AWS Cloud9 atau kamu | AWS CodeCommit tutorial untuk AWS Cloud9 |
| Anda (menggunakan GitHub) | AWS Cloud9 atau kamu | Buat lingkungan EC2atau SSH, dan gunakan antarmuka panel Git |

Topik


- Menciptakan EC2 Lingkungan
- Membuat Lingkungan SSH


## Menciptakan EC2 Lingkungan

Dalam prosedur ini, AWS Cloud9 buat EC2 lingkungan dan EC2 instans Amazon baru, dan menghubungkan lingkungan ke instance ini. AWS Cloud9 mengelola siklus hidup instance ini, termasuk memulai, menghentikan, dan memulai ulang instance sesuai kebutuhan. Jika Anda pernah menghapus lingkungan ini, AWS Cloud9 secara otomatis mengakhiri instans ini.

Anda dapat membuat lingkungan AWS Cloud9 EC2 pengembangan di AWS Cloud9 konsol atau dengan kode.

> **Note**
>
> Menyelesaikan prosedur ini dapat mengakibatkan biaya untuk Anda Akun AWS. Ini termasuk kemungkinan biaya untuk Amazon EC2. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

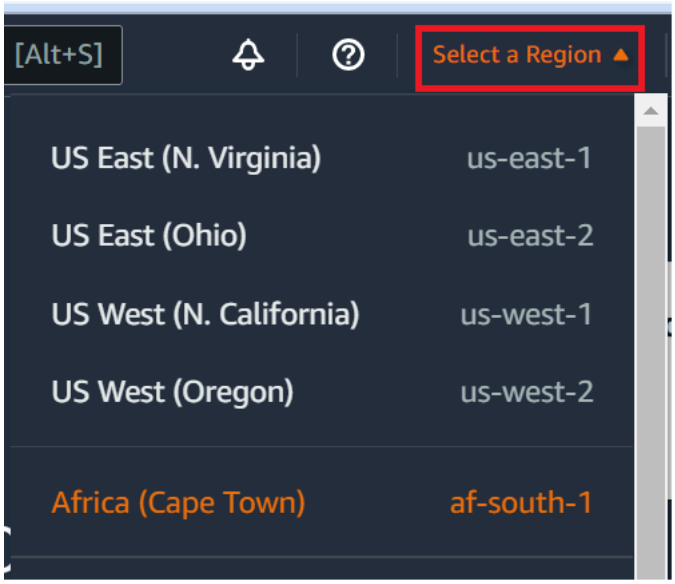
> **Warning**
>
> Masalah kompatibilitas ada dengan AWS Cloud9 dan CT kontrol AWS Control Tower proaktif. EC2.PR.8. Jika kontrol ini diaktifkan, Anda tidak dapat membuat EC2 lingkungan di AWS Cloud9. Untuk informasi selengkapnya tentang masalah ini, lihat Pemecahan Masalah AWS Cloud9.

## Prasyarat

Selesaikan langkah-langkahnya Menyiapkan AWS Cloud9 sehingga Anda dapat masuk ke AWS Cloud9 konsol dan membuat lingkungan.

## Buat EC2 lingkungan dengan konsol

1. Masuk ke AWS Cloud9 konsol:
   - Jika Anda satu-satunya yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, tanyakan Akun AWS administrator Anda untuk instruksi masuk.
   - Jika Anda adalah seorang siswa di kelas, mintalah petunjuk masuk kepada pengajar Anda.
2. Setelah Anda masuk ke AWS Cloud9 konsol, di bilah navigasi atas pilih Wilayah AWS untuk membuat lingkungan. Untuk daftar yang tersedia Wilayah AWS, lihat AWS Cloud9di Referensi Umum AWS.




3. Memilih tombol Menciptakan lingkungan yang besar di salah satu lokasi yang ditampilkan.

   Jika Anda belum memiliki AWS Cloud9 lingkungan, tombol ditampilkan di halaman selamat datang.

   Jika Anda sudah memiliki AWS Cloud9 lingkungan, tombol ditampilkan sebagai berikut.

1. Pada halaman Buat lingkungan, untuk Nama, masukkan nama untuk lingkungan Anda.
2. Untuk menambahkan deskripsi ke lingkungan Anda, masukkan di bidang Deskripsi.
3. Untuk jenis Lingkungan, pilih  EC2 Instans baru untuk membuat EC2 lingkungan Amazon:

   - EC2 Instans baru — Meluncurkan EC2 instans Amazon baru yang AWS Cloud9 dapat terhubung langsung melalui SSH. Anda dapat menggunakan Systems Manager untuk berinteraksi dengan EC2 instans Amazon baru, untuk informasi selengkapnya, lihatMengakses instance no-ingress EC2 dengan AWS Systems Manager.

- Komputasi yang ada — Meluncurkan EC2 instans Amazon yang sudah ada yang memerlukan detail login SSH yang EC2 instans Amazon harus memiliki aturan grup keamanan masuk.
  - Jika Anda memilih opsi Komputasi yang ada, peran layanan akan dibuat secara otomatis. Anda dapat melihat nama peran layanan dalam catatan di bagian bawah layar pengaturan.

> **Note**
>
> Shutdown otomatis tidak akan tersedia untuk AWS Cloud9 lingkungan yang dibuat menggunakan EC2 instans Amazon menggunakan komputasi yang ada.

> **Warning**
>
> Membuat EC2 instans Amazon untuk lingkungan Anda dapat mengakibatkan kemungkinan biaya Akun AWS untuk Amazon Anda EC2. Tidak ada biaya tambahan untuk menggunakan Systems Manager untuk mengelola koneksi ke EC2 instans Anda.

4. Untuk jenis Instance, pilih jenis instance dengan jumlah RAM dan v CPUs yang menurut Anda perlu untuk jenis tugas yang ingin Anda lakukan.

> **Warning**
>
> Memilih jenis instans dengan lebih banyak RAM dan v CPUs dapat mengakibatkan biaya tambahan Akun AWS untuk Amazon Anda EC2. Untuk informasi tentang jenis instans mana yang cocok untuk beban kerja Anda, lihat halaman Jenis EC2 Instance Amazon.

5. Untuk Platform, pilih jenis EC2 instans Amazon yang Anda inginkan: Amazon Linux 2023, Amazon Linux 2 atau Ubuntu 22.04 LTS. AWS Cloud9 membuat instance dan kemudian menghubungkan lingkungan ke sana.

> **Important**
>
> Kami menyarankan Anda memilih opsi Amazon Linux 2023 untuk EC2 lingkungan Anda. Selain menyediakan lingkungan runtime yang aman, stabil, dan berkinerja tinggi, Amazon Linux 2023 AMI menyertakan dukungan jangka panjang hingga 2024.

Untuk informasi lebih lanjut, lihat halaman AL2 023.

6. Pilih periode waktu untuk Timeout. Opsi ini menentukan berapa lama AWS Cloud9 tidak aktif sebelum hibernasi otomatis. Ketika semua instance browser web yang terhubung ke IDE untuk lingkungan ditutup, AWS Cloud9 menunggu jumlah waktu yang ditentukan dan kemudian mematikan EC2 instance Amazon untuk lingkungan.

   ⚠ Warning

   Memilih jangka waktu yang lebih lama dapat menghasilkan lebih banyak biaya untuk Anda Akun AWS.

7. Pada panel Pengaturan jaringan, pilih bagaimana lingkungan Anda diakses dari dua opsi berikut:

   - AWS Systems Manager (SSM) — Metode ini mengakses lingkungan menggunakan SSM tanpa membuka port masuk.

   - Secure Shell (SSH) — Metode ini mengakses lingkungan menggunakan SSH dan membutuhkan port inbound terbuka.

8. Pilih Pengaturan VPC untuk menampilkan Amazon Virtual Private Cloud dan Subnet untuk lingkungan Anda. AWS Cloud9 menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk berkomunikasi dengan instans Amazon EC2 yang baru dibuat. Untuk tutorial ini, kami rekomendasikan agar Anda tidak mengubah pengaturan default yang dipilih sebelumnya. Dengan pengaturan default, AWS Cloud9 mencoba untuk menggunakan VPC default dengan subnet tunggal di sama Akun AWS dan Wilayah sebagai lingkungan baru. Bergantung pada bagaimana Amazon VPC disiapkan, ikuti salah satu dari rangkaian instruksi berikut.

   Jika Anda tidak yakin apa yang harus dipilih, sebaiknya Anda melompat ke langkah berikutny a dalam prosedur ini.

   Jika Anda melewati pengaturan Jaringan (lanjutan) dan meninggalkan pengaturan default yang telah dipilih sebelumnya, AWS Cloud9 mencoba menggunakan VPC default dengan subnet tunggalnya. AWS Cloud9 memilih subnet berdasarkan tipe instance yang Anda pilih. Ini berada di AWS akun dan AWS Wilayah yang sama dengan lingkungan baru.

**Important**

Jika Anda memilih Existing compute sebagai tipe lingkungan, Anda dapat meluncurkan instance Anda ke subnet publik atau pribadi.

- Subnet publik: Lampirkan gateway internet ke subnet untuk memungkinkan agen SSM instance berkomunikasi dengan Systems Manager.
- Subnet pribadi: Buat gateway NAT untuk memungkinkan instance berkomunikasi dengan internet dan lainnya. Layanan AWS

Saat ini, Anda tidak dapat menggunakan kredensil sementara AWS terkelola untuk memungkinkan EC2 lingkungan mengakses Layanan AWS atas nama AWS entitas, seperti pengguna IAM.
Untuk informasi selengkapnya tentang mengonfigurasi subnet, lihat. Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Tidak | — | — | — | Jika tidak ada VPC ada, buatlah.<br><br>Untuk membuat VPC di lingkungan yang sama Akun AWS dan Wilayah dengan lingkungan baru, pilih Buat VPC baru, lalu ikuti petunjuk di layar. Untuk informasi selengkapnya, lihat Buat VPC plus sumber daya VPC lainnya.<br><br>Untuk membuat VPC di lingkungan yang berbeda Akun AWS dari lingkunga |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| | | | | n baru, lihat Bekerja dengan Dibagikan VPCs di Panduan Pengguna Amazon VPC. |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Ya | Ya | Ya | Ya | Lumpati ke langkah berikutnya dalam prosedur ini.<br><br>Saat Anda melewati pengatura n Jaringan (lanjutan) dan tidak mengubah pengaturan default yang telah dipilih sebelumnya, AWS Cloud9 mencoba menggunakan VPC default dengan subnet tunggalnya di akun dan Wilayah yang sama dengan lingkungan baru. |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Ya | Ya | Ya | Tidak | Jika VPC default memiliki beberapa subnet, perluas Pengaturan jaringan (lanjutan). Untuk Subnet, pilih subnet yang AWS Cloud9 ingin Anda gunakan di VPC default yang telah dipilih sebelumnya.<br><br>Jika VPC default tidak memiliki subnet, buatlah. Untuk melakukannya, pilih Buat subnet baru, lalu ikuti petunjuk di layar. Untuk informasi selengkapnya, lihat Buat |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| | | | | subnet untuk AWS Cloud9. |
| Ya | Ya | Tidak | Ya | Perluas Pengaturan jaringan. Untuk Jaringan (VPC), pilih VPC yang Anda inginkan untuk digunakan AWS Cloud9 . |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Ya | Ya | Tidak | Tidak | Perluas Pengaturan jaringan. Untuk Jaringan (VPC), pilih VPC yang Anda inginkan untuk digunakan AWS Cloud9 .<br><br>Jika VPC yang dipilih memiliki beberapa subnet, perluas Pengaturan jaringan (lanjutan). Untuk Subnet, pilih subnet yang AWS Cloud9 ingin Anda gunakan di VPC yang dipilih.<br><br>Jika VPC yang dipilih tidak memiliki subnet, buatlah. Untuk melakukan |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| | | | | nya, pilih Buat subnet baru, lalu ikuti petunjuk di layar. Untuk informasi selengkap nya, lihat Buat subnet untuk AWS Cloud9. |
| Ya | Tidak | Ya | — | AWS Cloud9 tidak dapat menggunakan VPC default di Akun AWS yang berbeda dari akun untuk lingkungan baru. Pilih opsi yang berbeda dalam daftar ini. |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Ya | Tidak | Tidak | Ya | Perluas Pengaturan jaringan. Untuk Jaringan (VPC), pilih VPC yang Anda inginkan untuk digunakan AWS Cloud9 .<br><br>**Note**<br>VPC harus berada di Wilayah yang sama dengan lingkungan baru, bahkan jika VPC berada di akun yang berbeda. |

| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| Ya | Tidak | Tidak | Tidak | Perluas Pengaturan jaringan. Untuk Jaringan (VPC), pilih VPC yang Anda inginkan untuk digunakan AWS Cloud9 .<br><br>Untuk Subnet, pilih subnet yang Anda ingin AWS Cloud9 gunakan di VPC yang dipilih.<br><br>Jika VPC yang dipilih tidak memiliki subnet, untuk membuat subnet untuk VPC di lingkungan yang berbeda Akun AWS dari lingkungan baru, lihat Bekerja dengan Bersama di |

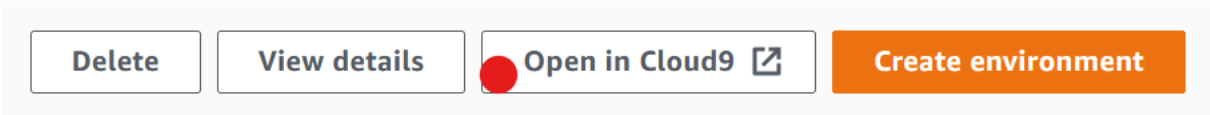
| Apakah Akun AWS memiliki akses ke VPC Amazon? | Apakah VPC itu sama Akun AWS dan Wilayah dengan lingkungan baru? | Apakah VPC itu VPC default untuk itu? Akun AWS | Apakah VPC itu berisi satu subnet? | Ikuti instruksi ini |
|---|---|---|---|---|
| | | | | Panduan Pengguna VPCs Amazon VPC.<br><br>**Note**<br>VPC dan subnet harus berada di Wilayah yang sama dengan lingkungan baru, bahkan jika VPC dan subnet berada di akun yang berbeda. |

Untuk informasi selengkapnya tentang pilihan ini, lihat Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9.

9. Tambahkan hingga 50 tag dengan menyediakan Kunci dan Nilai untuk setiap tag. Lakukan dengan memilih Tambahkan tag baru. Tag dilampirkan ke AWS Cloud9 lingkungan sebagai tag sumber daya, dan disebarkan ke sumber daya dasar berikut: AWS CloudFormation tumpukan, EC2 instans Amazon, dan grup EC2 keamanan Amazon. Untuk mempelajari lebih lanjut tentang tag, lihat Mengontrol Akses Menggunakan Tag AWS Sumber Daya di Panduan Pengguna IAM dan informasi lanjutan dalam panduan ini.

> **Warning**
>
> Jika Anda memperbarui tag ini setelah Anda membuatnya, perubahan tidak disebarkan ke sumber daya yang mendasarinya. Untuk informasi selengkapnya, lihat Menyebarkan pembaruan tanda ke sumber daya yang mendasari dalam informasi lanjutan tentang tanda.

10. Pilih Buat untuk membuat lingkungan Anda, dan kemudian Anda diarahkan ke halaman beranda. Jika akun berhasil dibuat, bilah flash hijau muncul di bagian atas AWS Cloud9 konsol. Anda dapat memilih lingkungan baru dan memilih Buka di Cloud9 untuk meluncurkan IDE.




Jika akun gagal dibuat, bilah flash merah muncul di bagian atas AWS Cloud9 konsol. Akun Anda mungkin gagal dibuat karena masalah dengan browser web Anda, izin AWS akses Anda, instans, atau jaringan terkait. Anda dapat menemukan informasi tentang kemungkinan perbaikan di bagian AWS Cloud9 Pemecahan Masalah.

> **Note**
>
> AWS Cloud9 mendukung keduanya IMDSv1 dan IMDSv2. Kami merekomendasikan untuk mengadopsi IMDSv2 karena memberikan tingkat keamanan yang ditingkatkan dibandingkan IMDSv1 dengan. Untuk informasi lebih lanjut tentang manfaat IMDSv2, lihat Blog AWS Keamanan. Untuk informasi tentang cara beralih ke IMDSv2 dari IMDSv1, lihat Transisi ke menggunakan Layanan Metadata Instans Versi 2 di Panduan EC2 Pengguna Amazon untuk Instans Linux.

> **Note**
>
> Jika lingkungan Anda menggunakan proxy untuk mengakses internet, Anda harus memberikan detail proxy AWS Cloud9 agar dapat menginstal dependensi. Untuk informasi selengkapnya, lihat Gagal menginstal dependensi.

## Membuat lingkungan dengan kode

Untuk menggunakan kode untuk membuat EC2 lingkungan AWS Cloud9, panggil operasi AWS Cloud9 create EC2 environment, sebagai berikut.

| | |
|---|---|
| AWS CLI | create-environment-ec2 |
| AWS SDK untuk C++ | CreateEnvironmentEC2Permintaan, CreateEnvironmentEC2Hasil |
| AWS SDK untuk Go | CreateEnvironmentEC2, CreateEnvironmentEC2Permintaan, CreateEnvironmentEC2WithContext |
| AWS SDK untuk Java | CreateEnvironmentEC2Permintaan, CreateEnvironment EC2 Hasil |
| AWS SDK untuk JavaScript | CreateEnvironment EC2 |
| AWS SDK untuk .NET | CreateEnvironmentEC2Permintaan, CreateEnvironmentEC2Tanggapan |
| AWS SDK untuk PHP | CreateEnvironment EC2 |
| AWS SDK for Python (Boto) | create_environment_ec2 |
| AWS SDK untuk Ruby | create_environment_ec2 |
| AWS Tools for Windows PowerShell | New-C9EnvironmentEC2 |
| AWS Cloud9 API | CreateEnvironmentEC2 |

> **Note**
>
> Jika lingkungan Anda menggunakan proxy untuk mengakses internet, Anda harus memberikan detail proxy AWS Cloud9 agar dapat menginstal dependensi. Untuk informasi selengkapnya, lihat Gagal menginstal dependensi.

# Membuat Lingkungan SSH

Anda membuat lingkungan pengembangan AWS Cloud9 SSH dengan AWS Cloud9 konsol. Anda tidak dapat membuat lingkungan SSH menggunakan CLI.

## Prasyarat

- Pastikan Anda menyelesaikan langkah-langkahnya terlebih Menyiapkan AWS Cloud9 dahulu. Dengan begitu, Anda dapat masuk ke AWS Cloud9 konsol dan membuat lingkungan.
- Identifikasi instance komputasi cloud yang ada (misalnya, EC2 instans Amazon di server Anda Akun AWS) atau server Anda sendiri yang ingin Anda sambungkan AWS Cloud9 ke lingkungan.
- Pastikan instans yang ada atau server Anda sendiri memenuhi semua Persyaratan hosting SSH. Ini termasuk memiliki versi tertentu dari Python, Node.js dan komponen lain yang sudah diinstal, mengatur izin khusus pada direktori yang Anda inginkan untuk memulai AWS Cloud9 setelah login, dan menyiapkan Amazon Virtual Private Cloud terkait.

## Membuat Lingkungan SSH

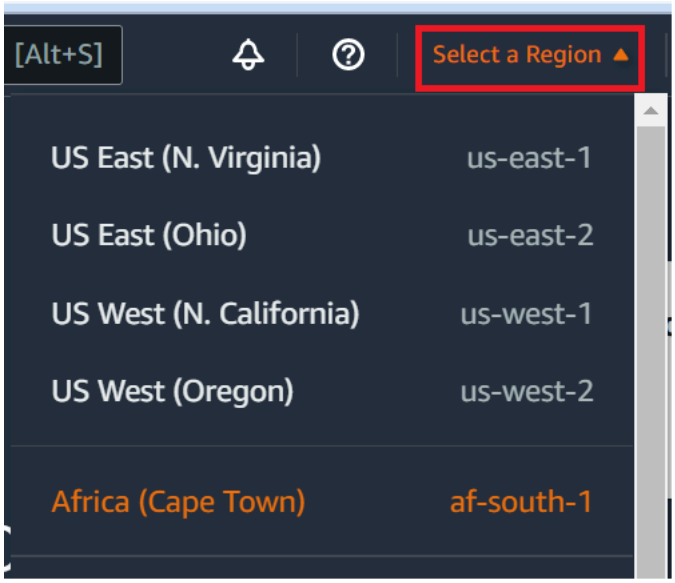
1. Pastikan Anda menyelesaikan prasyarat sebelumnya.
2. Hubungkan ke instans yang ada atau server Anda sendiri dengan menggunakan klien SSH, jika Anda belum terhubung dengannya. Ini memastikan bahwa Anda dapat menambahkan nilai kunci SSH publik yang diperlukan ke instance atau server. Ini dijelaskan kemudian dalam prosedur ini.

   > **Note**
   >
   > Untuk menyambung ke instans AWS Cloud komputasi yang ada, lihat satu atau beberapa sumber daya berikut:
   >
   > - Untuk Amazon EC2, lihat Connect to Your Linux Instance di Panduan EC2 Pengguna Amazon.

- Untuk Amazon Lightsail, lihat Menghubungkan ke instans Lightsail berbasis Linux/Unix Anda di Dokumentasi Amazon Lightsail.
- Untuk AWS Elastic Beanstalk, lihat Daftar dan Menghubungkan ke Instans Server di Panduan AWS Elastic Beanstalk Pengembang.
- Untuk AWS OpsWorks, lihat Menggunakan SSH untuk Masuk ke Instance Linux di Panduan AWS OpsWorks Pengguna.
- Untuk yang lain Layanan AWS, lihat dokumentasi untuk layanan khusus itu.

Untuk terhubung ke server Anda sendiri, gunakan SSH. SSH sudah diinstal pada sistem operasi macOS dan Linux. Untuk terhubung ke server Anda dengan menggunakan SSH pada Windows, Anda harus menginstal PuTTY.

3. Masuk ke AWS Cloud9 konsol, di https://console.aws.amazon.com/cloud9/.
4. Setelah Anda masuk ke AWS Cloud9 konsol, di bilah navigasi atas pilih Wilayah AWS untuk membuat lingkungan. Untuk daftar yang tersedia Wilayah AWS, lihat AWS Cloud9di Referensi Umum AWS.




5. Jika ini adalah pertama kalinya Anda membuat lingkungan pengembangan, halaman selamat datang akan ditampilkan. Di panel  AWS Cloud9 lingkungan baru, pilih Buat lingkungan.

   Jika sebelumnya Anda telah membuat lingkungan pengembangan, Anda juga dapat memperluas panel di sebelah kiri layar. Pilih lingkungan Anda, lalu pilih Buat lingkungan.

   Di halaman selamat datang:

Atau di halaman Lingkungan Anda:

6. Pada halaman Buat lingkungan, masukkan nama untuk lingkungan Anda.
7. Untuk Deskripsi, memasukkan sesuatu tentang lingkungan Anda. Untuk tutorial ini, gunakan `This environment is for the AWS Cloud9 tutorial.`
8. Untuk jenis Lingkungan, pilih Existing Compute dari opsi berikut:

   - EC2 Instans baru — Meluncurkan EC2 instans Amazon yang AWS Cloud9 dapat terhubung langsung melalui SSH atau SSM.

   - Komputasi yang ada - Meluncurkan EC2 instans Amazon yang ada yang memerlukan detail login SSH serta port 22 untuk dibuka. AWS Cloud9 terhubung ke instance melalui AWS Systems Manager.
     - Jika Anda memilih opsi Komputasi yang ada, peran layanan akan dibuat secara otomatis. Anda dapat melihat nama peran layanan di peran Layanan dan profil instans untuk bagian akses Systems Manager lebih jauh di bawah antarmuka. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager.

> **Warning**
>
> Membuat EC2 instance untuk lingkungan Anda dapat mengakibatkan biaya yang mungkin Akun AWS untuk Amazon Anda EC2. Tidak ada biaya tambahan untuk menggunakan Systems Manager untuk mengelola koneksi ke EC2 instans Anda.

> **Warning**
>
> AWS Cloud9 menggunakan kunci publik SSH untuk terhubung dengan aman ke server Anda. Untuk membuat koneksi aman, tambahkan kunci publik kami ke `~/.ssh/authorized_keys` file Anda dan berikan kredensi login Anda dalam langkah-langkah berikut. Pilih Salin kunci ke clipboard untuk menyalin kunci SSH, atau Lihat kunci SSH publik untuk menampilkannya.

9. Pada panel komputasi yang ada, untuk Pengguna, masukkan nama login yang Anda gunakan untuk terhubung ke instance atau server sebelumnya dalam prosedur ini. Misalnya, untuk contoh AWS Cloud komputasi, mungkin`ec2-user`,`ubuntu`, atau`root`.

> **Note**
>
> Kami menyarankan agar nama login dikaitkan dengan izin administratif atau pengguna administrator pada instance atau server. Lebih khusus lagi, kami menyarankan agar nama login ini memiliki instalasi Node.js pada instance atau server. Untuk memeriksa ini, dari terminal instance atau server Anda, jalankan perintah **`ls -l $(which node)`**(atau **`ls -l $(nvm which node)`**jika Anda menggunakan`nvm`). Perintah ini menampilkan nama pemilik instalasi Node.js. Ini juga menampilkan izin instalasi, nama grup, dan lokasi.

10. Untuk Host, masukkan alamat IP publik (lebih disukai) atau nama host dari instance atau server.
11. Untuk Port, masukkan port yang AWS Cloud9 ingin Anda gunakan untuk mencoba terhubung ke instance atau server. Atau, pertahankan port default.
12. Pilih Detail tambahan - opsional untuk menampilkan jalur lingkungan, jalur ke biner node.js dan informasi host lompat SSH.
13. Untuk jalur Lingkungan, masukkan jalur ke direktori pada instance atau server yang AWS Cloud9 ingin Anda mulai. Anda mengidentifikasi ini sebelumnya di prasyarat untuk prosedur ini. Jika Anda membiarkannya kosong, AWS Cloud9 menggunakan direktori tempat instans atau server Anda biasanya memulai setelah login. Ini biasanya merupakan direktori beranda atau default.
14. Untuk jalur biner Path to Node.js, masukkan informasi jalur untuk menentukan jalur ke biner Node.js pada instance atau server. Untuk mendapatkan jalur, Anda dapat menjalankan perintah **`which node`**(atau **`nvm which node`**jika Anda menggunakan`nvm`) pada instance atau server

Anda. Misalnya, jalurnya mungkin `/usr/bin/node`. Jika Anda membiarkannya kosong, AWS Cloud9 mencoba menebak di mana biner Node.js saat mencoba terhubung.

15. Untuk host lompat SSH, masukkan informasi tentang host lompat yang digunakan instance atau server. Gunakan format `USER_NAME@HOSTNAME:PORT_NUMBER` (misalnya,`ec2-user@ip-192-0-2-0:22`).

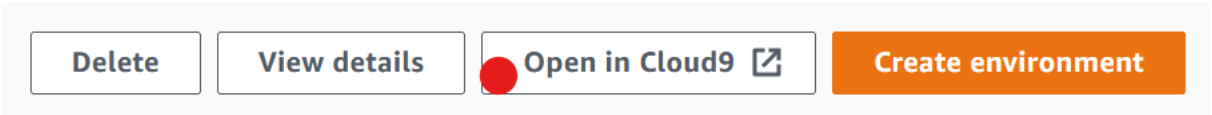
    Host lompat harus memenuhi persyaratan berikut:

    - Jump host harus dapat dijangkau melalui internet publik menggunakan SSH.
    - Jump host harus memungkinkan akses masuk oleh alamat IP apa pun melalui port tertentu.
    - Nilai kunci SSH publik yang disalin ke file `~/.ssh/authorized_keys` pada instans atau server yang ada juga harus disalin ke file `~/.ssh/authorized_keys` pada jump host.
    - Netcat harus diinstal.

16. Menambahkan hingga 50 tanda dengan menyediakan Kunci dan sebuah Nilai untuk setiap tanda. Lakukan dengan memilih Tambahkan tag baru. Tag dilampirkan ke AWS Cloud9 lingkungan sebagai tag sumber daya, dan disebarkan ke sumber daya dasar berikut: AWS CloudFormation tumpukan, EC2 instans Amazon, dan grup EC2 keamanan Amazon. Untuk mempelajari lebih lanjut tentang tag, lihat Mengontrol Akses Menggunakan Tag AWS Sumber Daya di Panduan Pengguna IAM dan informasi lanjutan tentang tag dalam panduan ini.

    > **Warning**
    >
    > Jika Anda memperbarui tag ini setelah Anda membuatnya, perubahan tidak disebarkan ke sumber daya yang mendasarinya. Untuk informasi selengkapnya, lihat Menyebarkan pembaruan tanda ke sumber daya yang mendasari dalam informasi lanjutan tentang tanda.

17. Pilih Buat untuk membuat lingkungan Anda, dan Anda kemudian diarahkan ke halaman beranda. Ketika akun berhasil dibuat, bilah flash hijau muncul di bagian atas AWS Cloud9 konsol. Anda dapat memilih lingkungan baru dan memilih Buka di Cloud9 untuk meluncurkan IDE.




Jika akun gagal dibuat, bilah flash merah muncul di bagian atas AWS Cloud9 konsol. Akun Anda mungkin gagal dibuat karena masalah dengan browser web Anda, izin AWS akses Anda, instans, atau jaringan terkait. Anda dapat menemukan informasi tentang kemungkinan perbaikan

pada masalah yang mungkin menyebabkan akun gagal di bagian AWS Cloud9 Pemecahan Masalah.

> **Note**
>
> Jika lingkungan Anda menggunakan proxy untuk mengakses internet, Anda harus memberikan detail proxy AWS Cloud9 agar dapat menginstal dependensi. Untuk informasi selengkapnya, lihat Gagal menginstal dependensi.

## Mengakses instance no-ingress EC2 dengan AWS Systems Manager

“ EC2 Instance no-ingress” yang dibuat untuk EC2 lingkungan memungkinkan untuk terhubung AWS Cloud9 ke EC2 instance Amazon-nya tanpa perlu membuka port masuk apa pun pada instance itu. Anda dapat memilih opsi no-ingress saat membuat EC2 lingkungan menggunakan konsol, antarmuka baris perintah, atau tumpukan.AWS CloudFormation Untuk informasi selengkapnya tentang cara membuat lingkungan menggunakan konsol atau antarmuka baris perintah, lihatthe section called “Langkah 1: Buat lingkungan”.

> **Important**
>
> Tidak ada biaya tambahan untuk menggunakan Systems Manager Session Manager untuk mengelola koneksi ke EC2 instans Anda.

Saat memilih jenis lingkungan di halaman Buat lingkungan konsol, Anda dapat memilih EC2 instance baru yang memerlukan konektivitas masuk atau EC2 instance tanpa masuk baru yang tidak memerlukan hal berikut:

- EC2Instans baru — Dengan pengaturan ini, grup keamanan untuk instance memiliki aturan untuk mengizinkan lalu lintas jaringan masuk. Lalu lintas jaringan masuk dibatasi untuk alamat IP yang disetujui untuk koneksi AWS Cloud9. Sebuah port inbound terbuka memungkinkan AWS Cloud9 untuk terhubung melalui SSH ke instance-nya. Jika menggunakan AWS Systems Manager Session Manager, Anda dapat mengakses EC2 instans Amazon melalui SSM tanpa membuka port masuk (tanpa masuk). Metode ini hanya berlaku untuk EC2 instans Amazon baru. Untuk informasi selengkapnya, lihat Manfaat menggunakan Systems Manager untuk EC2 lingkungan.

- Komputasi yang ada — Dengan penyiapan ini, EC2 instans Amazon yang ada diakses yang memerlukan detail login SSH yang harus memiliki aturan grup keamanan masuk. Jika Anda memilih opsi ini, peran layanan dibuat secara otomatis. Anda dapat melihat nama peran layanan dalam catatan di bagian bawah layar pengaturan.

Jika membuat lingkungan menggunakan AWS CLI, Anda dapat mengonfigurasi EC2 instance no-ingress dengan menyetel `--connection-type CONNECT_SSM` opsi saat memanggil perintah. `create-environment-ec2` Untuk informasi selengkapnya tentang membuat peran layanan dan profil instans yang diperlukan, lihat Mengelola profil instans untuk Systems Manager dengan AWS CLI.

Setelah Anda selesai membuat lingkungan yang menggunakan EC2 instance no-ingress, konfirmasikan hal berikut:

- Systems Manager Session Manager memiliki izin untuk melakukan tindakan atas EC2 instans atas nama Anda. Untuk informasi selengkapnya, lihat Mengelola izin Systems Manager.
- AWS Cloud9 pengguna dapat mengakses instance yang dikelola oleh Session Manager. Untuk informasi selengkapnya, lihat Memberi pengguna akses ke instans yang dikelola oleh Pengelola Sesi.

## Manfaat menggunakan Systems Manager untuk EC2 lingkungan

Mengizinkan Session Manager untuk menangani koneksi aman antara AWS Cloud9 dan EC2 instance-nya menawarkan dua manfaat utama:

- Tidak ada persyaratan untuk membuka port masuk untuk instans
- Opsi untuk meluncurkan instans ke subnet privat atau publik

No open inbound ports

> Koneksi aman antara AWS Cloud9 dan EC2 instance-nya ditangani oleh Session Manager. Session Manager adalah kemampuan Systems Manager yang dikelola sepenuhnya yang memungkinkan AWS Cloud9 untuk terhubung ke EC2 instance-nya tanpa perlu membuka port masuk.

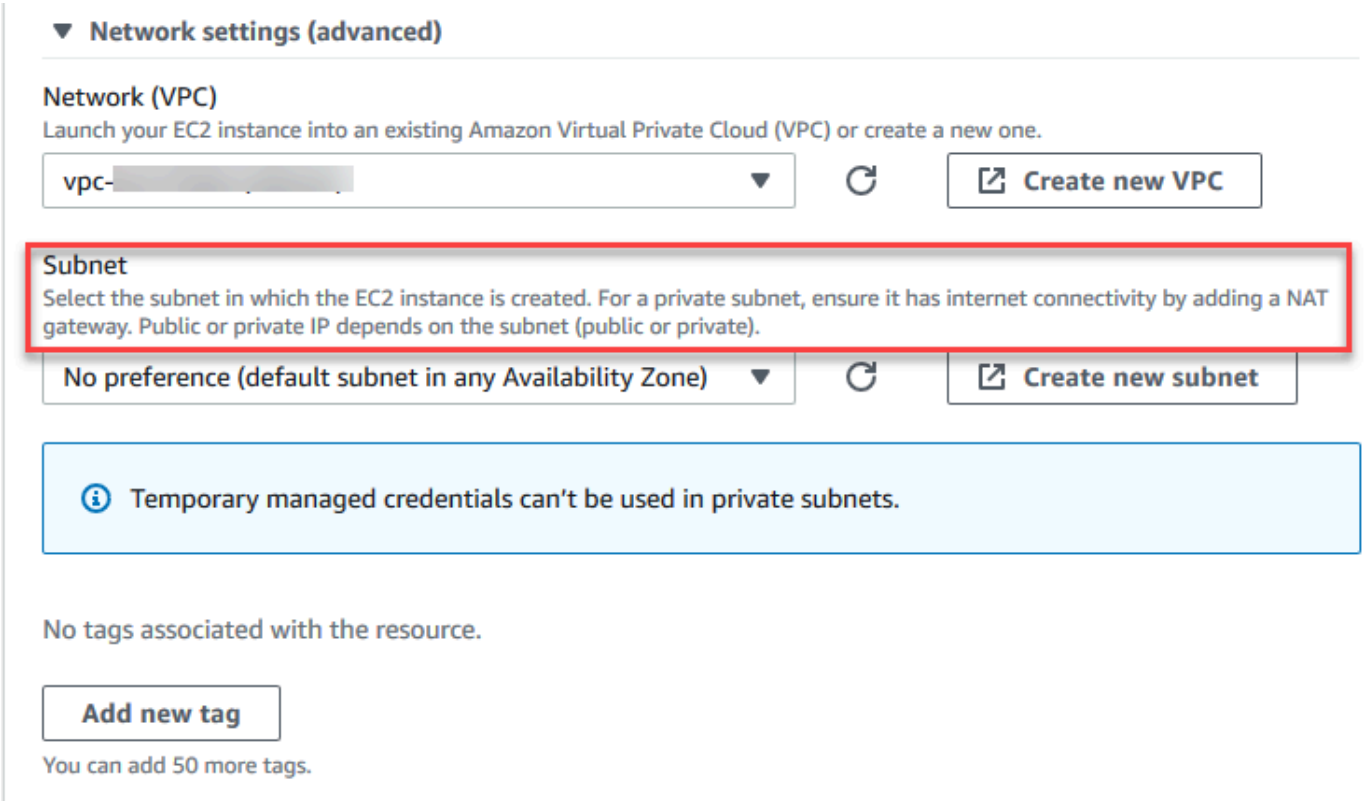
> **Important**
>
> Opsi untuk menggunakan Systems Manager untuk koneksi tanpa masuk saat ini hanya tersedia saat membuat lingkungan baru EC2 .

Dengan dimulainya sesi Pengelola Sesi, koneksi dibuat ke instans target. Dengan koneksi yang ada, lingkungan sekarang dapat berinteraksi dengan instans melalui layanan Systems Manager. Layanan Systems Manager berkomunikasi dengan instans melalui Systems Manager Agent (SSM Agent).

Secara default, Agen SSM diinstal pada semua instance yang digunakan oleh EC2 lingkungan.

Private/public subnets

Ketika memilih subnet untuk instans Anda di bagian Pengaturan jaringan (lanjutan), Anda dapat memilih subnet privat atau publik jika instans untuk lingkungan Anda diakses melalui Systems Manager.




Subnet pribadi

Untuk subnet pribadi, pastikan bahwa instance masih dapat terhubung ke layanan SSM. Hal ini dapat dilakukan dengan menyiapkan gateway NAT di subnet publik atau mengonfigurasi VPC endpoint untuk Systems Manager.

Keuntungan menggunakan gateway NAT adalah bahwa hal itu mencegah internet memulai koneksi ke instans di subnet privat. Instance untuk lingkungan Anda diberi alamat IP pribadi, bukan alamat publik. Jadi, gateway NAT meneruskan lalu lintas dari instance ke internet atau AWS layanan lain, dan kemudian mengirimkan respons kembali ke instance.

Untuk opsi VPC, buat setidaknya tiga titik akhir antarmuka yang diperlukan untuk Systems Manager: com.amazonaws.region.ssm, com.amazonaws.region.ec2messages, dan com.amazonaws.region.ssmmessages. Untuk informasi selengkapnya, lihat Membuat VPC endpoint untuk Systems Manager di Panduan Pengguna AWS Systems Manager .

> **Important**
>
> Saat ini, jika EC2 instance untuk lingkungan Anda diluncurkan ke subnet pribadi, Anda tidak dapat menggunakan kredensyal sementara AWS terkelola untuk memungkinkan EC2 lingkungan mengakses AWS layanan atas nama AWS entitas (pengguna IAM, misalnya).

Subnet publik

Jika lingkungan pengembangan Anda menggunakan SSM untuk mengakses EC2 instans, pastikan instans tersebut diberi alamat IP publik oleh subnet publik yang diluncurkan. Untuk melakukannya, Anda dapat menentukan alamat IP Anda sendiri atau mengaktifkan penugasan otomatis alamat IP publik. Untuk langkah-langkah yang terlibat dalam memodifikasi pengaturan IP penugasan otomatis, lihat Pengalamatan IP dalam VPC Anda di Panduan Pengguna Amazon VPC.

Untuk informasi selengkapnya tentang mengonfigurasi subnet privat dan publik untuk instans lingkungan Anda, lihat Buat subnet untuk AWS Cloud9.

## Mengelola izin Systems Manager

Secara default, Systems Manager tidak memiliki izin untuk melakukan tindakan pada EC2 instance. Akses disediakan melalui profil instans AWS Identity and Access Management (IAM). (Profil instance adalah wadah yang meneruskan informasi peran IAM ke EC2 instance saat peluncuran.)

Saat Anda membuat EC2 instance no-ingress menggunakan AWS Cloud9 konsol, service role (`AWSCloud9SSMAccessRole`) dan IAM instance profile (`AWSCloud9SSMInstanceProfile`) dibuat secara otomatis untuk Anda. (Anda dapat melihat `AWSCloud9SSMAccessRole` di konsol manajemen IAM. Profil instans tidak ditampilkan di konsol IAM.)

> **Important**
>
> Jika Anda membuat EC2 lingkungan no-ingress untuk pertama kalinya AWS CLI, Anda harus secara eksplisit menentukan peran layanan dan profil instance yang diperlukan. Untuk informasi selengkapnya, lihat Mengelola profil instans untuk Systems Manager dengan AWS CLI.

> **Important**
>
> Jika Anda membuat AWS Cloud9 lingkungan dan menggunakan Amazon EC2 Systems Manager dengan `AWSCloud9User` kebijakan `AWSCloud9Administrator` atau kebijakan yang dilampirkan, Anda juga harus melampirkan kebijakan khusus yang memiliki izin IAM tertentu, lihat. Kebijakan IAM khusus untuk pembuatan lingkungan SSM Hal ini disebabkan masalah izin dengan `AWSCloud9User` kebijakan `AWSCloud9Administrator` dan kebijakan.

Untuk perlindungan keamanan tambahan, peran AWS Cloud9 terkait layanan`AWSServiceRoleforAWSCloud9`, menampilkan `PassRole` pembatasan dalam kebijakannya. `AWSCloud9ServiceRolePolicy` Saat Anda meneruskan IAM role ke layanan, itu memungkinkan layanan tersebut untuk mengambil peran dan melakukan tindakan atas nama Anda. Dalam hal ini, `PassRole` izin memastikan bahwa hanya AWS Cloud9 dapat meneruskan `AWSCloud9SSMAccessRole` peran (dan izinnya) ke sebuah EC2 instance. Ini membatasi tindakan yang dapat dilakukan pada EC2 instance hanya untuk yang diperlukan oleh AWS Cloud9.

> **Note**
>
> Jika Anda tidak perlu lagi menggunakan Systems Manager untuk mengakses instans, Anda dapat menghapus peran layanan `AWSCloud9SSMAccessRole`. Untuk informasi lebih lanjut, lihat Menghapus peran atau profil instans dalam Panduan Pengguna IAM.

### Mengelola profil instans untuk Systems Manager dengan AWS CLI

Anda juga dapat membuat EC2 lingkungan tanpa masuknya dengan. AWS CLI Ketika Anda memanggil `create-environment-ec2`, atur opsi `--connection-type` ke `CONNECT_SSM`.

Jika Anda menggunakan opsi ini, peran layanan `AWSCloud9SSMAccessRole` dan `AWSCloud9SSMInstanceProfile` tidak dibuat secara otomatis. Jadi, untuk membuat profil layanan dan profil instance yang diperlukan, lakukan salah satu hal berikut:

- Buat EC2 lingkungan menggunakan konsol setelah memiliki peran `AWSCloud9SSMAccessRole` layanan dan `AWSCloud9SSMInstanceProfile` dibuat secara otomatis sesudahnya. Setelah dibuat, peran layanan dan profil instance tersedia untuk EC2 Lingkungan tambahan apa pun yang dibuat menggunakan AWS CLI.
- Jalankan AWS CLI perintah berikut untuk membuat peran layanan dan profil instance.

```
aws iam create-role --role-name AWSCloud9SSMAccessRole --path /service-role/ --
assume-role-policy-document '{"Version": "2012-10-17","Statement": [{"Effect":
 "Allow","Principal": {"Service": ["ec2.amazonaws.com","cloud9.amazonaws.com"]
  },"Action": "sts:AssumeRole"}]}'
aws iam attach-role-policy --role-name AWSCloud9SSMAccessRole --policy-arn
 arn:aws:iam::aws:policy/AWSCloud9SSMInstanceProfile
aws iam create-instance-profile --instance-profile-name AWSCloud9SSMInstanceProfile
 --path /cloud9/
aws iam add-role-to-instance-profile --instance-profile-name
 AWSCloud9SSMInstanceProfile --role-name AWSCloud9SSMAccessRole
```

## Memberi pengguna akses ke instans yang dikelola oleh Pengelola Sesi

Untuk membuka AWS Cloud9 lingkungan yang terhubung ke EC2 instans melalui Systems Manager, pengguna harus memiliki izin untuk operasi API`StartSession`. Operasi ini memulai koneksi ke EC2 instance terkelola untuk sesi Session Manager. Anda dapat memberi pengguna akses dengan menggunakan kebijakan terkelola AWS Cloud9 tertentu (disarankan) atau dengan mengedit kebijakan IAM dan menambahkan izin yang diperlukan.

| Metode | Deskripsi |
| --- | --- |
| Gunakan kebijakan AWS Cloud9 terkelola - spesifik | Sebaiknya gunakan kebijakan AWS terkelola untuk memungkinkan pengguna mengakses |

| Metode | Deskripsi |
| --- | --- |
| | EC2 instans yang dikelola oleh Systems Manager. Kebijakan terkelola menyediakan serangkaian izin untuk kasus AWS Cloud9 penggunaan standar dan dapat dengan mudah dilampirkan ke entitas IAM.<br><br>Semua kebijakan terkelola juga menyertak an izin untuk menjalankan operasi API `StartSession` . Berikut ini adalah kebijakan terkelola khusus untuk AWS Cloud9:<br><br>• `AWSCloud9Administrator` (`arn:aws:iam::aws:policy/AWS Cloud9Administrator` )<br>• `AWSCloud9User` (`arn:aws:i am::aws:policy/AWSCloud9User` )<br>• `AWSCloud9EnvironmentMember` (`arn:aws:iam::aws:policy/AWS Cloud9EnvironmentMember` )<br><br>**Important**<br>Jika Anda membuat AWS Cloud9 lingkungan dan menggunakan Amazon EC2 Systems Manager dengan `AWSCloud9User` kebijakan `AWSCloud9Administrator` atau kebijakan yang dilampirkan, Anda juga harus melampirkan kebijakan khusus yang memiliki izin IAM tertentu, lihat. Kebijakan IAM khusus untuk pembuatan lingkungan SSM Hal ini disebabkan masalah izin dengan `AWSCloud9User` kebijakan |

| Metode | Deskripsi |
| --- | --- |
| | `AWSCloud9Administrator` dan kebijakan.<br><br>Untuk informasi selengkapnya, lihat AWS kebijakan terkelola untuk AWS Cloud9. |
| Mengedit kebijakan IAM dan menambahkan pernyataan kebijakan yang diperlukan | Untuk mengedit kebijakan yang ada, Anda dapat menambahkan izin untuk `StartSession` API. Untuk mengedit kebijakan menggunakan AWS Management Console atau AWS CLI, ikuti petunjuk yang disediakan oleh Mengedit kebijakan IAM di Panduan Pengguna IAM.<br><br>Saat mengedit kebijakan, tambahkan policy statement (lihat berikut ini) yang mengizinkan operasi API `ssm:startSession` untuk berjalan. |

Anda dapat menggunakan izin berikut untuk menjalankan operasi `StartSession` API. Kunci `ssm:resourceTag` kondisi menentukan bahwa sesi Session Manager dapat dimulai untuk setiap instance (`Resource: arn:aws:ec2:*:*:instance/*`) dengan syarat bahwa instance adalah lingkungan AWS Cloud9 EC2 pengembangan (`aws:cloud9:environment`).

> **Note**
>
> Kebijakan terkelola berikut juga mencakup pernyataan kebijakan ini: `AWSCloud9Administrator`, `AWSCloud9User`, dan `AWSCloud9EnvironmentMember`.

```
{
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
```

```
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        }
```

## Menggunakan AWS CloudFormation untuk membuat lingkungan tanpa masuknya EC2

Saat menggunakan AWS CloudFormation template untuk menentukan lingkungan EC2 pengembangan Amazon tanpa masuk, lakukan hal berikut sebelum membuat tumpukan:

1. Buat peran layanan `AWSCloud9SSMAccessRole` dan profil instans `AWSCloud9SSMInstanceProfile`. Untuk informasi selengkapnya, lihat Membuat peran layanan dan profil instance dengan AWS CloudFormation template.
2. Perbarui kebijakan untuk pemanggilan AWS CloudFormation entitas IAM. Dengan cara ini, entitas dapat memulai sesi Session Manager yang terhubung ke EC2 instance. Untuk informasi selengkapnya, lihat Menambahkan izin Systems Manager ke kebijakan IAM.

### Membuat peran layanan dan profil instance dengan AWS CloudFormation template

Anda perlu membuat peran layanan `AWSCloud9SSMAccessRole` dan profil instans `AWSCloud9SSMInstanceProfile` untuk mengaktifkan Systems Manager mengelola EC2 instans yang mendukung lingkungan pengembangan Anda.

Jika sebelumnya Anda membuat `AWSCloud9SSMAccessRole` dan `AWSCloud9SSMInstanceProfile` dengan membuat EC2 lingkungan no-ingress with the console

atau menjalankan AWS CLI perintah, peran layanan dan profil instance sudah tersedia untuk digunakan.

> **Note**
>
> Misalkan Anda mencoba membuat AWS CloudFormation tumpukan untuk EC2 lingkungan tanpa masuk tetapi Anda tidak terlebih dahulu membuat peran layanan dan profil instance yang diperlukan. Kemudian, tumpukan tidak dibuat dan pesan kesalahan berikut ditampilkan: Profil instans AWSCloud9 SSMInstance Profil tidak ada di akun.

Saat membuat EC2 lingkungan no-ingress untuk pertama kalinya menggunakan AWS CloudFormation, Anda dapat menentukan `AWSCloud9SSMAccessRole` dan `AWSCloud9SSMInstanceProfile` sebagai sumber daya IAM dalam template.

Kutipan dari template sampel ini menunjukkan cara mendefinisikan sumber daya ini. `AssumeRole`Tindakan mengembalikan kredensyal keamanan yang menyediakan akses ke AWS Cloud9 lingkungan dan instance-nya EC2 .

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  AWSCloud9SSMAccessRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service:
              - cloud9.amazonaws.com
              - ec2.amazonaws.com
            Action:
              - 'sts:AssumeRole'
      Description: 'Service linked role for AWS Cloud9'
      Path: '/service-role/'
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AWSCloud9SSMInstanceProfile
      RoleName: 'AWSCloud9SSMAccessRole'

  AWSCloud9SSMInstanceProfile:
```

```
    Type: "AWS::IAM::InstanceProfile"
    Properties:
      InstanceProfileName: AWSCloud9SSMInstanceProfile
      Path: "/cloud9/"
      Roles:
        -
          Ref: AWSCloud9SSMAccessRole
```

## Menambahkan izin Systems Manager ke kebijakan IAM

Setelah mendefinisikan peran layanan dan profil instance dalam AWS CloudFormation template, pastikan bahwa entitas IAM yang membuat tumpukan memiliki izin untuk memulai sesi Session Manager. Sesi adalah koneksi yang dibuat ke EC2 instance menggunakan Session Manager.

> **Note**
>
> Jika Anda tidak menambahkan izin untuk memulai sesi Pengelola Sesi sebelum membuat tumpukan untuk EC2 lingkungan tanpa masuknya, `AccessDeniedException` kesalahan akan ditampilkan.

Tambahkan izin berikut ke kebijakan untuk entitas IAM dengan menelepon. AWS CloudFormation

```
{
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloudformation.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
```

```
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        }
```

## Mengonfigurasi VPC endpoint untuk Amazon S3 guna mengunduh dependensi

Jika EC2 instans AWS Cloud9 lingkungan Anda tidak memiliki akses ke internet, buat titik akhir VPC untuk bucket Amazon S3 tertentu. Bucket ini berisi dependensi yang diperlukan untuk menyimpan IDE Anda. up-to-date

Menyiapkan titik akhir VPC untuk Amazon S3 juga melibatkan penyesuaian kebijakan akses. Anda ingin kebijakan akses mengizinkan akses hanya ke bucket S3 tepercaya yang berisi dependensi yang akan diunduh.

> **Note**
>
> Anda dapat membuat dan mengonfigurasi titik akhir VPC menggunakan AWS Management Console,, AWS CLI atau Amazon VPC API. Prosedur berikut menunjukkan cara membuat titik akhir VPC dengan menggunakan antarmuka konsol.

Membuat dan mengonfigurasi VPC endpoint untuk Amazon S3

1. Di AWS Management Console, buka halaman konsol untuk Amazon VPC.
2. Pilih Titik akhir di bilah navigasi.
3. Di halaman Titik Akhir, pilih Buat Titik Akhir.
4. Di halaman Buat Titik Akhir, masukkan “s3" di bidang pencarian dan tekan Kembali untuk mencantumkan titik akhir yang tersedia untuk Amazon S3 saat ini. Wilayah AWS
5. Dari daftar titik akhir Amazon S3 yang dikembalikan, pilih jenis Gateway.
6. Selanjutnya, pilih VPC yang berisi instance lingkungan Anda. EC2
7. Sekarang pilih tabel rute VPC. Dengan cara ini, subnet terkait dapat mengakses titik akhir. EC2 Contoh lingkungan Anda ada di salah satu subnet ini.
8. Di bagian Kebijakan, pilih opsi Kustom, dan ganti kebijakan standar dengan yang berikut ini.

```
{
```

```
  "Version": "2008-10-17",
  "Statement": [
      {
          "Sid": "Access-to-C9-bucket-only",
          "Effect": "Allow",
          "Principal": "*",
          "Action": "s3:GetObject",
          "Resource": "arn:aws:s3:::{bucket_name}/content/dependencies/*"
      }
  ]
}
```

Untuk `Resource` elemen, ganti `{bucket_name}` dengan nama sebenarnya dari bucket yang tersedia di Anda Wilayah AWS. Misalnya, jika Anda menggunakan AWS Cloud9 di Wilayah Eropa (Irlandia), Anda menentukan yang berikut ini`"Resource": "arn:aws:s3:::static-eu-west-1-prod-static-hld3vzaf7c4h/content/dependencies/`:

Tabel berikut mencantumkan nama bucket untuk Wilayah AWS tempat AWS Cloud9 tersedia.

Ember Amazon S3 di Wilayah AWS Cloud9

| Wilayah AWS | Nama Bucket |
| --- | --- |
| AS Timur (Ohio) | `static-us-east-2-prod-static-1c3sfcvf9hy4m` |
| AS Timur (Virginia Utara) | `static-us-east-1-prod-static-mft1klnkc4hl` |
| AS Barat (Oregon) | `static-us-west-2-prod-static-p21mksqx9zlr` |
| AS Barat (California Utara) | `static-us-west-1-prod-static-16d59zrrp01z0` |
| Afrika (Cape Town) | `static-af-south-1-prod-static-v6v7i5ypdppv` |
| Asia Pasifik (Hong Kong) | `static-ap-east-1-prod-static-171xhpfkrorh6` |

| Wilayah AWS | Nama Bucket |
| --- | --- |
| Asia Pasifik (Mumbai) | `static-ap-south-1-prod-static-ykocre202i9d` |
| Asia Pasifik (Osaka) | `static-ap-northeast-3-prod-static-ivmxqzrx2ioi` |
| Asia Pasifik (Seoul) | `static-ap-northeast-2-prod-static-1wxyctlhwiajm` |
| Asia Pasifik (Singapura) | `static-ap-southeast-1-prod-static-13ibpyrx4vk6d` |
| Asia Pasifik (Sydney) | `static-ap-southeast-2-prod-static-1cjsl8bx27rfu` |
| Asia Pasifik (Tokyo) | `static-ap-northeast-1-prod-static-4fwvbdisquj8` |
| Kanada (Pusat) | `static-ca-central-1-prod-static-g80lpejy486c` |
| Eropa (Frankfurt) | `static-eu-central-1-prod-static-14lbgls2vrkh` |
| Eropa (Irlandia) | `static-eu-west-1-prod-static-hld3vzaf7c4h` |
| Eropa (London) | `static-eu-west-2-prod-static-36lbg202837x` |
| Eropa (Milan) | `static-eu-south-1-prod-static-1379tzkd3ni7d` |
| Eropa (Paris) | `static-eu-west-3-prod-static-1rwpkf766ke58` |
| Eropa (Stockholm) | `static-eu-north-1-prod-static-1qzw982y7yu7e` |

| Wilayah AWS | Nama Bucket |
| --- | --- |
| Timur Tengah (Bahrain) | `static-me-south-1-prod-static-gmljex38qtqx` |
| Amerika Selatan (Sao Paulo) | `static-sa-east-1-prod-static-1cl8k0y7opidt` |
| Israel (Tel Aviv) | `static-il-central-1-prod-static-k02vrnhcesue` |

9. Pilih Buat Titik Akhir.

   Jika Anda memberikan informasi konfigurasi yang benar, pesan akan menampilkan ID titik akhir yang dibuat.

10. Untuk memeriksa apakah IDE Anda dapat mengakses bucket Amazon S3, mulai sesi terminal dengan memilih Jendela, Terminal Baru pada bilah menu. Kemudian jalankan perintah berikut, ganti `{bucket_name}` dengan nama sebenarnya dari bucket untuk Region Anda.

    ```
    ping {bucket_name}.s3.{region}.amazonaws.com.
    ```

    Misalnya, jika Anda membuat endpoint untuk bucket S3 di Wilayah AS Timur (Virginia N.), jalankan perintah berikut.

    ```
    ping static-us-east-1-prod-static-mft1klnkc4hl.s3.us-east-1.amazonaws.com
    ```

    Jika ping mendapat respons, ini mengonfirmasi bahwa IDE Anda dapat mengakses bucket dan dependensinya.

Untuk informasi selengkapnya tentang fitur ini, lihat Titik Akhir untuk Amazon S3 di AWS PrivateLink Panduan.

## Mengonfigurasi VPC endpoint untuk konektivitas privat

Saat Anda meluncurkan instance ke subnet dengan opsi akses menggunakan Systems Manager, grup keamanannya tidak memiliki aturan masuk untuk mengizinkan lalu lintas jaringan masuk. Namun, kelompok keamanan memiliki aturan keluar yang memungkinkan lalu lintas keluar dari

instance. Ini diperlukan untuk mengunduh paket dan pustaka yang diperlukan agar AWS Cloud9 IDE tetap up to date.

Untuk mencegah lalu lintas keluar dan masuk misalnya, buat dan konfigurasikan titik akhir Amazon VPC untuk Systems Manager. Dengan titik akhir VPC antarmuka (titik akhir antarmuka), Anda dapat terhubung ke layanan yang didukung oleh. AWS PrivateLink AWS PrivateLink adalah teknologi yang dapat digunakan untuk mengakses Amazon EC2 dan Systems Manager secara pribadi APIs dengan menggunakan alamat IP pribadi. Untuk mengonfigurasi VPC endpoint agar menggunakan Systems Manager, ikuti petunjuk yang diberikan oleh sumber daya Pusat Pengetahuan ini.

> **Warning**
>
> Misalnya Anda mengonfigurasi grup keamanan yang tidak mengizinkan lalu lintas jaringan masuk atau keluar. Kemudian, EC2 instance yang mendukung AWS Cloud9 IDE Anda tidak memiliki akses internet. Anda perlu membuat endpoint Amazon S3 untuk VPC Anda agar memungkinkan akses ke dependensi yang terdapat dalam bucket S3 tepercaya. Selain itu, beberapa Layanan AWS, seperti AWS Lambda, mungkin tidak berfungsi sebagaimana dimaksud tanpa akses internet.
> Dengan AWS PrivateLink, ada biaya pemrosesan data untuk setiap gigabyte yang diproses melalui titik akhir VPC. Ini terlepas dari sumber lalu lintas atau tujuan. Untuk informasi lebih lanjut, lihat Harga AWS PrivateLink.

## Membuka lingkungan di AWS Cloud9

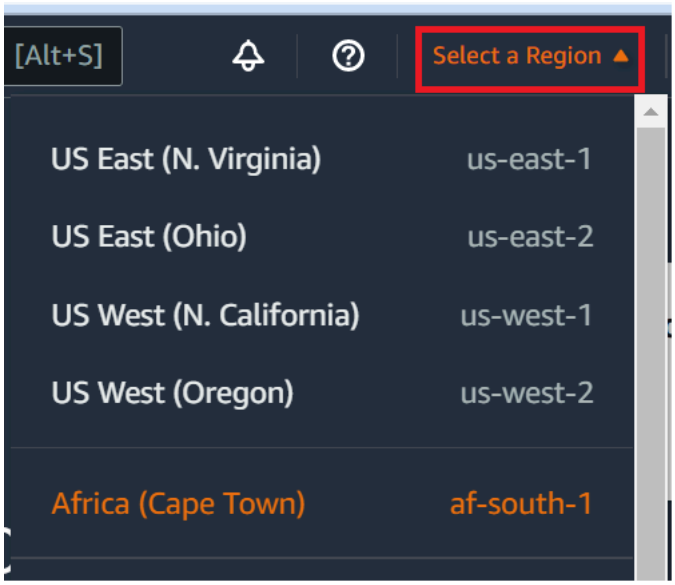
Prosedur ini menjelaskan cara membuka lingkungan di AWS Cloud9.

> **Note**
>
> Prosedur ini mengasumsikan bahwa Anda sudah menciptakan lingkungan AWS Cloud9 pengembangan. Untuk membuat lingkungan, lihat Membuat Lingkungan.

1. Masuk ke AWS Cloud9 konsol sebagai berikut:
   - Jika Anda satu-satunya yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, tanyakan Akun AWS administrator Anda untuk instruksi masuk.

> **Important**
>
> Jika Anda keluar dari Anda Akun AWS, AWS Cloud9 IDE masih dapat diakses hingga 5 menit setelahnya. Akses kemudian ditolak ketika izin yang diperlukan kedaluwarsa.

2. Di bilah navigasi atas, pilih Wilayah AWS tempat lingkungan berada.




3. Dalam daftar lingkungan, untuk lingkungan yang ingin Anda buka, lakukan salah satu tindakan berikut:
   - Di dalam kartu, pilih tautan Buka di Cloud9.
   - Pilih kartu, lalu pilih tombol Buka di Cloud9.

Jika lingkungan Anda tidak ditampilkan di konsol, coba lakukan satu atau beberapa tindakan berikut agar lingkungan ditampilkan.

- Di bilah menu tarik-turun di halaman Lingkungan, pilih satu atau lebih dari yang berikut ini.
  - Pilih lingkungan Saya untuk menampilkan semua lingkungan yang dimiliki AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.
  - Pilih Dibagikan dengan saya untuk menampilkan semua lingkungan yang diundang oleh AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.

  - Pilih Semua lingkungan akun untuk menampilkan semua lingkungan dalam lingkungan Akun AWS yang dipilih Wilayah AWS dan AWS entitas Anda memiliki izin untuk ditampilkan.
- Jika Anda merasa Anda adalah anggota lingkungan, tetapi lingkungan tidak ditampilkan dalam daftar Dibagikan dengan Anda, periksa dengan pemilik lingkungan.
- Di bilah navigasi atas, pilih yang berbeda Wilayah AWS.

## Menelepon Layanan AWS dari lingkungan di AWS Cloud9

Anda dapat menelepon Layanan AWS dari lingkungan AWS Cloud9 pengembangan. Misalnya, Anda dapat melakukan tindakan berikut:

- Mengunggah dan mengunduh data di bucket Amazon Simple Storage Service (Amazon S3).
- Mengirim notifikasi siaran melalui topik Amazon Simple Notification Service (Amazon SNS).
- Membaca dan menulis data di basis data Amazon DynamoDB (DynamoDB).

Anda dapat menelepon Layanan AWS dari lingkungan Anda dengan beberapa cara. Misalnya, Anda dapat menggunakan perintah AWS Command Line Interface (AWS CLI) atau AWS CloudShell to run dari sesi terminal. Anda juga dapat menelepon Layanan AWS dari kode yang Anda jalankan di lingkungan Anda. Anda dapat melakukan ini dengan menggunakan AWS SDKs untuk bahasa pemrograman seperti JavaScript, Python, Ruby, PHP, Go, dan C++. Untuk informasi selengkapnya, lihat Sampel AWS CLI dan aws-shell, Panduan AWS Command Line Interface Pengguna, dan. AWS SDKs

Setiap kali AWS CLI, kode AWS CloudShell, atau kode Anda memanggil Layanan AWS, AWS CLI AWS CloudShell, atau kode Anda harus menyediakan satu set kredensi AWS akses bersama dengan panggilan. Kredensial ini menentukan apakah pemanggil memiliki izin yang sesuai untuk melakukan panggilan. Jika kredensial tidak mencakup izin yang sesuai, panggilan gagal.

Ada beberapa cara untuk memberikan kredensial ke lingkungan Anda. Tabel berikut menjelaskan beberapa pendekatan.

| Tipe lingkungan | Pendekatan |
| --- | --- |
| EC2 | Gunakan kredensyal sementara AWS terkelola.<br><br>Kami merekomendasikan pendekatan ini untuk suatu EC2 lingkungan. AWS kredensya |

| Tipe lingkungan | Pendekatan |
| --- | --- |
| | l sementara terkelola mengelola kredensya l AWS akses di EC2 lingkungan atas nama Anda, sementara juga mengikuti AWS praktik terbaik keamanan.<br><br>Jika Anda menggunakan EC2 lingkungan, Anda dapat melewati sisa topik ini. Ini karena kredenal sementara yang AWS dikelola sudah disiapkan untuk Anda di lingkungan.<br><br>Untuk informasi selengkapnya, lihat Kredensial Sementara terkelola AWS. |
| EC2 | Lampirkan profil instans IAM ke instans.<br><br>Hanya gunakan pendekatan ini jika karena alasan tertentu Anda tidak dapat menggunak an kredenal sementara AWS terkelola. Mirip dengan kredenal sementara AWS terkelola , profil instans mengelola kredenal AWS akses atas nama Anda. Namun, Anda harus membuat, mengelola, dan melampirkan profil instance ke EC2 instans Amazon sendiri.<br><br>Untuk petunjuk, lihat Membuat dan Menggunak an Profil Instans untuk mengelola Kredensial Sementara. |

| Tipe lingkungan | Pendekatan |
| --- | --- |
| EC2 atau SSH | Simpan kredensyal AWS akses permanen Anda di lingkungan.<br><br>Pendekatan ini kurang aman dibanding kan menggunakan kredensyal AWS akses sementara. Namun, ini satu-satunya pendekata n yang didukung untuk lingkungan SSH.<br><br>Untuk petunjuk, lihat Membuat dan Menyimpan Kredensial Akses Permanen di Lingkungan. |
| EC2 atau SSH | Masukkan kredensyal AWS akses permanen Anda langsung ke kode Anda.<br><br>Kami tidak menyarankan pendekatan ini karena tidak mengikuti praktik terbaik AWS keamanan.<br><br>Karena kami mencegah pendekatan ini, kami tidak mencakupnya dalam topik ini. |

## Membuat dan menggunakan profil instans untuk mengelola kredensial sementara

> **Note**
>
> Anda tidak dapat menggunakan prosedur ini untuk lingkungan pengembangan AWS Cloud9 SSH. Sebagai gantinya, lanjutkan ke Membuat dan Menyimpan Kredensial Akses Permanen di Lingkungan.
> Sebaiknya gunakan kredensial sementara AWS terkelola, bukan profil instans. Ikuti petunjuk ini hanya jika karena alasan tertentu Anda tidak dapat menggunakan kredenal sementara yang AWS dikelola. Untuk informasi selengkapnya, lihat Kredensial Sementara terkelola AWS.

Prosedur ini menggunakan IAM dan Amazon EC2 untuk membuat dan melampirkan profil instans IAM ke EC2 instans Amazon yang terhubung ke lingkungan Anda. Profil instans ini mengelola kredensial sementara atas nama Anda. Prosedur ini mengasumsikan Anda telah membuat lingkungan di AWS Cloud9. Untuk membuat lingkungan, lihat Membuat Lingkungan.

Anda dapat menyelesaikan tugas-tugas ini dengan EC2 konsol IAM dan Amazon atau Antarmuka Baris AWS Perintah ()AWS CLI.

## Membuat profil instans dengan konsol IAM

> **Note**
>
> Jika Anda sudah memiliki peran IAM yang berisi profil instans, lanjutkan ke Lampirkan Profil Instance ke Instance dengan EC2 Konsol Amazon.

1. Masuk ke konsol IAM, di https://console.aws.amazon.com/iam.

   Untuk langkah ini, kami sarankan Anda masuk menggunakan kredensi tingkat administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

2. Di bilah navigasi, pilih Peran.

   > **Note**
   >
   > Anda tidak dapat menggunakan konsol IAM untuk membuat profil instans dengan sendirinya. Anda harus membuat IAM role, yang berisi profil instans.

3. Pilih Buat peran.
4. Pada halaman Pilih jenis entitas tepercaya, dengan Layanan AWSsudah dipilih, untuk Pilih layanan yang akan menggunakan peran ini, pilih EC2.
5. Untuk Pilih kasus penggunaan Anda, pilih EC2.
6. Pilih Berikutnya: Izin.
7. Pada halaman Lampirkan kebijakan izin, dalam daftar kebijakan, pilih kotak di samping AdministratorAccess, lalu pilih Berikutnya: Tinjau.

> **Note**
>
> AdministratorAccessKebijakan ini memungkinkan akses tidak terbatas ke semua AWS tindakan dan sumber daya di seluruh Anda Akun AWS. Gunakan hanya untuk tujuan eksperimen. Untuk informasi selengkapnya, lihat langkah Kebijakan IAM dalam Panduan Pengguna IAM.

8. Pada halaman Tinjauan, untuk Nama Peran, masukkan nama untuk peran (misalnya,`my-demo-cloud9-instance-profile`).
9. Pilih Buat Peran.

Lewati ke depan untuk Melampirkan Profil Instance ke Instance dengan EC2 Konsol Amazon.

## Membuat profil instans dengan AWS CLI

> **Note**
>
> Jika Anda sudah memiliki IAM role yang berisi profil instans, lanjutkan ke Melampirkan Profil Instans ke Instans dengan AWS CLI.
> Untuk topik ini, kami sarankan Anda mengonfigurasi kredensial tingkat administrator yang AWS CLI menggunakan. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

> **Note**
>
> Jika Anda menggunakan kredensyal sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

1. Tentukan hubungan kepercayaan AWS untuk peran IAM yang diperlukan profil instance. Untuk melakukannya, buat lalu simpan file dengan isi berikut (misalnya, `my-demo-cloud9-instance-profile-role-trust.json`).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Dengan menggunakan terminal atau command prompt, beralihlah ke direktori tempat Anda baru saja menyimpan file ini.

3. Buat IAM role untuk profil instans. Untuk melakukan ini, jalankan `create-role` perintah IAM. Ketika Anda melakukannya, tentukan nama untuk peran IAM baru (misalnya,`my-demo-cloud9-instance-profile-role`), dan nama file yang baru saja Anda simpan.

```
aws iam create-role --role-name my-demo-cloud9-instance-profile-role --assume-role-policy-document file://my-demo-cloud9-instance-profile-role-trust.json
```

4. Lampirkan izin AWS akses ke peran IAM profil instance. Untuk melakukan ini, jalankan `attach-role-policy` perintah IAM. Tentukan nama peran IAM yang ada dan Nama Sumber Daya Amazon (ARN) dari kebijakan terkelola AWS yang diberi nama. `AdministratorAccess`

```
aws iam attach-role-policy --role-name my-demo-cloud9-instance-profile-role --policy-arn arn:aws:iam::aws:policy/AdministratorAccess
```

> **Note**
>
> AdministratorAccessKebijakan ini memungkinkan akses tidak terbatas ke semua AWS tindakan dan sumber daya di seluruh Anda Akun AWS. Gunakan hanya untuk tujuan

eksperimen. Untuk informasi selengkapnya, lihat langkah Kebijakan IAM dalam Panduan Pengguna IAM.

5. Buat profil instans. Untuk melakukannya, jalankan perintah IAM `create-instance-profile`, tentukan nama untuk profil instans baru (misalnya, `my-demo-cloud9-instance-profile`).

```
aws iam create-instance-profile --instance-profile-name my-demo-cloud9-instance-
profile
```

6. Lampirkan IAM role ke profil instans. Untuk melakukannya, jalankan perintah IAM `add-role-to-instance-profile`, tentukan nama IAM role dan profil instans yang ada.

```
aws iam add-role-to-instance-profile --role-name my-demo-cloud9-instance-profile-role
 --instance-profile-name my-demo-cloud9-instance-profile
```

Lanjutkan ke Membuat Profil Instans dengan AWS CLI.

## Lampirkan profil instance ke instance dengan EC2 konsol Amazon

1. Masuk ke EC2 konsol Amazon, di https://console.aws.amazon.com/ec2.

   Untuk langkah ini, kami sarankan Anda masuk menggunakan kredensi tingkat administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

2. Di bilah navigasi, pastikan pemilih Wilayah menampilkan pemilih Wilayah AWS yang cocok dengan lingkungan Anda. Misalnya, jika Anda menciptakan lingkungan di Wilayah AS Timur (Ohio), pilih US East (Ohio) di pemilih Wilayah di sini.
3. Pilih tautan Instans Berjalan atau, di panel navigasi, perluas Instans, lalu pilih Instans.
4. Dalam daftar instans, pilih instans dengan Nama yang menyertakan nama lingkungan Anda. Misalnya, jika nama lingkungan Anda`my-demo-environment`, pilih instance dengan Nama yang disertakan my-demo-environment.
5. Pilih Tindakan, Keamanan, Ubah peran IAM.

> **Note**
>
> Meskipun Anda melampirkan peran ke instans, peran tersebut berisi profil instans.

6. Pada halaman Ubah peran IAM, untuk peran IAM, pilih nama peran yang Anda identifikasi atau yang Anda buat di prosedur sebelumnya, lalu pilih Terapkan.
7. Kembali di lingkungan, gunakan AWS CLI untuk menjalankan `aws configure` perintah atau AWS CloudShell untuk menjalankan `configure` perintah. Jangan tentukan nilai apa pun untuk ID Kunci AWS Akses atau Kunci Akses AWS Rahasia (tekan `Enter` setelah setiap permintaan ini). Untuk nama Wilayah Default, tentukan yang Wilayah AWS terdekat dengan Anda atau Wilayah tempat AWS sumber daya Anda berada. Misalnya, `us-east-2` untuk Wilayah US East (Ohio). Untuk daftar Wilayah, lihat Wilayah AWS dan Titik Akhir di. Referensi Umum Amazon Web Services Secara opsional, tentukan nilai untuk Format keluaran default (misalnya, `json`).

Anda sekarang dapat mulai menelepon Layanan AWS dari lingkungan Anda. Untuk menggunakan AWS CLI, aws-shell, atau keduanya untuk menelepon Layanan AWS, lihat Sampel AWS CLI dan aws-shell. Untuk menelepon Layanan AWS dari kode Anda, lihat tutorial dan contoh kami yang lain.

## Lampirkan profil instance ke instance dengan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensyal sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

1. Jalankan EC2 `associate-iam-instance-profile` perintah Amazon. Tentukan nama profil instance dan ID serta Wilayah AWS ID EC2 instans Amazon untuk lingkungan.

```
aws ec2 associate-iam-instance-profile --iam-instance-profile Name=my-demo-cloud9-
instance-profile --region us-east-2 --instance-id i-12a3b45678cdef9a0
```

   Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS ID untuk instance dan `i-12a3b45678cdef9a0` dengan ID instance.

   Untuk mendapatkan ID instance, Anda dapat, misalnya, menjalankan EC2 `describe-instances` perintah Amazon, menentukan nama dan Wilayah AWS ID lingkungan.

```
aws ec2 describe-instances --region us-east-2 --filters Name=tag:Name,Values=*my-
environment* --query "Reservations[*].Instances[*].InstanceId" --output text
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS ID untuk instance dan `my-environment` dengan nama lingkungan.

2. Kembali ke lingkungan, gunakan AWS CLI untuk menjalankan `aws configure` perintah atau aws-shell untuk menjalankan `configure` perintah. Jangan tentukan nilai apa pun untuk ID Kunci AWS Akses atau Kunci Akses AWS Rahasia. Tekan `Enter` setelah masing-masing petunjuk ini. Untuk nama Wilayah Default, tentukan yang Wilayah AWS terdekat dengan Anda atau Wilayah tempat AWS sumber daya Anda berada. Misalnya, `us-east-2` untuk Wilayah US East (Ohio). Untuk daftar Wilayah, lihat AWS Wilayah dan Titik Akhir di. Referensi Umum Amazon Web Services Secara opsional, tentukan nilai untuk Format keluaran default (misalnya, `json`).

Anda sekarang dapat mulai menelepon Layanan AWS dari lingkungan Anda. Untuk menggunakan AWS CLI, aws-shell, atau keduanya untuk menelepon Layanan AWS, lihat Sampel AWS CLI dan aws-shell. Untuk menelepon Layanan AWS dari kode Anda, lihat tutorial dan contoh kami yang lain.

## Membuat dan menyimpan kredensial akses permanen di Lingkungan

> **Note**
>
> Jika Anda menggunakan lingkungan AWS Cloud9 EC2 pengembangan, sebaiknya gunakan kredensyal sementara AWS terkelola, bukan kredensyal akses AWS permanen. Untuk bekerja dengan kredensyal sementara yang AWS dikelola, lihat. AWS kredensi sementara yang dikelola

Di bagian ini, Anda menggunakan AWS Identity and Access Management (IAM) untuk menghasilkan satu set kredensyal permanen. The AWS CLI, yang aws-shell, atau kode Anda dapat menggunakan kumpulan kredensyal ini saat menelepon. Layanan AWS Set ini mencakup ID kunci AWS akses dan kunci akses AWS rahasia, yang unik untuk pengguna Anda di Anda Akun AWS. Jika Anda sudah memiliki ID kunci AWS akses dan kunci akses AWS rahasia, perhatikan kredensialnya, lalu lewati ke Simpan Kredensyal Akses Permanen di Lingkungan.

Anda dapat membuat satu set kredensial permanen dengan konsol IAM atau AWS CLI.

## Memberikan akses programatis

Pengguna membutuhkan akses terprogram jika mereka ingin berinteraksi dengan AWS luar. AWS Management Console Cara untuk memberikan akses terprogram tergantung pada jenis pengguna yang mengakses AWS.

Untuk memberi pengguna akses programatis, pilih salah satu opsi berikut.

| Pengguna mana yang membutuhkan akses programatis? | Untuk | Oleh |
| --- | --- | --- |
| Identitas tenaga kerja<br><br>(Pengguna yang dikelola di Pusat Identitas IAM) | Gunakan kredensi sementara untuk menandatangani permintaan terprogram ke AWS CLI,, AWS SDKs atau. AWS APIs | Mengikuti petunjuk untuk antarmuka yang ingin Anda gunakan.<br><br>• Untuk AWS CLI, lihat Mengkonfigurasi yang akan AWS CLI digunakan AWS IAM Identity Center dalam Panduan AWS Command Line Interface Pengguna.<br>• Untuk AWS SDKs, alat, dan AWS APIs, lihat Autentikasi Pusat Identitas IAM di Panduan Referensi Alat AWS SDKs dan Alat. |
| IAM | Gunakan kredensi sementara untuk menandatangani permintaan terprogram ke AWS CLI,, AWS SDKs atau. AWS APIs | Mengikuti petunjuk dalam Menggunakan kredensi sementara dengan AWS sumber daya di Panduan Pengguna IAM. |
| IAM | (Tidak direkomendasikan) Gunakan kredensi jangka panjang untuk menandatangani permintaan terprogra | Mengikuti petunjuk untuk antarmuka yang ingin Anda gunakan. |

| Pengguna mana yang membutuhkan akses programatis? | Untuk | Oleh |
| --- | --- | --- |
| | m ke AWS CLI,, AWS SDKs atau. AWS APIs | • Untuk mengetahui AWS CLI, lihat Mengautentikasi menggunakan kredensi pengguna IAM di Panduan Pengguna.AWS Command Line Interface<br>• Untuk AWS SDKs dan alat, lihat Mengautentikasi menggunakan kredensyal jangka panjang di Panduan Referensi Alat AWS SDKs dan Alat.<br>• Untuk AWS APIs, lihat Mengelola kunci akses untuk pengguna IAM di Panduan Pengguna IAM. |

## Buat kredensyal akses permanen dengan AWS CLI

> **Note**
>
> Untuk bagian ini, kami sarankan Anda mengonfigurasi kredensial tingkat administrator yang AWS CLI menggunakan di bagian Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

> **Note**
>
> Jika Anda menggunakan kredensyal sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda

dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

Jalankan `create-access-key` perintah IAM untuk membuat kunci AWS akses baru dan kunci akses AWS rahasia yang sesuai untuk pengguna.

```
aws iam create-access-key --user-name MyUser
```

Dalam perintah sebelumnya, ganti `MyUser` dengan nama pengguna.

Di lokasi yang aman, simpan nilai `AccessKeyId` dan `SecretAccessKey` yang ditampilkan. Setelah Anda menjalankan `create-access-key` perintah IAM, ini adalah satu-satunya waktu Anda dapat menggunakan AWS CLI untuk melihat kunci akses AWS rahasia pengguna. Untuk membuat kunci akses AWS rahasia baru bagi pengguna nanti jika diperlukan, lihat Membuat, Memodifikasi, dan Melihat Kunci Akses (API, CLI PowerShell,) di Panduan Pengguna IAM.

### Menyimpan kredensial akses permanen di Lingkungan

Dalam prosedur ini, Anda menggunakan AWS Cloud9 IDE untuk menyimpan kredenal AWS akses permanen Anda di lingkungan Anda. Prosedur ini mengasumsikan Anda sudah membuat lingkungan di AWS Cloud9, membuka lingkungan, dan menampilkan AWS Cloud9 IDE di browser web Anda. Untuk informasi selengkapnya, lihat Membuat Lingkungan dan Membuka Lingkungan.

> **Note**
>
> Prosedur berikut menunjukkan cara menyimpan kredensial akses permanen Anda dengan menggunakan variabel lingkungan. Jika Anda memiliki AWS CLI atau aws-shell diinstal di lingkungan Anda, Anda dapat menggunakan **`aws configure`**perintah untuk AWS CLI atau **`configure`**perintah untuk aws-shell untuk menyimpan kredensyal akses permanen Anda sebagai gantinya. Untuk instruksi, lihat Konfigurasi Cepat dalam Panduan Pengguna AWS Command Line Interface .

1. Dengan lingkungan Anda terbuka, di AWS Cloud9 IDE, mulailah sesi terminal baru, jika belum dimulai. Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.
2. Jalankan setiap perintah berikut, satu perintah pada satu waktu, untuk mengatur variabel lingkungan lokal yang mewakili kredensial akses permanen Anda. Dalam perintah ini,

setelah ituAWS_ACCESS_KEY_ID:, masukkan ID kunci AWS akses Anda. Setelah ituAWS_SECRET_ACCESS_KEY, masukkan kunci akses AWS rahasia Anda. Setelah ituAWS_DEFAULT_REGION_ID, masukkan Wilayah AWS pengidentifikasi yang terkait dengan yang Wilayah AWS terdekat dengan Anda (atau pilihan Anda Wilayah AWS). Untuk daftar pengidentifikasi yang tersedia, lihat Wilayah AWS dan Titik Akhir di. Referensi Umum Amazon Web Services Misalnya, untuk AS Timur (Ohio), Anda menggunakanus-east-2.

```
export AWS_ACCESS_KEY_ID=
export AWS_SECRET_ACCESS_KEY=
export AWS_DEFAULT_REGION=
```

3. Perhatikan bahwa variabel lingkungan sebelumnya hanya valid untuk sesi terminal saat ini. Untuk membuat variabel lingkungan ini tersedia di seluruh sesi terminal, Anda harus menambahkannya ke file profil shell Anda sebagai variabel lingkungan pengguna, sebagai berikut.

   a. Di jendela Lingkungan IDE, pilih ikon roda gigi, lalu pilih Tampilkan Beranda di Favorit. Ulangi langkah ini dan pilih Tampilkan File Tersembunyi juga.

   b. Buka file ~/.bashrc.

   c. Masukkan atau tempel kode berikut di akhir file. Dalam perintah ini, setelah ituAWS_ACCESS_KEY_ID:, masukkan ID kunci AWS akses Anda. Setelah ituAWS_SECRET_ACCESS_KEY, masukkan kunci akses AWS rahasia Anda. Setelah ituAWS_DEFAULT_REGION_ID, masukkan Wilayah AWS pengidentifikasi yang terkait dengan yang Wilayah AWS terdekat dengan Anda (atau pilihan Anda Wilayah AWS). Untuk daftar pengidentifikasi yang tersedia, lihat Wilayah AWS dan Titik Akhir di. Referensi Umum Amazon Web Services Misalnya, untuk Wilayah AS Timur (Ohio), Anda menggunakanus-east-2.

   ```
   export AWS_ACCESS_KEY_ID=
   export AWS_SECRET_ACCESS_KEY=
   export AWS_DEFAULT_REGION=
   ```

   d. Simpan file tersebut.

   e. Sumberkan file ~/.bashrc untuk memuat variabel lingkungan baru ini.

   ```
   . ~/.bashrc
   ```

Anda sekarang dapat mulai menelepon Layanan AWS dari lingkungan Anda. Untuk menggunakan AWS CLI atau aws-shell untuk menelepon Layanan AWS, lihat Sampel AWS CLI dan aws-shell. Untuk menelepon Layanan AWS dari kode Anda, lihat tutorial dan contoh kami yang lain.

# Mengubah pengaturan lingkungan di AWS Cloud9

Anda dapat mengubah preferensi atau pengaturan untuk lingkungan AWS Cloud9 pengembangan.

- Ubah Preferensi Lingkungan
- Ubah Pengaturan Lingkungan dengan Konsol
- Ubah Pengaturan Lingkungan dengan Kode

## Mengubah preferensi lingkungan

1. Buka lingkungan yang ingin Anda ubah pengaturannya. Untuk membuka lingkungan, lihat Membuka Lingkungan.
2. Di AWS Cloud9 IDE, pada bilah menu, pilih AWS Cloud9, Preferensi.
3. Di jendela Preferensi, pilih Pengaturan Proyek.
4. Ubah pengaturan proyek apa pun yang tersedia sesuai keinginan Anda. Ini termasuk pengaturan seperti Editor Kode (Ace) dan Temukan di File.

> **Note**
>
> Untuk informasi selengkapnya, lihat Perubahan Pengaturan Proyek yang Dapat Anda Lakukan.

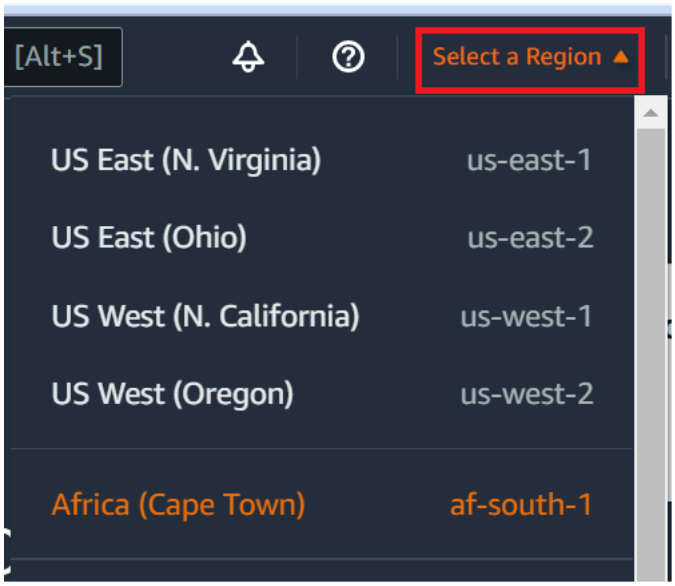
### Menyesuaikan batas waktu lingkungan di IDE AWS Cloud9

Langkah-langkah berikut menguraikan cara memperbarui periode batas waktu untuk EC2 lingkungan Amazon di IDE. AWS Cloud9 Ini akan menjadi jumlah waktu sebelum lingkungan berhenti.

1. Buka lingkungan yang ingin Anda konfigurasikan.
2. Di AWS Cloud9 IDE, pada bilah menu, pilih AWS Cloud9Preferensi.
3. Di jendela Preferensi, gulir ke bagian  EC2 instans Amazon.
4. Pilih nilai batas waktu dari daftar yang tersedia dan perbarui.

# Mengubah pengaturan lingkungan dengan konsol

1. Masuk ke AWS Cloud9 konsol sebagai berikut:
   - Jika Anda satu-satunya individu yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, lihat Akun AWS administrator untuk petunjuk masuk.
2. Di bilah navigasi atas, pilih Wilayah AWS tempat lingkungan berada.




3. Dalam daftar lingkungan, untuk lingkungan yang pengaturannya ingin Anda ubah, lakukan salah satu hal berikut.
   - Pilih judul kartu untuk lingkungan. Kemudian pilih Lihat detail di halaman berikutnya.
   - Pilih kartu untuk lingkungan, lalu pilih tombol Lihat detail.
4. Buat perubahan Anda, dan kemudian pilih Simpan perubahan.

   Anda dapat menggunakan AWS Cloud9 konsol untuk mengubah pengaturan berikut.
   - Untuk EC2 lingkungan, Nama dan Deskripsi.
   - Untuk lingkungan SSH: Nama, Deskripsi, Pengguna, Host, Port, Jalur lingkungan, Jalur biner Node.js, dan Host lompatan SSH.

   Untuk mengubah pengaturan lainnya, lakukan hal berikut.
   - Untuk EC2 lingkungan, lakukan hal berikut.

    - Anda tidak dapat mengubah Jenis, Grup keamanan, VPC, Subnet, Jalur lingkungan, atau ARN Lingkungan.
    - Untuk Izin atau Jumlah anggota, lihat Mengubah Peran Akses Anggota Lingkungan, Menghapus Pengguna Anda, Mengundang pengguna IAM, dan Menghapus Anggota Lingkungan Lain.
    - EC2 Misalnya jenis, Memori, atau vCPU, lihat Memindahkan atau Mengubah Ukuran Lingkungan.
- Untuk lingkungan SSH, lakukan hal berikut.
    - Anda tidak dapat mengubah Jenis atau ARN Lingkungan.
    - Untuk Izin atau Jumlah anggota, lihat Mengubah Peran Akses Anggota Lingkungan, Menghapus Pengguna Anda, Mengundang Pengguna IAM, dan Menghapus Anggota Lingkungan Lainnya.

Jika lingkungan Anda tidak ditampilkan di konsol, coba lakukan satu atau beberapa tindakan berikut agar lingkungan ditampilkan.

- Di bilah menu tarik-turun di halaman Lingkungan, pilih satu atau lebih dari yang berikut ini.
    - Pilih lingkungan Saya untuk menampilkan semua lingkungan yang dimiliki AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.
    - Pilih Dibagikan dengan saya untuk menampilkan semua lingkungan yang diundang oleh AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.
    - Pilih Semua lingkungan akun untuk menampilkan semua lingkungan dalam lingkungan Akun AWS yang dipilih Wilayah AWS dan AWS entitas Anda memiliki izin untuk ditampilkan.
- Jika Anda merasa Anda adalah anggota lingkungan, tetapi lingkungan tidak ditampilkan dalam daftar Dibagikan dengan Anda, periksa dengan pemilik lingkungan.
- Di bilah navigasi atas, pilih yang berbeda Wilayah AWS.

## Mengubah pengaturan lingkungan dengan kode

Untuk menggunakan kode untuk mengubah pengaturan lingkungan AWS Cloud9, panggil operasi lingkungan AWS Cloud9 pembaruan, sebagai berikut.

| | |
|---|---|
| AWS CLI | perbaruan-lingkungan |

| | |
|---|---|
| AWS SDK untuk C++ | UpdateEnvironmentRequest, UpdateEnvironmentResult |
| AWS SDK untuk Go | UpdateEnvironment, UpdateEnvironmentRequest, UpdateEnvironmentWithContext |
| AWS SDK untuk Java | UpdateEnvironmentRequest, UpdateEnvironmentResult |
| AWS SDK untuk JavaScript | UpdateEnvironment |
| AWS SDK untuk .NET | UpdateEnvironmentRequest, UpdateEnvironmentResponse |
| AWS SDK untuk PHP | UpdateEnvironment |
| AWS SDK for Python (Boto) | update_environment |
| AWS SDK untuk Ruby | update_environment |
| AWS Tools for Windows PowerShell | Update-C9Environment |
| AWS Cloud9 API | UpdateEnvironment |

# Bekerja dengan lingkungan bersama di AWS Cloud9

Lingkungan bersama adalah lingkungan AWS Cloud9 pengembangan yang banyak pengguna diundang untuk berpartisipasi. Topik ini memberikan instruksi untuk berbagi lingkungan AWS Cloud9 dan bagaimana berpartisipasi dalam lingkungan bersama.

Untuk mengundang pengguna untuk berpartisipasi dalam lingkungan yang Anda miliki, ikuti salah satu dari serangkaian prosedur ini. Pilih berdasarkan jenis pengguna yang ingin Anda undang.

- Jika Anda adalah pengguna yang Akun AWS sama dengan lingkungan, Anda harus Mengundang Pengguna di Akun yang Sama dengan Lingkungan.
- Jika Anda adalah AWS Cloud9 administrator yang Akun AWS sama dengan lingkungan, khususnya pengguna Akun AWS root, pengguna administrator atau pengguna dengan kebijakan AWS `AWSCloud9Administrator` terkelola terlampir, maka Anda harus mengundang AWS Cloud9 administrator sendiri, lihat Mengundang Pengguna di Akun yang Sama dengan

Lingkungan, atau minta AWS Cloud9 administrator mengundang dirinya sendiri (atau orang lain yang sama Akun AWS), lihat Memiliki AWS Cloud9 Administrator di Akun yang Sama dengan Lingkungan Mengundang Diri Sendiri atau Orang Lain.

## Kasus Penggunaan Lingkungan Bersama

Lingkungan bersama baik untuk kasus penggunaan berikut:

- Pemrograman pasangan (juga dikenal sebagai pemrograman sejawat): Di sinilah dua pengguna bekerja sama pada kode yang sama dalam satu lingkungan. Dalam pemrograman simultan, biasanya satu pengguna menulis kode sementara pengguna lain mengamati kode yang ditulis. Pengamat memberikan masukan langsung dan umpan balik kepada penulis kode. Posisi ini sering beralih selama proyek. Tanpa lingkungan bersama, tim programmer pasangan biasanya duduk di depan satu mesin. Hanya satu pengguna pada satu waktu yang dapat menulis kode. Dengan lingkungan bersama, kedua pengguna dapat duduk di depan mesin mereka sendiri. Selain itu, mereka dapat menulis kode pada saat yang sama, bahkan jika mereka berada di kantor fisik yang berbeda.
- Kelas ilmu komputer: Ini berguna ketika guru atau asisten pengajar ingin mengakses lingkungan siswa. Melakukannya bisa untuk meninjau pekerjaan rumah siswa atau memperbaiki masalah dengan lingkungan mereka secara real time. Siswa juga dapat bekerja sama dengan teman sekelas mereka dalam proyek pekerjaan rumah bersama, menulis kode bersama dalam satu lingkungan secara langsung. Mereka dapat melakukan ini meskipun mereka mungkin berada di lokasi yang berbeda menggunakan sistem operasi komputer dan jenis browser web yang berbeda.
- Situasi lain adalah ketika beberapa pengguna perlu berkolaborasi pada kode yang sama secara langsung.

## Tentang peran akses anggota lingkungan

Sebelum Anda berbagi lingkungan atau berpartisipasi dalam lingkungan bersama AWS Cloud9, Anda harus memahami tingkat izin akses untuk lingkungan bersama. Kami menyebut tingkat izin ini peran akses anggota lingkungan.

Lingkungan bersama di AWS Cloud9 menawarkan tiga peran akses anggota lingkungan: pemilik, baca/tulis, dan hanya-baca.

- Pemilik memiliki kontrol penuh atas lingkungan. Setiap lingkungan memiliki satu dan hanya satu pemilik, yang merupakan pencipta lingkungan. Pemilik dapat melakukan tindakan berikut.

  - Menambahkan, mengubah, dan menghapus anggota untuk lingkungan
  - Buka, lihat, dan edit file
  - Jalankan kode
  - Mengubah pengaturan lingkungan
  - Mengobrol dengan anggota lain
  - Menghapus pesan chat yang ada

  Di AWS Cloud9 IDE, pemilik lingkungan ditampilkan dengan akses Baca+Tulis.
- Seorang anggota baca/tulis dapat melakukan tindakan berikut.
  - Buka, lihat, dan edit file
  - Jalankan kode
  - Ubah berbagai pengaturan lingkungan dari dalam AWS Cloud9 IDE
  - Mengobrol dengan anggota lain
  - Menghapus pesan chat yang ada

  Dalam AWS Cloud9 IDE, anggota baca/tulis ditampilkan dengan akses Baca+Tulis.
- Seorang anggota hanya-baca dapat melakukan tindakan berikut.
  - Buka dan lihat file
  - Mengobrol dengan anggota lain
  - Menghapus pesan chat yang ada

  Di AWS Cloud9 IDE, anggota hanya-baca ditampilkan dengan akses Hanya Baca.

Sebelum pengguna dapat menjadi pemilik atau anggota lingkungan, pengguna tersebut harus memenuhi salah satu kriteria berikut.

- Pengguna adalah pengguna Akun AWS root.
- Pengguna adalah pengguna administrator. Untuk informasi selengkapnya, lihat Membuat Pengguna dan Grup Admin IAM Pertama Anda di Panduan Pengguna IAM.
- Pengguna adalah pengguna yang termasuk dalam grup IAM, pengguna yang mengambil peran, atau pengguna federasi yang mengambil peran, dan grup atau peran tersebut memiliki kebijakan AWS terkelola atau `AWSCloud9User` (`AWSCloud9Administrator`atau`AWSCloud9EnvironmentMember`, hanya untuk menjadi

anggota) yang dilampirkan. Untuk informasi selengkapnya, lihat Kebijakan terkelola AWS (ditentukan sebelumnya).

- Untuk melampirkan salah satu kebijakan terkelola sebelumnya ke grup IAM, Anda dapat menggunakan AWS Management Console atau AWS Antarmuka Baris Perintah (AWS CLI) seperti yang dijelaskan dalam prosedur berikut.
- Anda dapat membuat peran di IAM dengan salah satu kebijakan terkelola sebelumnya untuk diasumsikan oleh pengguna atau pengguna federasi. Untuk informasi selengkapnya, lihat Membuat Peran di Panduan Pengguna IAM. Agar pengguna atau pengguna gabungan mengambil peran tersebut, lihat cakupan asumsi peran dalam Menggunakan IAM Role di Panduan Pengguna IAM.

## Melampirkan kebijakan AWS terkelola AWS Cloud9 untuk grup menggunakan konsol

Prosedur berikut menguraikan cara melampirkan kebijakan AWS terkelola AWS Cloud9 untuk grup menggunakan konsol.

1. Masuk ke AWS Management Console, jika Anda belum masuk.

   Untuk langkah ini, kami sarankan Anda masuk menggunakan kredensi tingkat administrator IAM di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.
2. Buka konsol IAM. Untuk melakukannya, di bilah navigasi konsol tersebut, pilih Layanan. Lalu pilih IAM.
3. Pilih Grup.
4. Pilih nama grup.
5. Pada tab Izin, untuk Kebijakan Terkelola, pilih Lampirkan Kebijakan.
6. Dalam daftar nama kebijakan, pilih salah satu kotak berikut ini.
   - AWSCloud9Pengguna (lebih disukai) atau AWSCloud9Administrator untuk memungkinkan setiap pengguna dalam grup menjadi pemilik lingkungan
   - AWSCloud9EnvironmentMember untuk mengaktifkan setiap pengguna dalam grup menjadi anggota saja

   (Jika Anda tidak melihat salah satu dari nama kebijakan ini dalam daftar, ketik nama kebijakan di kotak Pencarian untuk menampilkannya.)
7. Pilih Lampirkan kebijakan.

### Lampirkan kebijakan AWS terkelola AWS Cloud9 untuk grup menggunakan AWS CLI

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

Jalankan `attach-group-policy` perintah IAM untuk melampirkan kebijakan AWS terkelola AWS Cloud9 ke grup. Tentukan nama grup dan Nama Sumber Daya Amazon (ARN) kebijakan:

```
aws iam attach-group-policy --group-name MyGroup --policy-arn arn:aws:iam::aws:policy/
POLICY_NAME
```

Dengan perintah sebelumnya, ganti `MyGroup` dengan nama grup. Ganti `POLICY_NAME` dengan nama salah satu kebijakan AWS terkelola berikut.

- `AWSCloud9User` (lebih diutamakan) atau `AWSCloud9Administrator` untuk memungkinkan setiap pengguna dalam grup menjadi pemilik lingkungan
- `AWSCloud9EnvironmentMember` untuk mengaktifkan setiap pengguna dalam grup menjadi anggota saja

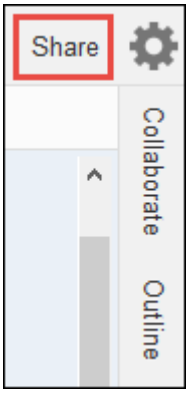
## Undang pengguna di akun yang sama dengan Lingkungan

Gunakan petunjuk di bagian ini untuk berbagi lingkungan AWS Cloud9 pengembangan yang Anda miliki Akun AWS dengan pengguna di akun yang sama.

1. Misalkan pengguna yang ingin Anda undang bukan salah satu dari jenis pengguna berikut. Pastikan pengguna yang ingin Anda undang sudah memiliki peran akses anggota lingkungan yang sesuai. Untuk instruksi, lihat Tentang Peran Akses Anggota Lingkungan.
   - Pengguna Akun AWS root.
   - Pengguna Administrator.

- Pengguna yang termasuk dalam grup IAM, pengguna yang mengambil peran, atau pengguna gabungan yang mengambil peran, dan grup atau peran tersebut memiliki kebijakan terkelola yang dilampirkan. AWS `AWSCloud9Administrator`

2. Buka lingkungan yang Anda miliki dan ingin mengundang pengguna, jika lingkungan belum terbuka.
3. Di bilah menu di AWS Cloud9 IDE, lakukan salah satu hal berikut.
   - Pilih Jendela, Bagikan.
   - Pilih Bagikan (terletak di sebelah ikon roda gigi Preferensi).




4. Dalam kotak dialog Bagikan lingkungan ini, untuk Undang Anggota, ketikkan salah satu dari yang berikut ini.
   - Untuk mengundang pengguna IAM, masukkan nama pengguna.
   - Untuk mengundang pengguna Akun AWS root, masukkan`arn:aws:iam::123456789012:root`. Ganti `123456789012` dengan Akun AWS ID Anda.
   - Untuk mengundang pengguna dengan peran yang diasumsikan atau pengguna federasi dengan peran yang diasumsikan, masukkan`arn:aws:sts::123456789012:assumed-role/MyAssumedRole/MyAssumedRoleSession`. Ganti `123456789012` dengan Akun AWS ID Anda, `MyAssumedRole` dengan nama peran yang diasumsikan. Ganti `MyAssumedRoleSession` dengan nama sesi untuk peran yang diasumsikan.
5. Untuk menjadikan pengguna ini sebagai anggota hanya-baca, pilih R. Untuk membuat pengguna ini membaca/menulis, pilih RW.
6. Pilih Undang.

> **Note**
>
> Jika Anda menjadikan pengguna ini anggota baca/tulis, kotak dialog akan ditampilkan, berisi informasi tentang kemungkinan menempatkan kredensi AWS keamanan Anda dalam risiko. Informasi berikut memberikan lebih banyak latar belakang tentang masalah ini.
>
> Anda harus berbagi lingkungan hanya dengan orang-orang yang Anda percaya. Anggota baca/tulis mungkin dapat menggunakan, kode AWS CLI AWS CloudShell, atau AWS SDK di lingkungan Anda untuk mengambil tindakan AWS atas nama Anda. Selain itu, jika Anda menyimpan kredensi AWS akses permanen Anda di lingkungan, anggota tersebut berpotensi menyalin kredensi tersebut dan menggunakannya di luar lingkungan. Menghapus kredensi AWS akses permanen Anda dari lingkungan Anda dan menggunakan kredenal AWS akses sementara sebagai gantinya tidak sepenuhnya mengatasi masalah ini. Ini mengurangi kesempatan anggota untuk menyalin kredensial sementara tersebut dan menggunakannya di luar lingkungan (karena kredensial sementara itu hanya akan berfungsi untuk waktu yang terbatas). Namun, kredensi sementara masih memungkinkan anggota baca/tulis untuk mengambil tindakan AWS dari lingkungan atas nama Anda.

7. Hubungi pengguna untuk memberi tahu mereka bahwa mereka dapat membuka lingkungan ini dan mulai menggunakannya.

## Minta AWS Cloud9 administrator di akun yang sama dengan Lingkungan mengundang dirinya sendiri atau orang lain

> **Note**
>
> Jika Anda menggunakan kredensi sementara AWS terkelola, Anda tidak dapat menggunakan sesi terminal di AWS Cloud9 IDE untuk menjalankan beberapa atau semua perintah di bagian ini. Untuk mengatasi praktik terbaik AWS keamanan, kredensi sementara AWS terkelola tidak mengizinkan beberapa perintah dijalankan. Sebagai gantinya, Anda dapat menjalankan perintah tersebut dari instalasi terpisah dari AWS Command Line Interface (AWS CLI).

Jenis pengguna berikut dapat mengundang diri mereka sendiri (atau pengguna lain yang sama Akun AWS) ke lingkungan apa pun di akun yang sama.

- Pengguna Akun AWS root.
- Pengguna administrator.
- Pengguna yang termasuk dalam grup IAM, pengguna yang mengambil peran, atau pengguna gabungan yang mengambil peran, dan grup atau peran tersebut memiliki kebijakan terkelola yang dilampirkan. AWS `AWSCloud9Administrator`

Misalkan pengguna yang diundang bukan salah satu jenis pengguna sebelumnya. Pastikan bahwa pengguna sudah memiliki peran akses anggota lingkungan yang sesuai. Untuk instruksi, lihat Tentang Peran Akses Anggota Lingkungan.

Untuk mengundang pengguna, gunakan AWS CLI atau AWS CloudShell untuk menjalankan AWS Cloud9 `create-environment-membership` perintah.

```
aws cloud9 create-environment-membership --environment-id
 12a34567b8cd9012345ef67abcd890e1 --user-arn USER_ARN --permissions PERMISSION_LEVEL
```

Pada perintah sebelumnya, ganti `12a34567b8cd9012345ef67abcd890e1` dengan ID lingkungan. Ganti `PERMISSION_LEVEL` dengan `read-write` atau`read-only`. Dan, ganti `USER_ARN` dengan salah satu dari berikut ini:

- Untuk mengundang pengguna IAM, masukkan`arn:aws:iam::123456789012:user/MyUser`. Ganti `123456789012` dengan Akun AWS ID Anda dan ganti `MyUser` dengan nama pengguna.
- Untuk mengundang pengguna Akun AWS root, masukkan`arn:aws:iam::123456789012:root`. Ganti `123456789012` dengan Akun AWS ID Anda.
- Untuk mengundang pengguna dengan peran yang diasumsikan atau pengguna federasi dengan peran yang diasumsikan, masukkan`arn:aws:sts::123456789012:assumed-role/MyAssumedRole/MyAssumedRoleSession`. Ganti `123456789012` dengan Akun AWS ID Anda. Ganti `MyAssumedRole` dengan nama peran yang diasumsikan. Dan, ganti `MyAssumedRoleSession` dengan nama sesi untuk peran yang diasumsikan.

Misalnya, untuk mengundang pengguna Akun AWS root untuk ID akun `123456789012` ke lingkungan dengan ID `12a34567b8cd9012345ef67abcd890e1` sebagai anggota baca/tulis, jalankan perintah berikut.

```
aws cloud9 create-environment-membership --environment-id
 12a34567b8cd9012345ef67abcd890e1 --user-arn arn:aws:iam::123456789012:root --
permissions read-write
```

> **Note**
>
> Jika Anda menggunakan AWS CloudShell, hilangkan `aws` awalan dari perintah sebelumnya.

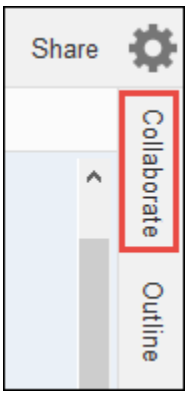
## Membuka Lingkungan bersama

Untuk membuka lingkungan bersama, Anda dapat menggunakan AWS Cloud9 dasbor Anda. Gunakan AWS Cloud9 IDE untuk melakukan tindakan dan menyelesaikan pekerjaan di lingkungan bersama. Contohnya bekerja dengan file dan mengobrol dengan anggota tim lainnya.

1. Pastikan kebijakan akses terkait dilampirkan ke grup atau peran untuk pengguna Anda. Untuk informasi selengkapnya, lihat Tentang Peran Akses Anggota Lingkungan.
2. Masuk ke AWS Cloud9 konsol sebagai berikut:
   - Jika Anda satu-satunya individu yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan Pusat Identitas IAM, lihat Akun AWS administrator untuk petunjuk masuk.
   - Jika Anda adalah seorang siswa di kelas, hubungi pengajar Anda untuk mendapatkan petunjuk masuk.
3. Buka lingkungan bersama dari AWS Cloud9 dasbor Anda. Untuk informasi selengkapnya, lihat Membuka Lingkungan di AWS Cloud9.

Anda menggunakan jendela Kolaborasi untuk berinteraksi dengan anggota lain, seperti yang dijelaskan di bagian lain dari topik ini.

> **Note**
>
> Jika jendela Kolaborasi tidak terlihat, pilih Kolaborasi. Jika tombol Kolaborasi tidak terlihat, pada bilah menu, pilih Jendela, Kolaborasi.

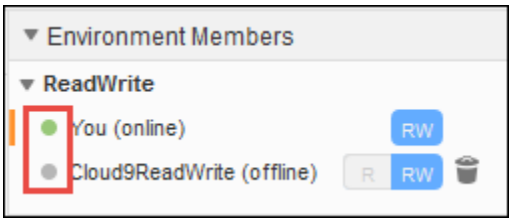



## Lihat daftar anggota lingkungan

Dengan lingkungan bersama yang terbuka, di jendela Kolaborasi, perluas Anggota Lingkungan, jika daftar anggota tidak terlihat.

Lingkaran di samping setiap anggota menunjukkan status online mereka, sebagai berikut:

- Anggota aktif memiliki lingkaran hijau.
- Anggota offline memiliki lingkaran abu-abu.
- Anggota siaga memiliki lingkaran oranye.




Untuk menggunakan kode untuk mendapatkan daftar anggota lingkungan, panggil operasi AWS Cloud9 menjelaskan keanggotaan lingkungan, sebagai berikut.

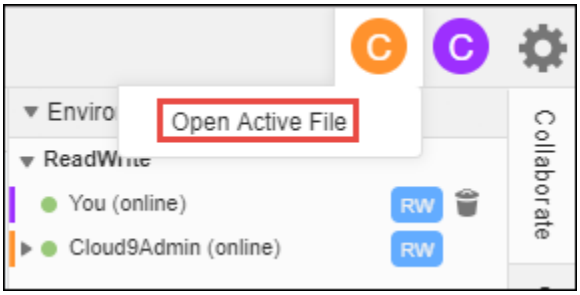
| | |
|---|---|
| AWS CLI | describe-environment-memberships |
| AWS SDK untuk C++ | DescribeEnvironmentMembershipsRequest, DescribeEnvironmentMembershipsResult |
| AWS SDK untuk Go | DescribeEnvironmentMemberships, DescribeEnvironmentMembershipsRequest, DescribeEnvironmentMembershipsWithContext |

| | |
|---|---|
| AWS SDK untuk Java | DescribeEnvironmentMembershipsRequest, DescribeEnvironmentMembershipsResult |
| AWS SDK untuk JavaScript | describeEnvironmentMemberships |
| AWS SDK untuk .NET | DescribeEnvironmentMembershipsRequest, DescribeEnvironmentMembershipsResponse |
| AWS SDK untuk PHP | describeEnvironmentMemberships |
| AWS SDK for Python (Boto) | describe_environment_memberships |
| AWS SDK untuk Ruby | describe_environment_memberships |
| AWS Tools for Windows PowerShell | Get-C9EnvironmentMembershipList |
| AWS Cloud9 API | DescribeEnvironmentMemberships |

## Membuka file aktif anggota lingkungan

Langkah ini menunjukkan bagaimana Anda dapat membuka file aktif anggota lingkungan.

Dengan lingkungan bersama terbuka, di bar menu, pilih nama anggota. Kemudian, pilih Buka File Aktif.

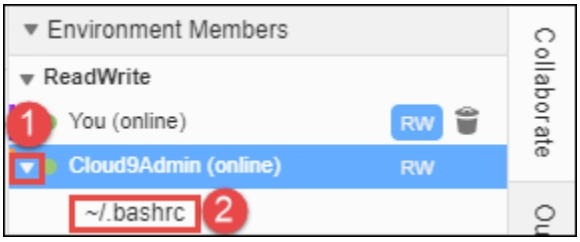



## Membuka file terbuka anggota lingkungan

Langkah ini menunjukkan bagaimana Anda dapat membuka file terbuka anggota lingkungan.

1. Dengan lingkungan bersama yang terbuka, di jendela Kolaborasi, perluas Anggota Lingkungan, jika daftar anggota tidak terlihat.
2. Perluas nama pengguna yang file terbuka yang ingin Anda buka di lingkungan Anda.

3. Buka (klik dua kali) nama file yang ingin Anda buka.




## Pergi ke kursor aktif anggota lingkungan

Langkah ini menunjukkan bagaimana Anda dapat menavigasi kursor aktif anggota lingkungan.

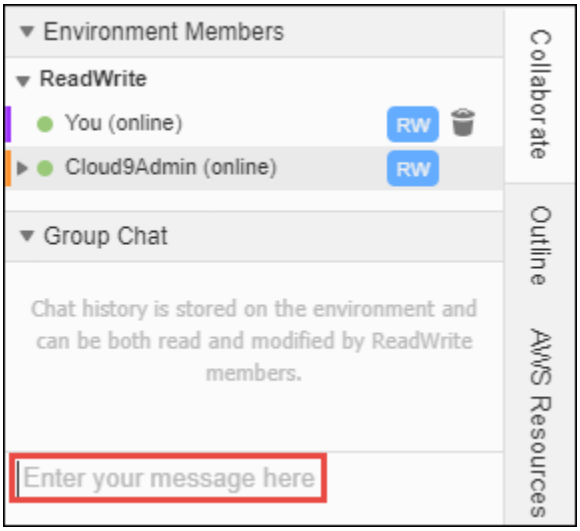
1. Dengan lingkungan bersama yang terbuka, di jendela Kolaborasi, perluas Anggota Lingkungan, jika daftar anggota tidak terlihat.
2. Buka menu konteks (klik kanan) untuk nama anggota, lalu pilih Tampilkan Lokasi.

## Mengelola obrolan di Lingkungan bersama

Topik ini menunjukkan bagaimana Anda dapat mengobrol dengan anggota lingkungan lain, melihat pesan obrolan di Lingkungan bersama, dan menghapusnya dari Lingkungan bersama.

### Mengobrol dengan anggota lingkungan lainnya

Dengan lingkungan bersama yang terbuka, di bagian bawah jendela Kolaborasi, untuk Masukkan pesan Anda di sini, masukkan pesan obrolan Anda, lalu tekan `Enter`.



### Melihat pesan chat di Lingkungan bersama

Dengan lingkungan bersama terbuka, di jendela Kolaborasi, perluas Obrolan Grup, jika daftar pesan obrolan tidak terlihat.

### Menghapus pesan obrolan dari Lingkungan bersama

Dengan lingkungan bersama terbuka, di jendela Kolaborasi, buka menu konteks (klik kanan) untuk pesan obrolan di Obrolan Grup. Kemudian, pilih Hapus Pesan.

> **Note**
>
> Bila Anda menghapus pesan chat, pesan tersebut akan dihapus dari lingkungan untuk semua anggota.

### Menghapus pesan chat dari Lingkungan bersama

Dengan lingkungan bersama terbuka, di jendela Kolaborasi, buka menu konteks (klik kanan) di mana saja di Obrolan Grup. Kemudian, pilih Hapus riwayat.

> **Note**
>
> Saat Anda menghapus semua pesan obrolan, pesan tersebut akan dihapus dari lingkungan untuk semua anggota.

## Mengubah peran akses anggota lingkungan

Langkah ini menunjukkan bagaimana Anda dapat mengubah peran akses anggota lingkungan. Anda juga dapat menggunakan kode untuk mengubah peran akses dan memperbarui keanggotaan AWS Cloud9 lingkungan.

1. Buka lingkungan yang Anda miliki dan yang berisi anggota yang peran aksesnya ingin Anda ubah, jika lingkungan belum terbuka. Untuk informasi selengkapnya, lihat Membuka Lingkungan di AWS Cloud9.
2. Jika daftar anggota tidak terlihat, perluas Anggota Lingkungan di jendela Kolaborasi.
3. Lakukan salah satu tindakan berikut:

- Di samping nama anggota yang peran aksesnya ingin Anda ubah, pilih R atau RW untuk menjadikan pemilik anggota ini atau baca/tulis, masing-masing.
- Untuk mengubah anggota baca/tulis menjadi hanya-baca, buka menu konteks (klik kanan) untuk nama anggota, lalu pilih Batalkan Akses Tulis.
- Untuk mengubah anggota hanya-baca menjadi baca/tulis, buka menu konteks (klik kanan) untuk nama anggota, lalu pilih Berikan Akses Baca+Tulis.

> **Note**
>
> Jika Anda menjadikan pengguna ini anggota baca/tulis, kotak dialog akan ditampilkan, berisi informasi tentang kemungkinan menempatkan kredensi AWS keamanan Anda dalam risiko. Kecuali Anda mempercayai pengguna tersebut untuk mengambil tindakan atas nama Anda, jangan menjadikan pengguna sebagai anggota baca/tulis. AWS Untuk informasi selengkapnya, lihat catatan terkait di Undang Pengguna di Akun yang Sama dengan Lingkungan.

Untuk menggunakan kode untuk mengubah peran akses anggota lingkungan, panggil operasi keanggotaan lingkungan AWS Cloud9 pembaruan, sebagai berikut.

| | |
|---|---|
| AWS CLI | update-environment-membership |
| AWS SDK untuk C++ | UpdateEnvironmentMembershipRequest, UpdateEnvironmentMembershipResult |
| AWS SDK untuk Go | UpdateEnvironmentMembership, UpdateEnvironmentMembershipRequest, UpdateEnvironmentMembershipWithContext |
| AWS SDK untuk Java | UpdateEnvironmentMembershipRequest, UpdateEnvironmentMembershipResult |
| AWS SDK untuk JavaScript | updateEnvironmentMembership |
| AWS SDK untuk .NET | UpdateEnvironmentMembershipRequest, UpdateEnvironmentMembershipResponse |
| AWS SDK untuk PHP | updateEnvironmentMembership |

| | |
|---|---|
| AWS SDK for Python (Boto) | update_environment_membership |
| AWS SDK untuk Ruby | update_environment_membership |
| AWS Tools for Windows PowerShell | Update-C9EnvironmentMembership |
| AWS Cloud9 API | UpdateEnvironmentMembership |

## Hapus pengguna Anda dari Lingkungan bersama

Langkah ini menunjukkan bagaimana Anda dapat menghapus pengguna dari lingkungan bersama.

> **Note**
>
> Jika Anda pemilik lingkungan, Anda tidak dapat menghapus pengguna dari lingkungan. Tindakan menghapus pengguna Anda dari lingkungan tidak menghapus pengguna Anda dari IAM.

1. Dengan lingkungan bersama yang terbuka, di jendela Kolaborasi, perluas Anggota Lingkungan, jika daftar anggota tidak terlihat.
2. Lakukan salah satu tindakan berikut:
   - Di sebelah Anda, pilih ikon tempat sampah.
   - Buka menu konteks (klik kanan) untuk Anda, lalu pilih Tinggalkan lingkungan.
3. Saat diminta, pilih Tinggalkan.

Untuk menggunakan kode untuk menghapus pengguna Anda dari lingkungan bersama, panggil operasi keanggotaan lingkungan AWS Cloud9 hapus, sebagai berikut.

| | |
|---|---|
| AWS CLI | delete-environment-membership |
| AWS SDK untuk C++ | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResult |

| | |
|---|---|
| AWS SDK untuk Go | DeleteEnvironmentMembership, DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipWithContext |
| AWS SDK untuk Java | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResult |
| AWS SDK untuk JavaScript | deleteEnvironmentMembership |
| AWS SDK untuk .NET | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResponse |
| AWS SDK untuk PHP | deleteEnvironmentMembership |
| AWS SDK for Python (Boto) | delete_environment_membership |
| AWS SDK untuk Ruby | delete_environment_membership |
| AWS Tools for Windows PowerShell | Remove-C9EnvironmentMembership |
| AWS Cloud9 API | DeleteEnvironmentMembership |

## Hapus anggota lingkungan lain

Langkah ini menunjukkan bagaimana Anda dapat menghapus anggota selain pengguna Anda dari lingkungan.

> **Note**
>
> Untuk menghapus anggota selain pengguna Anda dari lingkungan, Anda harus masuk AWS Cloud9 dengan menggunakan kredensi pemilik lingkungan.
> Menghapus anggota tidak menghapus pengguna dari IAM.

1. Buka lingkungan yang berisi anggota yang ingin Anda hapus, jika lingkungan belum terbuka. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.
2. Di jendela Kolaborasi, perluas Anggota Lingkungan, jika daftar anggota tidak terlihat.
3. Lakukan salah satu hal berikut ini:

- Di samping nama anggota yang ingin Anda hapus, pilih ikon tempat sampah.
- Buka menu konteks (klik kanan) untuk nama anggota yang ingin Anda hapus, lalu pilih Cabut Akses.

4. Saat diminta, pilih Hapus Anggota.

Untuk menggunakan kode untuk menghapus anggota dari lingkungan, panggil operasi keanggotaan lingkungan AWS Cloud9 hapus, sebagai berikut.

| | |
|---|---|
| AWS CLI | delete-environment-membership |
| AWS SDK untuk C++ | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResult |
| AWS SDK untuk Go | DeleteEnvironmentMembership, DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipWithContext |
| AWS SDK untuk Java | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResult |
| AWS SDK untuk JavaScript | deleteEnvironmentMembership |
| AWS SDK untuk .NET | DeleteEnvironmentMembershipRequest, DeleteEnvironmentMembershipResponse |
| AWS SDK untuk PHP | deleteEnvironmentMembership |
| AWS SDK for Python (Boto) | delete_environment_membership |
| AWS SDK untuk Ruby | delete_environment_membership |
| AWS Tools for Windows PowerShell | Remove-C9EnvironmentMembership |
| AWS Cloud9 API | DeleteEnvironmentMembership |

## Praktik terbaik berbagi lingkungan

Kami merekomendasikan praktik berikut saat berbagi lingkungan:

- Hanya undang anggota baca/tulis yang Anda percayai ke lingkungan Anda.
- Untuk EC2 lingkungan, anggota baca/tulis dapat menggunakan kredenal AWS akses pemilik lingkungan untuk melakukan panggilan dari lingkungan ke. Layanan AWS Ini bukan kredensialnya sendiri. Untuk mencegah hal ini, pemilik lingkungan dapat menonaktifkan kredensi sementara AWS terkelola untuk lingkungan. Namun, ini juga mencegah pemilik lingkungan melakukan panggilan. Untuk informasi selengkapnya, lihat Kredensial Sementara terkelola AWS.
- Aktifkan AWS CloudTrail untuk melacak aktivitas di lingkungan Anda. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS CloudTrail.
- Jangan gunakan pengguna Akun AWS root Anda untuk membuat dan berbagi lingkungan. Gunakan pengguna IAM di akun sebagai gantinya. Untuk informasi selengkapnya, lihat Hanya Akses Pertama Kali: Kredensial Pengguna Root Anda dan pengguna IAM di Panduan Pengguna IAM.

# Memindahkan AWS Cloud9 IDE dari volume Amazon EBS

Anda dapat memindahkan lingkungan AWS Cloud9 pengembangan dari satu EC2 instans Amazon ke instans Amazon lainnya. Misalnya, Anda mungkin ingin melakukan tindakan berikut:

- Transfer lingkungan dari EC2 instans Amazon yang terganggu atau berkinerja dengan cara yang tidak terduga dibandingkan dengan instance sehat.
- Mentransfer lingkungan dari instans yang sudah ada ke lingkungan yang memiliki pembaruan sistem terbaru.
- Menambah atau mengurangi sumber daya komputasi instans karena lingkungan terlalu banyak digunakan atau kurang digunakan pada instans saat ini.

Anda dapat memutakhirkan dari satu AMI yang AWS Cloud9 didukung ke AMI lainnya dengan bermigrasi ke AWS Cloud9 EC2 lingkungan baru, sambil menyimpan file proyek. Anda mungkin ingin meningkatkan ke versi lain dari AMI karena:

- AMI dari lingkungan saat ini telah mencapai end-of-life dan tidak lagi didukung.
- Paket yang Anda butuhkan sudah usang di AMI saat ini.

Anda juga dapat mengubah ukuran volume Amazon Elastic Block Store (Amazon EBS) yang terkait dengan instans Amazon EC2 untuk suatu lingkungan. Misalnya, Anda mungkin ingin melakukan salah satu atau kedua tindakan berikut:

- Tingkatkan ukuran volume karena Anda kehabisan ruang penyimpanan pada instans.
- Kurangi ukuran volume karena Anda tidak ingin membayar ruang penyimpanan tambahan yang tidak digunakan.

Sebelum Anda memindahkan atau mengubah ukuran lingkungan, Anda dapat mencoba menghentikan beberapa proses yang berjalan di lingkungan atau menambahkan file swap ke lingkungan. Untuk informasi selengkapnya tentang menangani memori rendah atau penggunaan CPU yang tinggi, lihat Pemecahan Masalah.

> **Note**
>
> Topik ini hanya menjelaskan pemindahan lingkungan dari satu EC2 instans Amazon ke instans Amazon lainnya atau mengubah ukuran volume Amazon EBS. Untuk mengubah ukuran lingkungan dari salah satu server Anda sendiri atau untuk mengubah ruang penyimpanan untuk salah satu server Anda sendiri, lihat dokumentasi server Anda.

Terakhir, Anda dapat mengenkripsi sumber daya Amazon EBS untuk memastikan keamanan keduanya data-at-rest dan data-in-transit antara instans dan penyimpanan EBS terlampir.

## Memindahkan sebuah lingkungan

Sebelum Anda memulai proses pemindahan, perhatikan kondisi berikut:

- Anda tidak dapat memindahkan lingkungan ke EC2 instance Amazon dengan jenis yang sama. Ketika Anda pindah, Anda harus memilih jenis EC2 instans Amazon yang berbeda untuk instans baru.

> **Important**
>
> Jika Anda memindahkan lingkungan ke jenis EC2 instans Amazon lainnya, jenis instans tersebut juga harus didukung oleh AWS Cloud9 saat ini Wilayah AWS. Untuk memeriksa jenis instance yang tersedia di setiap Wilayah, buka halaman Konfigurasi pengaturan yang ditampilkan saat membuat EC2 lingkungan dengan konsol. Pilihan Anda di bagian Jenis instans ditentukan oleh Wilayah AWS yang dipilih di kanan atas konsol.

- Anda harus menghentikan EC2 instans Amazon yang terkait dengan lingkungan sebelum Anda dapat mengubah jenis instans. Saat instance dihentikan, Anda dan anggota mana pun tidak dapat menggunakan lingkungan yang terkait dengan instance yang dihentikan.
- AWS memindahkan instance ke perangkat keras baru, namun, ID instance tidak berubah.
- Jika instance berjalan di VPC Amazon dan memiliki IPv4 alamat publik, lepaskan AWS alamat dan berikan instance alamat publik IPv4 baru. Instans mempertahankan IPv4 alamat pribadinya dan alamat atau IPv6 alamat IP Elastis apa pun.
- Rencanakan waktu henti saat instans Anda dihentikan. Prosesnya mungkin memakan waktu beberapa menit.

Untuk memindahkan sebuah lingkungan

1. (Opsional) Jika jenis instans baru memerlukan driver yang tidak diinstal pada instance yang ada, sambungkan ke instance Anda dan instal driver tersebut. Untuk informasi selengkapnya, lihat Kompatibilitas untuk mengubah ukuran instans di EC2 Panduan Pengguna Amazon.
2. Tutup semua tab peramban web yang saat ini menampilkan lingkungan.

   > **Important**
   >
   > Jika Anda tidak menutup semua tab browser web yang saat ini menampilkan lingkungan, AWS Cloud9 mungkin mengganggu penyelesaian prosedur ini. Secara khusus, AWS Cloud9 mungkin mencoba pada waktu yang salah selama prosedur ini untuk memulai ulang EC2 instance Amazon yang terkait dengan lingkungan. Instans harus tetap berhenti hingga langkah terakhir dalam prosedur ini.

3. Masuk ke AWS Management Console, jika Anda belum masuk, di https://console.aws.amazon.com.

   Kami menyarankan Anda masuk menggunakan kredensyal tingkat administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.
4. Buka EC2 konsol Amazon. Untuk melakukan ini, dalam daftar Layanan, pilih EC2.
5. Di bilah AWS navigasi, pilih Wilayah AWS yang berisi lingkungan yang ingin Anda pindahkan (misalnya, US East (Ohio)).
6. Di panel navigasi layanan, perluas Instans, lalu pilih Instans.
7. Dalam daftar instans, pilih salah satu yang terkait dengan lingkungan yang ingin Anda pindahkan. Untuk EC2 lingkungan, nama instance dimulai dengan `aws-cloud9-` diikuti oleh

nama lingkungan. Misalnya, jika lingkungan diberi nama`my-demo-environment`, nama instance dimulai dengan`aws-cloud9-my-demo-environment`.

8. Jika Status Instance tidak Dihentikan, pilih Actions, Instance state, Stop. Saat diminta, pilih Ya, Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
9. Setelah Status Instans dihentikan, dengan instans masih dipilih, pilih Tindakan, Pengaturan Instans, Ubah Tipe Instans.
10. Di kotak dialog Ubah Tipe Instans pilih tipe instans yang baru untuk lingkungan yang akan digunakan.

> **Note**
>
> Jika jenis instance yang Anda inginkan tidak muncul dalam daftar, itu tidak kompatibel dengan konfigurasi instance. Misalnya, instance mungkin tidak kompatibel karena tipe virtualisasi.

11. (Opsional) Jika tipe instans yang Anda pilih mendukung pengoptimalan EBS, pilih EBS–dioptimalkan untuk mengaktifkan pengoptimalan EBS, atau hapus EBS–dioptimalkan untuk menonaktifkan pengoptimalan EBS.

> **Note**
>
> Jika tipe instans yang Anda pilih adalah EBS–dioptimalkan secara default, EBS-dioptimalkan dipilih dan Anda tidak dapat menghapusnya.

12. Pilih Terapkan untuk menerima pengaturan baru.

> **Note**
>
> Jika Anda tidak memilih jenis instans yang berbeda untuk Tipe Instans sebelumnya dalam prosedur ini, tidak ada yang terjadi setelah Anda memilih Terapkan.

13. Buka kembali lingkungan. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

Untuk informasi selengkapnya tentang prosedur sebelumnya, lihat Mengubah jenis instans di EC2 Panduan Pengguna Amazon.

# Memindahkan AWS Cloud9 EC2 lingkungan ke Amazon Machine Image (AMI) yang berbeda

Topik ini menjelaskan cara memigrasikan AWS Cloud9 EC2 lingkungan dari satu AMI Amazon Linux ke AMI lain yang AWS Cloud9 didukung.

> **Note**
>
> Jika Anda ingin memindahkan lingkungan Anda ke instance baru tanpa memperbarui versi OS, lihatthe section called “Memindahkan sebuah lingkungan”.

Anda dapat memigrasikan data antar lingkungan menggunakan salah satu prosedur berikut:

Untuk memindahkan lingkungan dengan mengunduh arsip ke mesin lokal

1. Buat lingkungan baru di Availability Zone yang sama dengan gambar dasar yang berbeda:
   a. Selesaikan langkah-langkah di the section called “Menciptakan EC2 Lingkungan” bagian untuk membuat lingkungan baru.

      > **Note**
      >
      > Saat memilih Platform, pilih platform tempat Anda ingin memigrasikan lingkungan Anda.

   b. Secara default, lingkungan dibuat dengan volume 10 GiB. Jika Anda tidak memiliki cukup ruang untuk mengunggah atau membongkar arsip ke lingkungan baru, selesaikan langkah-langkah dalam the section called “Ubah ukuran volume Amazon EBS” prosedur untuk mengubah ukuran volume Amazon EBS.
2. Buka lingkungan yang ingin Anda migrasikan di AWS Cloud9 IDE.
3. Setelah AWS Cloud9 IDE dimuat, pilih File > Unduh proyek dari menu untuk mengunduh arsip dengan isi direktori proyek lingkungan.
4. Buka AWS Cloud9 IDE di lingkungan baru.
5. Pilih File> Unggah file lokal... untuk mengunggah arsip.
6. (Opsional) Untuk mencadangkan `.c9` direktori lama ke`.c9.backup`, di terminal lingkungan, jalankan perintah berikut:

```
cp .c9 .c9.backup
```

Anda mungkin memerlukan file cadangan ini jika Anda ingin mengembalikan file konfigurasi nanti.

7. Untuk membongkar arsip, jalankan perintah berikut:

```
tar xzvf <old_environment_name>.tar.gz -C ~/
```

8. Untuk menghapus arsip dari direktori proyek, jalankan perintah berikut:

```
rm <old_environment_name>.tar.gz
```

Pastikan lingkungan baru berfungsi seperti yang diharapkan.

9. Anda sekarang dapat menghapus lingkungan lama.

Untuk memindahkan lingkungan menggunakan volume Amazon EBS

Jika Anda tidak dapat mengunduh arsip, atau jika arsip yang dihasilkan terlalu besar, Anda dapat menggunakan volume Amazon EBS untuk bermigrasi. Juga, metode ini memungkinkan Anda untuk menyalin file yang terletak di luar `~/environment` direktori.

1. Tutup semua tab AWS Cloud9 IDE yang terbuka di lingkungan yang ada.
2. Selesaikan langkah-langkah berikut untuk menghentikan instance yang ada:
   a. Di AWS Cloud9 konsol, pilih lingkungan yang akan dinavigasi untuk melihat detailnya.
   b. Pada halaman Environment details, di bawah tab EC2instance, pilih Manage EC2 instance.
   c. Di EC2 konsol, pilih instance untuk menavigasi ke detail instance.
   d. Pastikan bahwa status Instance diatur ke Berhenti. Jika tidak, pilih Stop instance dari daftar dropdown status Instance. Saat diminta, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
3. Buat lingkungan baru di Availability Zone yang sama dengan gambar dasar yang berbeda:
   a. Selesaikan langkah-langkah di the section called “Menciptakan EC2 Lingkungan” bagian untuk membuat lingkungan baru.

> **Note**
>
> Saat memilih Platform, pilih platform tempat Anda ingin memigrasikan lingkungan Anda.

b. Secara default, lingkungan dibuat dengan volume 10 GiB. Jika Anda tidak memiliki cukup ruang untuk memindahkan file dari volume sumber ke lingkungan baru, selesaikan langkah-langkah dalam the section called “Ubah ukuran volume Amazon EBS” prosedur untuk mengubah ukuran volume Amazon EBS.

4. Selesaikan langkah-langkah berikut untuk melepaskan volume dari instance yang ada:

   a. Pada halaman ringkasan Instance, pilih tab Storage dan pilih volume. Nama perangkat dari volume yang dipilih harus sama dengan yang ditentukan dalam nama perangkat Root dari bagian detail perangkat Root.

   b. Pada halaman detail volume, pilih Tindakan > Lepaskan volume.

   c. Setelah volume berhasil dilepas, pilih Actions > Attach volume lalu temukan dan pilih instance lingkungan baru dari daftar dropdown. Nama EC2 instans Amazon yang Anda pilih harus berisi nama AWS Cloud9 lingkungan yang diawali`aws-cloud9`.

5. Buka AWS Cloud9 IDE di lingkungan baru.

6. Setelah lingkungan dimuat, untuk mengidentifikasi perangkat volume yang baru terpasang, jalankan perintah berikut di terminal:

```
lsblk
```

Dalam output sampel berikut, `nvme0n1` partisi perangkat root `nvme0n1p1` sudah terpasang, maka `nvme1n1p1` partisi juga harus dipasang. Jalur lengkap untuk perangkatnya adalah`/dev/nvme1n1p1`:

```
Admin:~/environment $ lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
nvme0n1       259:0    0  10G  0 disk
##nvme0n1p1   259:2    0  10G  0 part /
##nvme0n1p127 259:3    0   1M  0 part
##nvme0n1p128 259:4    0  10M  0 part /boot/efi
nvme1n1       259:1    0  10G  0 disk
##nvme1n1p1   259:5    0  10G  0 part
```

```
##nvme1n1p128 259:6    0   1M  0 part
```

> **Note**
>
> Outputnya bervariasi saat Anda menjalankan perintah ini di terminal Anda.

7. Selesaikan langkah-langkah berikut di terminal lingkungan untuk me-mount volume yang ada:

   a. Untuk membuat direktori sementara di mana partisi volume akan dipasang, jalankan perintah berikut:

   ```
   MOUNT_POINT=$(mktemp -d)
   ```

   b. Berdasarkan output sampel `lsblk` perintah, tentukan jalur perangkat berikut yang akan dipasang:

   ```
   MOUNT_DEVICE=/dev/nvme1n1p1
   ```

   > **Note**
   >
   > Outputnya bervariasi saat Anda menjalankan perintah ini di terminal Anda.

   c. Untuk me-mount volume yang ada, jalankan perintah berikut:

   ```
   sudo mount $MOUNT_DEVICE $MOUNT_POINT
   ```

   d. Selesaikan langkah-langkah berikut untuk memverifikasi apakah volume yang ada terpasang dengan benar:

      i. Untuk memastikan bahwa volume termasuk dalam output, jalankan perintah berikut:

      ```
      df -h
      ```

      ii. Untuk memverifikasi isi volume, jalankan perintah berikut:

      ```
      ls $MOUNT_POINT/home/ec2-user/environment/
      ```

8. (Opsional) Untuk mencadangkan `.c9` direktori lama ke`.c9.backup`, di terminal lingkungan, jalankan perintah berikut:

```
cp .c9 .c9.backup
```

Anda mungkin memerlukan file cadangan ini jika Anda ingin mengembalikan file konfigurasi nanti.

9. Untuk menyalin lingkungan lama dari volume yang ada, jalankan perintah berikut:

```
cp -R $MOUNT_POINT/home/ec2-user/environment ~
```

> **Note**
>
> Jika diperlukan, Anda juga dapat menyalin file atau direktori di luar direktori lingkungan menggunakan perintah sebelumnya.

Pastikan lingkungan baru berfungsi seperti yang diharapkan.

10. Untuk melepas perangkat sebelumnya, jalankan salah satu dari dua perintah berikut:

```
sudo umount $MOUNT_DEVICE
```

```
sudo umount $MOUNT_POINT
```

11. Pilih Lepaskan volume dari daftar dropdown Tindakan untuk melepaskan volume yang Anda lampirkan pada Langkah 3.
12. Anda sekarang dapat menghapus lingkungan lama dan volumenya.

> **Note**
>
> Karena volume tidak lagi dilampirkan ke EC2 instans Amazon lingkungan, Anda harus menghapusnya secara manual. Anda dapat melakukan ini dengan memilih Hapus pada halaman Detail volume.

## Ubah ukuran volume Amazon EBS yang digunakan lingkungan

Langkah ini menunjukkan bagaimana Anda dapat mengubah ukuran volume Amazon EBS.

1. Buka lingkungan yang terkait dengan EC2 instans Amazon untuk volume Amazon EBS yang ingin Anda ubah ukurannya.
2. Di AWS Cloud9 IDE untuk lingkungan, buat file dengan konten berikut, lalu simpan file dengan ekstensi `.sh` (misalnya,`resize.sh`).

> **Catatan**
>
> Skrip ini berfungsi untuk volume Amazon EBS yang terhubung ke EC2 instans yang menjalankan AL2 023, Amazon Linux 2, Amazon Linux, atau Ubuntu Server dan dikonfigurasi untuk digunakan IMDSv2.
>
> Skrip ini juga mengubah ukuran volume Amazon EBS yang diekspos sebagai perangkat NVMe blok Nitrocontoh berbasis. Untuk daftar contoh berdasarkan sistem Nitro, lihat Nitroinstance berbasis di Panduan EC2 Pengguna Amazon.

```
#!/bin/bash

# Specify the desired volume size in GiB as a command line argument. If not
 specified, default to 20 GiB.
SIZE=${1:-20}

# Get the ID of the environment host Amazon EC2 instance.
TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 60")
INSTANCEID=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" -v
 http://169.254.169.254/latest/meta-data/instance-id 2> /dev/null)
REGION=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/
latest/meta-data/placement/region 2> /dev/null)

# Get the ID of the Amazon EBS volume associated with the instance.
VOLUMEID=$(aws ec2 describe-instances \
  --instance-id $INSTANCEID \
  --query "Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.VolumeId" \
  --output text \
  --region $REGION)

# Resize the EBS volume.
aws ec2 modify-volume --volume-id $VOLUMEID --size $SIZE
```

```
# Wait for the resize to finish.
while [ \
  "$(aws ec2 describe-volumes-modifications \
    --volume-id $VOLUMEID \
    --filters Name=modification-state,Values="optimizing","completed" \
    --query "length(VolumesModifications)"\
    --output text)" != "1" ]; do
sleep 1
done

# Check if we're on an NVMe filesystem
if [[ -e "/dev/xvda" && $(readlink -f /dev/xvda) = "/dev/xvda" ]]
then
# Rewrite the partition table so that the partition takes up all the space that it
 can.
  sudo growpart /dev/xvda 1
# Expand the size of the file system.
# Check if we're on AL2 or AL2023
  STR=$(cat /etc/os-release)
  SUBAL2="VERSION_ID=\"2\""
  SUBAL2023="VERSION_ID=\"2023\""
  if [[ "$STR" == *"$SUBAL2"* || "$STR" == *"$SUBAL2023"* ]]
  then
    sudo xfs_growfs -d /
  else
    sudo resize2fs /dev/xvda1
  fi

else
# Rewrite the partition table so that the partition takes up all the space that it
 can.
  sudo growpart /dev/nvme0n1 1

# Expand the size of the file system.
# Check if we're on AL2 or AL2023
  STR=$(cat /etc/os-release)
  SUBAL2="VERSION_ID=\"2\""
  SUBAL2023="VERSION_ID=\"2023\""
  if [[ "$STR" == *"$SUBAL2"* || "$STR" == *"$SUBAL2023"* ]]
  then
    sudo xfs_growfs -d /
  else
    sudo resize2fs /dev/nvme0n1p1
  fi
```

```
fi
```

3. Dari sesi terminal di IDE, pindah ke direktori yang berisi file `resize.sh`. Kemudian jalankan salah satu dari perintah berikut, ganti `20` dengan ukuran di GiB yang ingin Anda ubah ukuran volume Amazon EBS menjadi:

   - ```
     bash resize.sh 20
     ```
   - ```
     chmod +x resize.sh
     ./resize.sh 20
     ```

## Enkripsi volume Amazon EBS yang menggunakan AWS Cloud9

Topik ini menunjukkan cara mengenkripsi volume Amazon EBS tuntuk EC2 instans yang digunakan oleh lingkungan pengembangan. AWS Cloud9

Enkripsi Amazon EBS mengenkripsi data berikut:

- Data at rest di dalam volume
- Semua data yang bergerak antara volume dan instance
- Semua snapshot yang dibuat dari volume
- Semua volume yang dibuat dari snapshot tersebut

Anda memiliki dua opsi enkripsi untuk volume Amazon EBS yang digunakan oleh lingkungan AWS Cloud9 EC2 pengembangan:

- Enkripsi secara default — Anda dapat mengonfigurasi Akun AWS untuk menerapkan enkripsi volume EBS baru dan salinan snapshot yang Anda buat. Enkripsi secara default diaktifkan pada tingkat file Wilayah AWS. Jadi, Anda tidak dapat mengaktifkannya untuk volume atau snapshot individual di Wilayah itu. Selain itu, Amazon EBS mengenkripsi volume yang dibuat saat Anda meluncurkan instans. Jadi, Anda harus mengaktifkan pengaturan ini sebelum membuat EC2 lingkungan. Untuk informasi selengkapnya, lihat Enkripsi secara default di Panduan EC2 Pengguna Amazon.
- Enkripsi volume Amazon EBS yang ada yang digunakan oleh EC2 lingkungan — Anda dapat mengenkripsi volume Amazon EBS tertentu yang sudah dibuat untuk instans. EC2 Opsi ini melibatkan penggunaan AWS Key Management Service (AWS KMS) untuk mengelola akses ke

volume terenkripsi. Untuk prosedur yang relevan, lihat Enkripsi volume Amazon EBS yang ada yang menggunakan AWS Cloud9.

**Important**

Jika AWS Cloud9 IDE Anda menggunakan volume Amazon EBS yang dienkripsi secara default, peran AWS Identity and Access Management terkait layanan AWS Cloud9 memerlukan akses ke volume EBS ini AWS KMS key . Jika akses tidak disediakan, AWS Cloud9 IDE mungkin gagal diluncurkan dan debugging mungkin sulit.
Untuk menyediakan akses, tambahkan peran terkait layanan untuk AWS Cloud9,`AWSServiceRoleForAWSCloud9`, ke kunci KMS yang digunakan oleh volume Amazon EBS Anda. Untuk informasi selengkapnya tentang tugas ini, lihat Membuat AWS Cloud9 IDE yang menggunakan volume Amazon EBS dengan enkripsi default di Pola Panduan AWS Preskriptif.

## Enkripsi volume Amazon EBS yang ada yang menggunakan AWS Cloud9

Mengenkripsi volume Amazon EBS yang ada melibatkan penggunaan AWS KMS untuk membuat kunci KMS. Setelah Anda membuat snapshot volume untuk diganti, Anda menggunakan tombol KMS untuk mengenkripsi salinan snapshot.

Selanjutnya, Anda membuat volume terenkripsi dengan snapshot itu. Kemudian, Anda mengganti volume yang tidak terenkripsi dengan melepaskannya dari EC2 instance dan melampirkan volume terenkripsi.

Terakhir, Anda harus memperbarui kebijakan kunci untuk kunci yang dikelola pelanggan untuk mengaktifkan akses untuk peran AWS Cloud9 layanan.

**Note**

Prosedur berikut berfokus pada penggunaan kunci yang dikelola pelanggan untuk mengenkripsi volume. Anda juga dapat menggunakan Kunci yang dikelola AWS for an Layanan AWS di akun Anda. Alias untuk Amazon EBS adalah. `aws/ebs` Jika Anda memilih opsi default ini untuk enkripsi, lewati langkah 1 di mana Anda membuat kunci terkelola pelanggan. Juga, lewati langkah 8 di mana Anda memperbarui kebijakan utama. Ini karena Anda tidak dapat mengubah kebijakan kunci untuk file Kunci yang dikelola AWS.

Untuk mengenkripsi volume Amazon EBS yang ada

1. Di AWS KMS konsol, buat kunci KMS simetris. Untuk informasi selengkapnya, lihat Membuat kunci KMS simetris di Panduan AWS Key Management Service Pengembang.
2. Di EC2 konsol Amazon, hentikan instans yang didukung Amazon EBS yang digunakan oleh lingkungan. Anda dapat menghentikan instans menggunakan konsol tersebut atau baris perintah.
3. Di panel navigasi EC2 konsol Amazon, pilih Snapshots untuk membuat snapshot dari volume yang ada yang ingin Anda enkripsi.
4. Di panel navigasi EC2 konsol Amazon, pilih Snapshots untuk menyalin snapshot. Di kotak dialog Salin snapshot, lakukan hal berikut untuk mengaktifkan enkripsi:
   - Pilih Enkripsi snapshot ini.
   - Untuk Master Key, pilih tombol KMS yang Anda buat sebelumnya. (Jika Anda menggunakan Kunci yang dikelola AWS, pertahankan pengaturan aws/ebs (default).)
5. Buat volume baru dari snapshot terenkripsi.

   > **Note**
   >
   > Volume Amazon EBS baru yang dibuat dari snapshot terenkripsi secara otomatis dienkripsi.

6. Lepaskan volume Amazon EBS lama dari instans Amazon EC2 .
7. Lampirkan volume terenkripsi baru ke instans Amazon EC2 .
8. Perbarui kebijakan kunci untuk kunci KMS menggunakan tampilan AWS Management Console default, tampilan AWS Management Console kebijakan, atau AWS KMS API. Tambahkan pernyataan kebijakan kunci berikut untuk mengizinkan AWS Cloud9 layanan,`AWSServiceRoleForAWSCloud9`, untuk mengakses kunci KMS.

   > **Note**
   >
   > Jika Anda menggunakan Kunci yang dikelola AWS, lewati langkah ini.

```
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
```

```
    "Principal": {
        "AWS": "arn:{Partition}:iam::{AccountId}:role/aws-service-role/
cloud9.amazonaws.com/AWSServiceRoleForAWSCloud9"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
   },
   {
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:{Partition}:iam::{AccountId}:role/aws-service-role/
cloud9.amazonaws.com/AWSServiceRoleForAWSCloud9"
    },
    "Action": [
        "kms:CreateGrant",
        "kms:ListGrants",
        "kms:RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": "true"
        }
    }
}
```

9. Mulai ulang EC2 instance Amazon. Untuk informasi selengkapnya tentang memulai ulang EC2 instans Amazon, lihat Berhenti dan mulai instans Anda.

# Menghapus lingkungan di AWS Cloud9

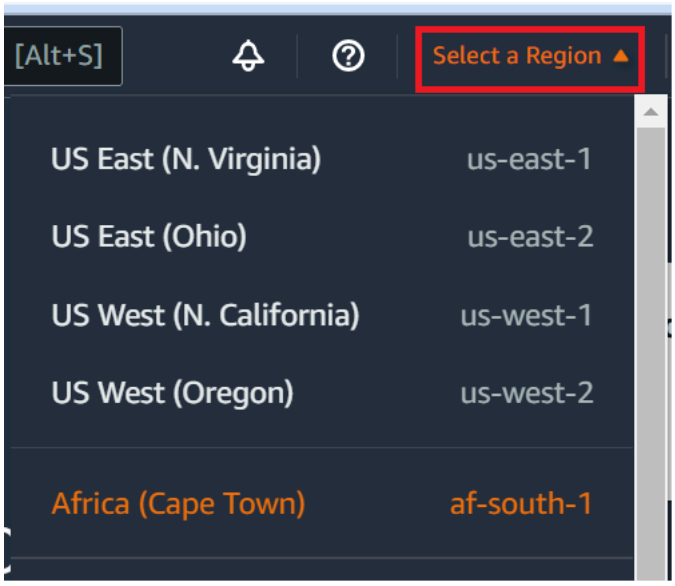
Untuk mencegah tagihan yang sedang berlangsung ke lingkungan AWS Cloud9 pengembangan yang Akun AWS terkait dengan lingkungan pengembangan yang tidak lagi Anda gunakan, hapus lingkungan.

- Menghapus Lingkungan dengan Konsol
- Menghapus Lingkungan dengan Kode

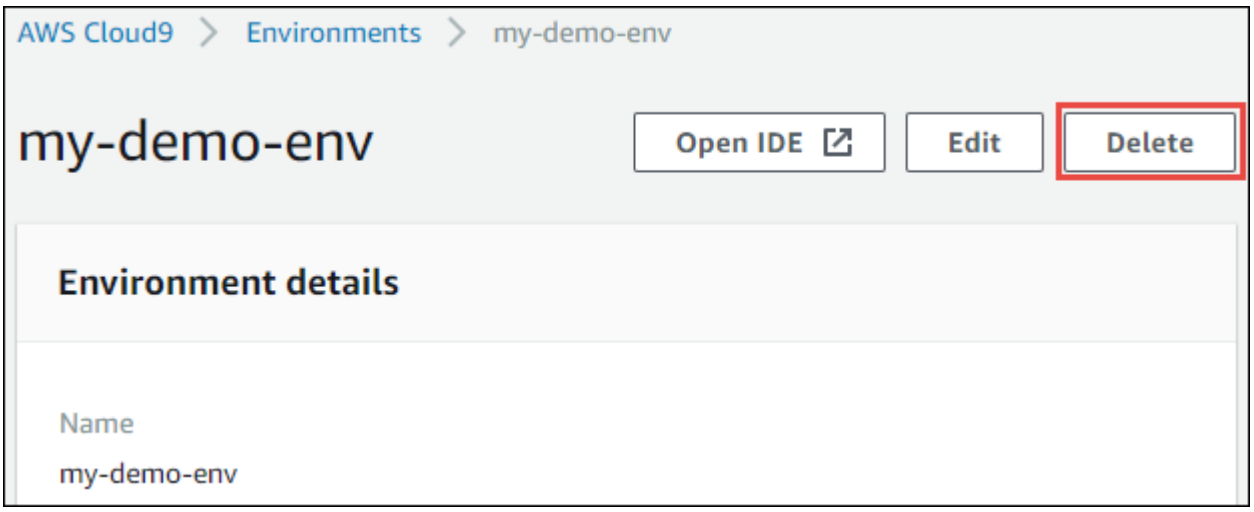

## Menghapus Lingkungan dengan konsol

> **Warning**
>
> Saat Anda menghapus lingkungan, AWS Cloud9 menghapus lingkungan secara permanen. Ini termasuk menghapus secara permanen semua pengaturan terkait, data pengguna, dan kode yang tidak terikat. Lingkungan yang dihapus tidak dapat dipulihkan.

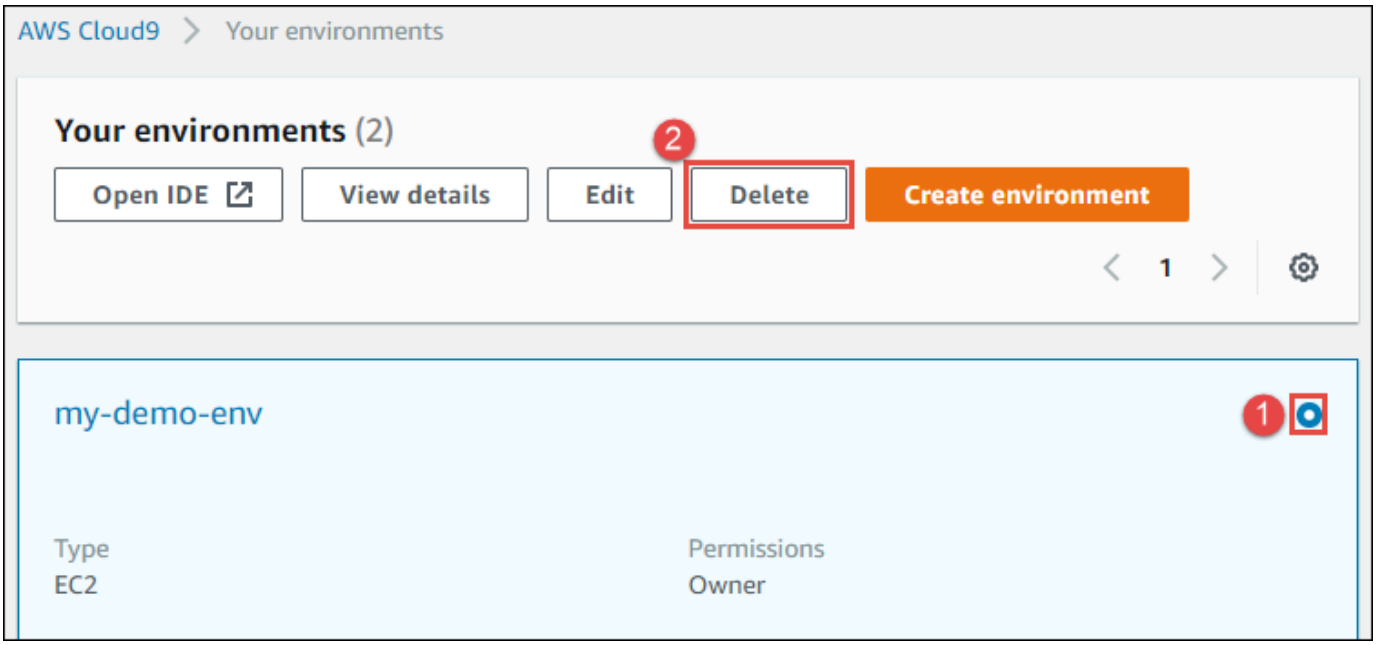
1. Masuk ke AWS Cloud9 konsol:
   - Jika Anda satu-satunya yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, tanyakan Akun AWS administrator Anda untuk instruksi masuk.
2. Di bilah navigasi atas, pilih Wilayah AWS tempat lingkungan berada.




3. Dalam daftar lingkungan, untuk lingkungan yang ingin Anda hapus, lakukan salah satu tindakan berikut.
   - Pilih judul kartu untuk lingkungan. Kemudian, pilih Hapus di halaman berikutnya.




- Pilih kartu untuk lingkungan, dan kemudian pilih tombol Hapus.




4. Dalam kotak dialog Hapus, ketikkan `Delete`, lalu pilih Hapus.

- EC2 lingkungan

  AWS Cloud9 juga menghentikan EC2 instance Amazon yang terhubung ke lingkungan itu.

> **Note**
>
> Jika penghapusan akun gagal, sebuah banner ditampilkan di bagian atas halaman web konsol. Selain itu, kartu untuk lingkungan, jika ada, menunjukkan bahwa penghapusan lingkungan gagal.

- Lingkungan SSH

  Jika lingkungan terhubung ke EC2 instans Amazon, AWS Cloud9 tidak menghentikan instance itu. Jika Anda tidak menghentikan instans itu nanti, Anda Akun AWS mungkin terus memiliki biaya berkelanjutan untuk Amazon EC2 terkait dengan instance itu.

5. Jika lingkungan adalah lingkungan SSH, AWS Cloud9 tinggalkan subdirektori tersembunyi di instance komputasi awan atau server Anda sendiri yang terhubung ke lingkungan itu. Jika Anda ingin menghapusnya, Anda sekarang dapat menghapus subdirektori itu dengan aman. Subdirektori tersebut bernama `.c9`. Subdirektori terletak di direktori jalur Lingkungan yang Anda tentukan saat Anda membuat lingkungan.

   Jika lingkungan Anda tidak ditampilkan di konsol, coba lakukan satu atau beberapa tindakan berikut agar lingkungan ditampilkan.

   - Di bilah menu tarik-turun di halaman Lingkungan, pilih satu atau lebih dari yang berikut ini.
     - Pilih lingkungan Saya untuk menampilkan semua lingkungan yang dimiliki AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.
     - Pilih Dibagikan dengan saya untuk menampilkan semua lingkungan yang diundang oleh AWS entitas Anda dalam lingkungan yang dipilih Wilayah AWS dan Akun AWS.
     - Pilih Semua lingkungan akun untuk menampilkan semua lingkungan dalam lingkungan Akun AWS yang dipilih Wilayah AWS dan AWS entitas Anda memiliki izin untuk ditampilkan.
   - Jika Anda merasa Anda adalah anggota lingkungan, tetapi lingkungan tidak ditampilkan dalam daftar Dibagikan dengan Anda, periksa dengan pemilik lingkungan.
   - Di bilah navigasi atas, pilih yang berbeda Wilayah AWS.

## Menghapus Lingkungan dengan kode

> **Warning**
>
> Saat Anda menghapus lingkungan, AWS Cloud9 menghapus lingkungan secara permanen. Ini termasuk menghapus secara permanen semua pengaturan terkait, data pengguna, dan kode yang tidak terikat. Lingkungan yang dihapus tidak dapat dipulihkan.

Untuk menggunakan kode untuk menghapus lingkungan AWS Cloud9, panggil operasi lingkungan AWS Cloud9 hapus, sebagai berikut.

| | |
|---|---|
| AWS CLI | hapus-lingkungan |
| AWS SDK untuk C++ | DeleteEnvironmentRequest, DeleteEnvironmentResult |
| AWS SDK untuk Go | DeleteEnvironment, DeleteEnvironmentRequest, DeleteEnvironmentWithContext |
| AWS SDK untuk Java | DeleteEnvironmentRequest, DeleteEnvironmentResult |
| AWS SDK untuk JavaScript | DeleteEnvironment |
| AWS SDK untuk .NET | DeleteEnvironmentRequest, DeleteEnvironmentResponse |
| AWS SDK untuk PHP | DeleteEnvironment |
| AWS SDK for Python (Boto) | delete_environment |
| AWS SDK untuk Ruby | delete_environment |
| AWS Tools for Windows PowerShell | Remove-C9Environment |
| AWS Cloud9 API | DeleteEnvironment |

# Bekerja dengan AWS Cloud9 IDE

Sebuah lingkungan pengembangan terpadu (IDE) menyediakan seperangkat alat produktivitas coding seperti editor kode sumber, debugger, dan alat-alat membangun.

**Important**

Kami merekomendasikan praktik terbaik berikut untuk menggunakan AWS Cloud9:

- Gunakan kontrol sumber dan buat cadangan lingkungan Anda sesering mungkin. AWS Cloud9 tidak melakukan backup otomatis.
- Lakukan pembaruan rutin perangkat lunak di lingkungan Anda. AWS Cloud9 tidak melakukan pembaruan perangkat lunak otomatis.
- AWS CloudTrailAktifkan AWS akun Anda untuk melacak aktivitas di lingkungan Anda. Untuk informasi selengkapnya, silakan lihat Pencatatan panggilan AWS Cloud9 API dengan AWS CloudTrail
- Hanya berbagi lingkungan Anda dengan pengguna tepercaya. Berbagi lingkungan Anda dapat membahayakan kredensi AWS akses Anda. Untuk informasi selengkapnya, silakan lihat Bekerja dengan lingkungan bersama di AWS Cloud9

Pelajari cara bekerja dengan AWS Cloud9 IDE dengan membaca satu atau lebih topik ini.

Topik


- Tur AWS Cloud9 IDE
- Dukungan bahasa di AWS Cloud9 IDE
- Dukungan bahasa yang ditingkatkan di AWS Cloud9 IDE
- Referensi perintah bilah menu untuk AWS Cloud9 IDE
- Menemukan dan Mengganti Teks di AWS Cloud9 IDE
- Mempratinjau file di IDE AWS Cloud9
- Mempratinjau aplikasi yang sedang berjalan di IDE AWS Cloud9
- Bekerja dengan Revisi File di IDE AWS Cloud9
- Bekerja dengan File Gambar di AWS Cloud9 IDE
- Bekerja dengan Pembangun, Pelari, dan Debugger di IDE AWS Cloud9

- Bekerja dengan Variabel Lingkungan Kustom di AWS Cloud9 IDE
- Bekerja dengan pengaturan proyek di AWS Cloud9 IDE
- Bekerja dengan pengaturan pengguna di AWS Cloud9 IDE
- Bekerja dengan pengaturan AWS proyek dan pengguna di AWS Cloud9 IDE
- Bekerja dengan Keybindings di IDE AWS Cloud9
- Bekerja dengan tema di AWS Cloud9 IDE
- Mengelola skrip inisialisasi di IDE AWS Cloud9
- Referensi Keybindings Default macOS untuk IDE AWS Cloud9
- Referensi Keybindings macOS Vim untuk IDE AWS Cloud9
- Referensi Keybindings macOS Emacs untuk IDE AWS Cloud9
- Referensi Keybindings Sublime macOS untuk IDE AWS Cloud9
- Referensi Keybindings Default Windows/Linux untuk IDE AWS Cloud9
- Referensi Keybindings Vim Windows/Linux untuk IDE AWS Cloud9
- Referensi Keybindings Windows/Linux Emacs untuk IDE AWS Cloud9
- Referensi Keybindings Sublime Windows/Linux untuk IDE AWS Cloud9
- Referensi perintah untuk AWS Cloud9 IDE

# Tur AWS Cloud9 IDE

Topik ini memberikan tur dasar tentang lingkungan pengembangan AWS Cloud9 terintegrasi (IDE). Untuk memanfaatkan sepenuhnya tur ini, ikuti langkah-langkah yang ditunjukkan di bawah ini secara berurutan.

Topik

- Prasyarat
- Langkah 1: Bilah menu
- Langkah 2: Dasbor
- Langkah 3: Jendela lingkungan
- Langkah 4: Editor, tab, dan panel
- Langkah 5: Konsol

- Langkah 6: Buka bagian file
- Langkah 7: Selokan
- Langkah 8: Bilah status
- Langkah 9: jendela Garis Besar
- Langkah 10: jendela Pergi
- Langkah 11: tab Segera
- Langkah 12: Daftar proses
- Langkah 13: Preferensi
- Langkah 14: Terminal
- Langkah 15: Jendela Debugger
- Pikiran akhir

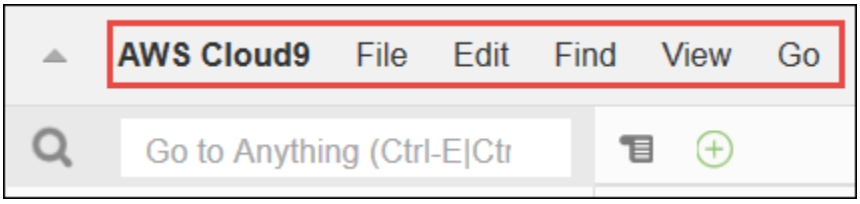

## Prasyarat

Untuk mengikuti tur ini, Anda harus memiliki AWS akun dan lingkungan AWS Cloud9 pengembangan terbuka. Untuk mempelajari cara melakukan hal-hal ini, Anda dapat mengikuti langkah-langkah di Mulai menggunakan AWS Cloud9. Atau, Anda dapat menjelajahi topik-topik terkait yang terpisah seperti Menyiapkan AWS Cloud9 dan Bekerja dengan lingkungan di AWS Cloud9.

> **Warning**
>
> Memiliki lingkungan AWS Cloud9 pengembangan dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk Amazon EC2 jika Anda menggunakan EC2 lingkungan. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

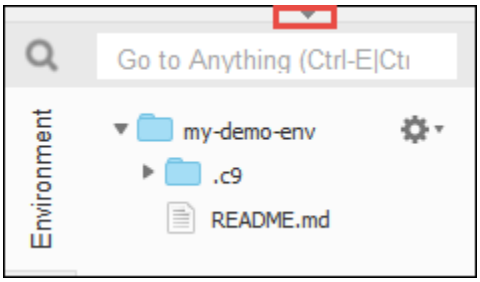
## Langkah 1: Bilah menu

bilah menu, di tepi atas IDE, berisi perintah umum untuk bekerja dengan file dan kode dan mengubah pengaturan IDE. Anda juga dapat melakukan pratinjau dan menjalankan kode dari bilah menu.

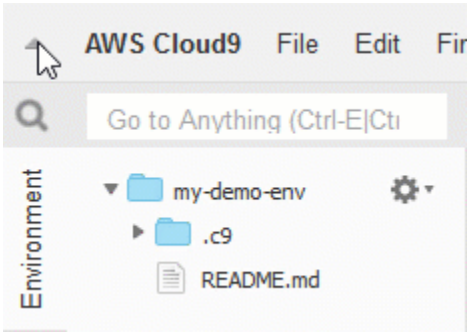


Anda dapat menyembunyikan bilah menu dengan memilih panah di tepinya, sebagai berikut.

Anda dapat menampilkan bilah menu lagi dengan memilih panah di tengah tempat bar menu sebelumnya, sebagai berikut.




Bandingkan hasil Anda dengan berikut.




Anda dapat menggunakan IDE untuk bekerja dengan satu set file dalam beberapa bagian berikutnya dalam tutorial ini. Untuk mengatur file-file ini, memilih File, File Baru.

Selanjutnya, salin teks berikut ke tab penyunting `Untitled1`.

```
fish.txt
--------
A fish is any member of a group of organisms that consist of
all gill-bearing aquatic craniate animals that lack limbs with
digits. They form a sister group to the tunicates, together
forming the olfactores. Included in this definition are
lampreys and cartilaginous and bony fish as well as various
extinct related groups.
```

Untuk menyimpan file, memilih File, Simpan. Beri nama file `fish.txt`, lalu memilih Simpan.

Ulangi petunjuk ini, menyimpan file kedua sebagai `cat.txt`, dengan konten berikut.

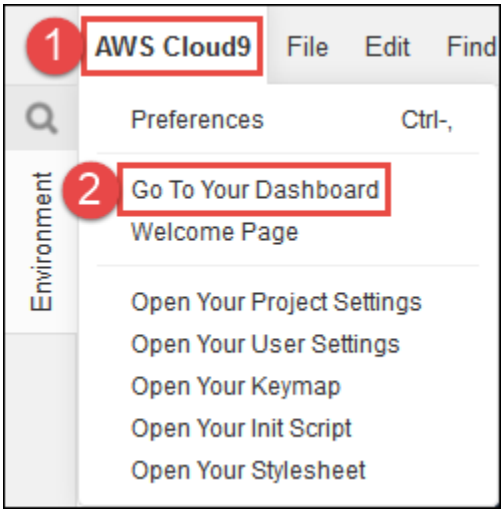
```
cat.txt
-------
The domestic cat is a small, typically furry, carnivorous mammal.
They are often called house cats when kept as indoor pets or
simply cats when there is no need to distinguish them from
other felids and felines. Cats are often valued by humans for
companionship and for their ability to hunt.
```

Sering ada beberapa cara untuk melakukan hal-hal di IDE. Misalnya, untuk menyembunyikan bilah menu, alih-alih memilih panah di tepinya, Anda bisa memilih Melihat, Bilah Menu. Untuk membuat file baru, alih-alih memilih File, File Baru Anda dapat menekan `Alt-N` (untuk Windows/Linux) atau `Control-N` (untuk macOS). Untuk mengurangi panjang tutorial ini, kami hanya menjelaskan satu cara untuk melakukan sesuatu. Ketika Anda merasa lebih nyaman dengan IDE, jangan ragu untuk bereksperimen dan mencari tahu cara yang terbaik untuk Anda.

## Langkah 2: Dasbor

Dasbor memberi Anda akses cepat ke setiap lingkungan Anda. Dari dasbor, Anda dapat membuat, membuka, dan mengubah pengaturan untuk lingkungan.

Untuk membuka dasbor, pada bilah menu, memilih AWS Cloud9, Pergi ke Dasbor Anda.




Untuk melihat pengaturan lingkungan Anda, pilih judul di dalam my-demo-environmentkartu. Untuk kembali ke dasbor, gunakan tombol kembali peramban web Anda atau breadcrumb navigasi yang disebut Lingkungan.

Untuk membuka IDE untuk lingkungan Anda, pilih Buka IDE di dalam my-demo-environmentkartu.

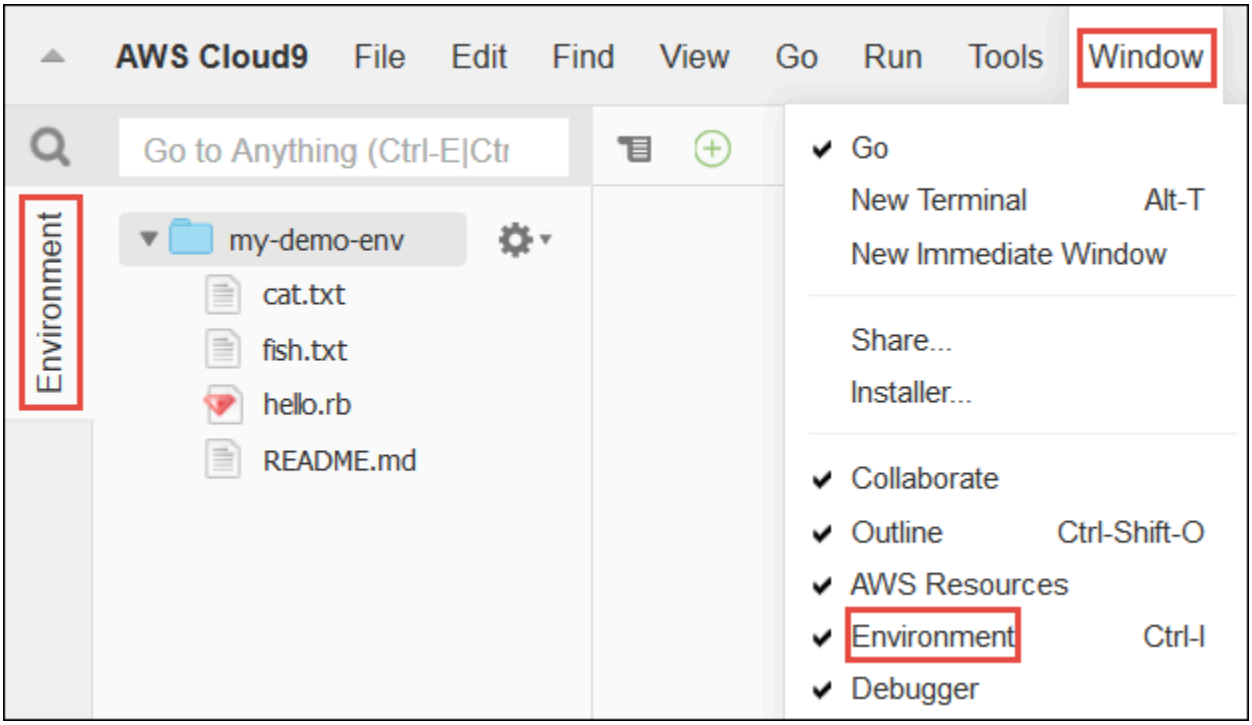
> **Note**
>
> Perlu waktu beberapa saat agar IDE ditampilkan lagi.

## Langkah 3: Jendela lingkungan

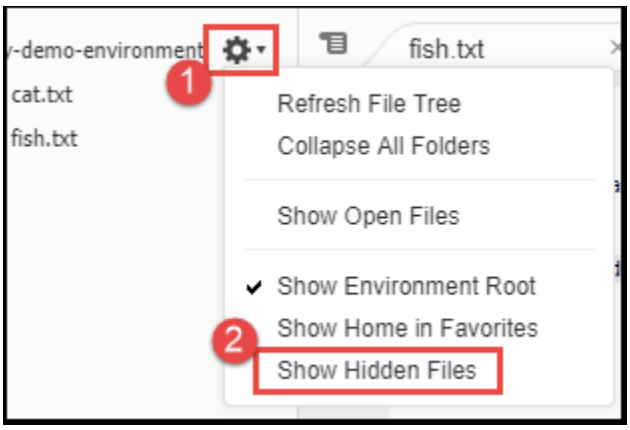
Jendela Lingkungan menampilkan daftar folder dan file di lingkungan. Anda juga dapat menampilkan berbagai jenis file, seperti file tersembunyi.

Untuk menampilkan atau menyembunyikan jendela Lingkungan, memilih tombol Lingkungan.

Untuk menampilkan atau menyembunyikan jendela Lingkungandan Lingkungan, memilih Jendela, Lingkungan pada bilah menu.




Untuk menampilkan atau menyembunyikan file tersembunyi, memilih ikon roda gigi di jendela Lingkungan, lalu memilih Tampilkan File Tersembunyi.

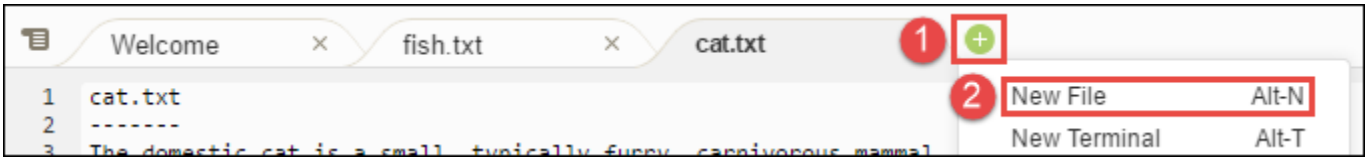



## Langkah 4: Editor, tab, dan panel

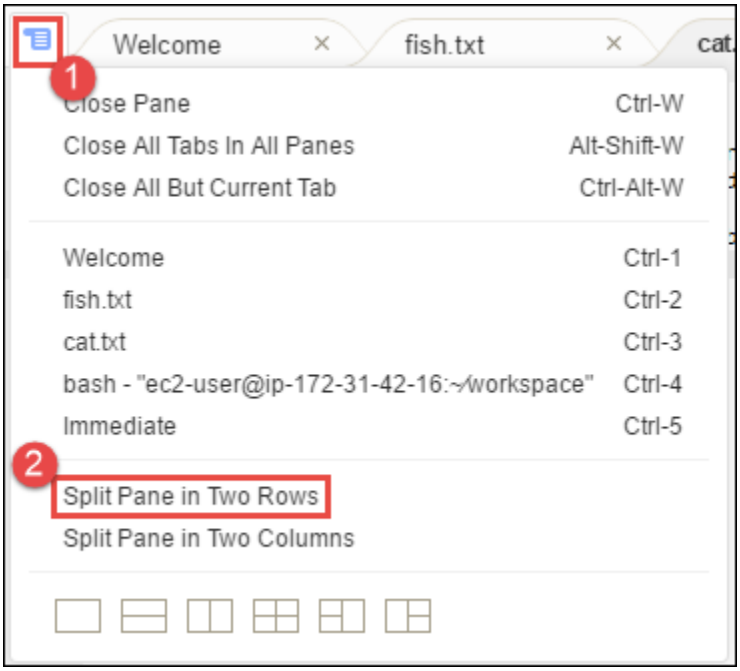
Editor adalah tempat Anda dapat melakukan hal-hal seperti tulis kode, menjalankan sesi terminal, dan mengubah pengaturan IDE. Setiap instans dari file terbuka, sesi terminal, dan sebagainya diwakili oleh tab. Tab dapat dikelompokkan ke panel. Tab ditampilkan di tepi panel mereka.

Untuk menampilkan atau menyembunyikan tab, memilih Melihat, Tombol Tab pada bilah menu.

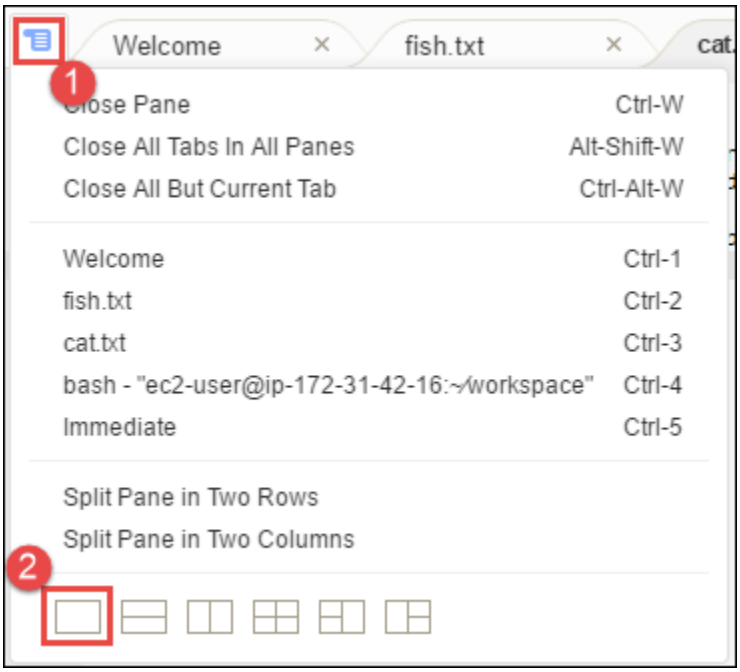
Untuk membuka tab baru, memilih ikon + di tepi baris tab. Kemudian memilih salah satu perintah yang tersedia, misalnya, File Baru, sebagai berikut.




Untuk menampilkan dua panel, memilih ikon yang terlihat seperti menu drop-down, yang berada di tepi deretan tab. Lalu memilih Panel Terpisah dalam Dua Baris sebagai berikut.

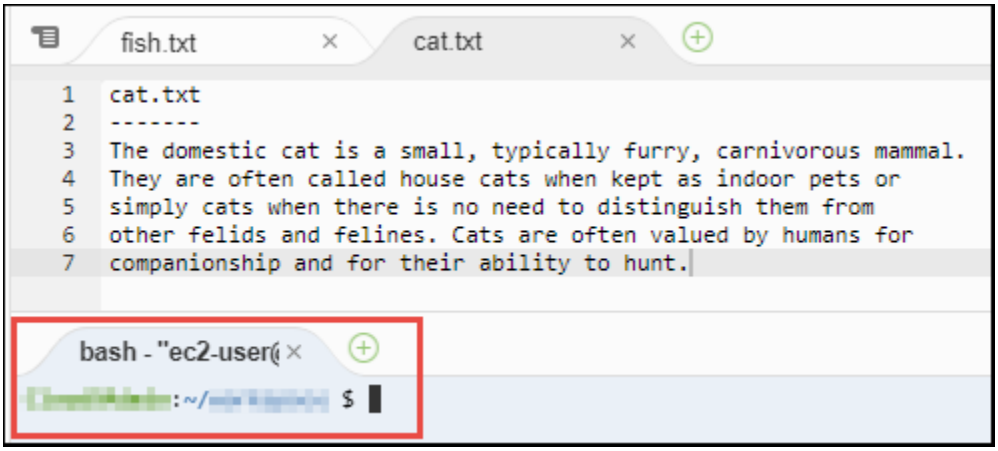



Untuk kembali ke satu panel, memilih ikon menu drop-down lagi, lalu memilih ikon persegi tunggal, sebagai berikut.



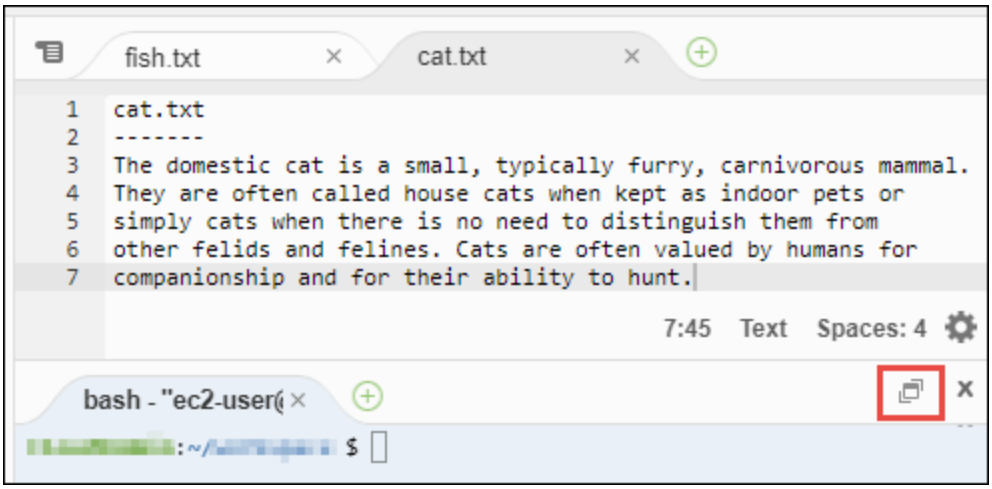

## Langkah 5: Konsol

konsol tersebut adalah tempat alternatif untuk membuat dan mengelola tab. Secara default, ini berisi tab Terminal, tetapi juga dapat berisi jenis tab lainnya.



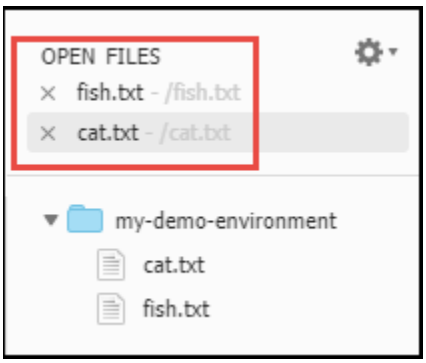

Untuk menampilkan atau menyembunyikan konsol tersebut, memilih Melihat, Konsol pada bilah menu.

Untuk memperluas atau mengecilkan konsol tersebut, memilih ikon resize, yang berada di tepi konsol, sebagai berikut.




## Langkah 6: Buka bagian file

Bagian Buka File menunjukkan daftar semua file yang saat ini terbuka di editor. Buka File adalah bagian dari jendela Lingkungan.

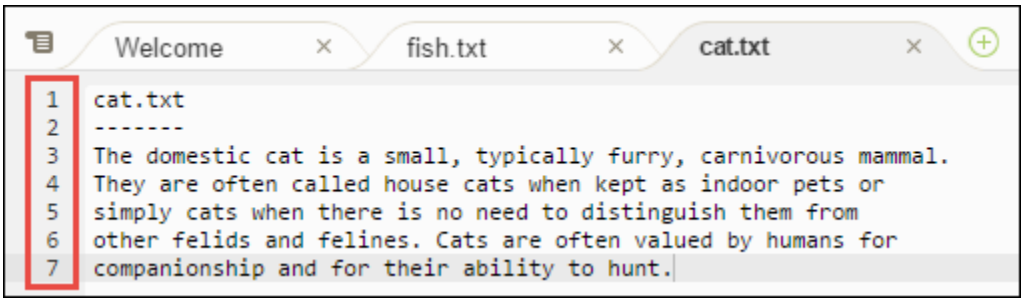



Untuk menampilkan atau menyembunyikan bagian Buka File, memilih Melihat, Buka File pada bilah menu.

Untuk beralih di antara file yang terbuka, memilih file yang menarik dari daftar.

## Langkah 7: Selokan

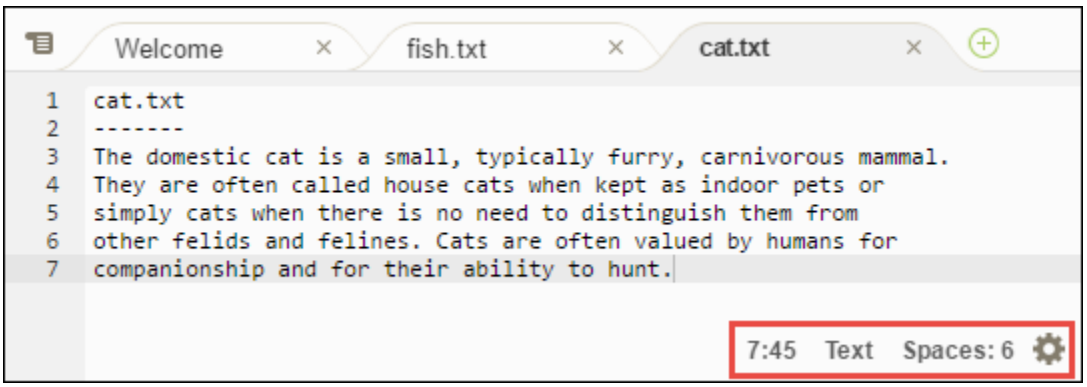
selokan, di tepi setiap file di editor, menunjukkan hal-hal seperti nomor baris dan simbol kontekstual saat Anda bekerja dengan file.




Untuk menampilkan atau menyembunyikan selokan, memilih Melihat, Selokan pada bilah menu.

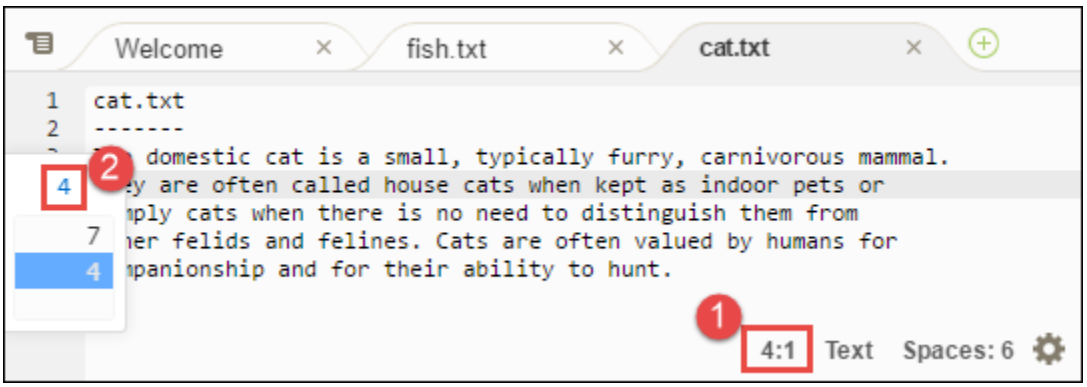
## Langkah 8: Bilah status

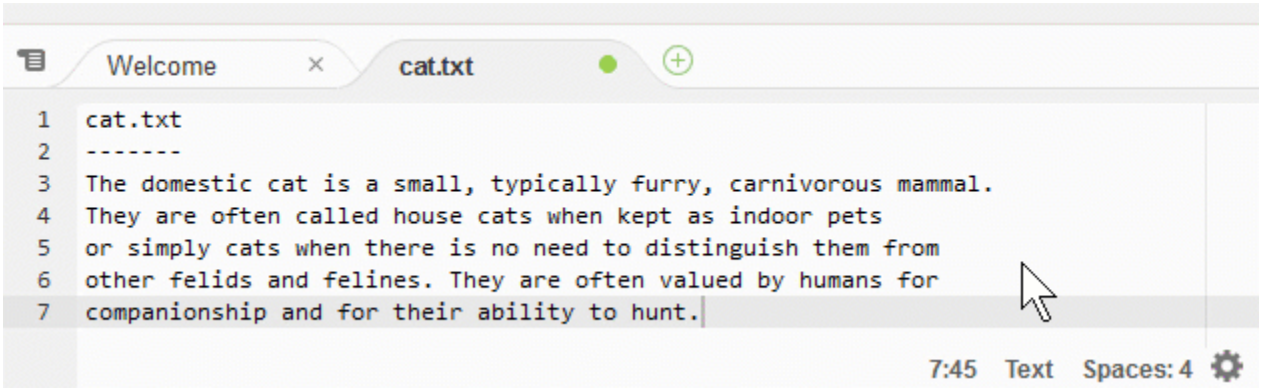
Parameter bilah status, di tepi setiap file di editor, menunjukkan hal-hal seperti nomor baris dan karakter, preferensi jenis file, pengaturan ruang dan tab, dan pengaturan editor terkait.



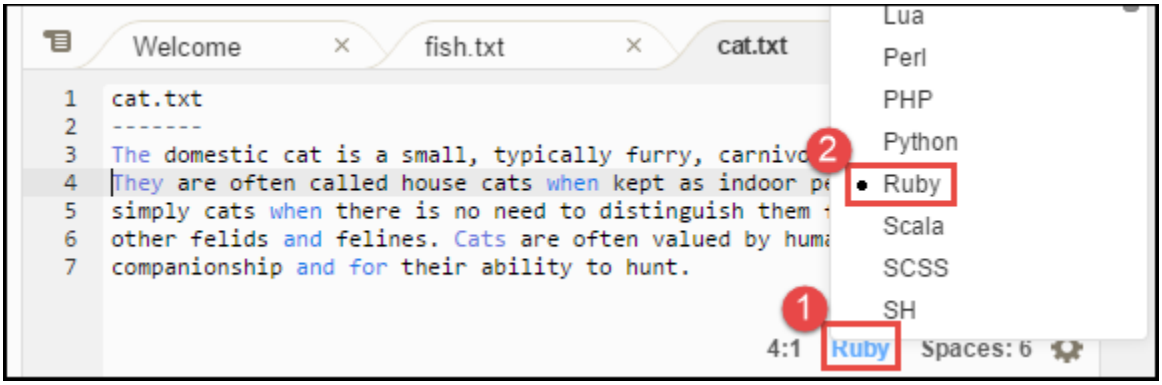

Untuk menampilkan atau menyembunyikan bilah status, memilih Melihat, Bilah Status pada bilah menu.

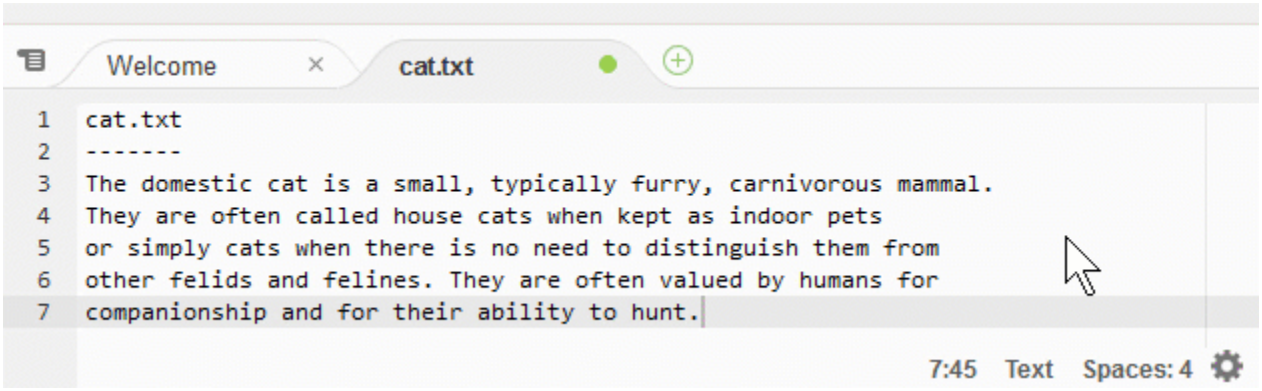
Untuk membuka nomor baris tertentu, memilih tab dengan file yang diminati. Kemudian di bilah status, memilih baris dan nomor karakter (itu harus sesuatu seperti 7:45). Ketik nomor baris (seperti `4`), lalu tekan `Enter`, sebagai berikut.




Untuk mengubah preferensi jenis file, di bilah status, memilih jenis file yang berbeda. Misalnya, untuk cat.txt, memilih Ruby untuk melihat perubahan warna sintaks. Untuk kembali ke warna teks biasa, memilih Teks Biasa, sebagai berikut.




## Langkah 9: jendela Garis Besar

Anda dapat menggunakan jendela Garis Besar untuk pergi ke lokasi file tertentu dengan cepat.

Untuk menampilkan atau menyembunyikan jendela Garis Besar dan tombol Garis Besar, memilih Jendela, Garis Besar pada bilah menu.

Untuk melihat bagaimana jendela Garis Besar bekerja, membuat file bernama `hello.rb`. Salin kode berikut ke dalam file dan simpan.

```
def say_hello(i)
  puts "Hello!"
  puts "i is #{i}"
end

def say_goodbye(i)
  puts "i is now #{i}"
  puts "Goodbye!"
end

i = 1
say_hello(i)
```

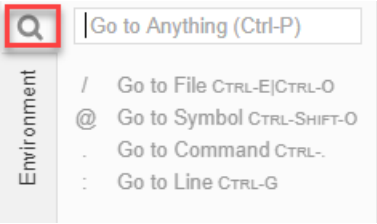
```
i += 1
say_goodbye(i)
```

Untuk menampilkan atau menyembunyikan konten jendela Garis Besar, memilih tombol Garis Besar.

Di jendela Garis Besar, memilih say_hello(i), lalu pilih say_goodbye(i), sebagai berikut.

## Langkah 10: jendela Pergi

Anda dapat menggunakan jendela Pergi untuk membuka file dalam editor, pergi ke definisi simbol, jalankan perintah, atau pergi ke baris dalam file aktif dalam editor.




Untuk menampilkan konten jendela Pergi, memilih Pergi (ikon kaca pembesar).

Untuk menampilkan atau menyembunyikan jendela Pergi dan tombol Pergi, memilih Jendela, Pergi pada bilah menu.

Dengan jendela Pergi terbuka, Anda dapat:

- Ketik garis miring ke depan (`/`) diikuti oleh sebagian atau seluruh nama file. Dalam daftar file yang cocok yang ditampilkan, memilih file untuk membukanya di editor. Misalnya, mengetik `/fish` daftar `fish.txt`, saat mengetik `/.txt` tandai `fish.txt` dan `cat.txt`.

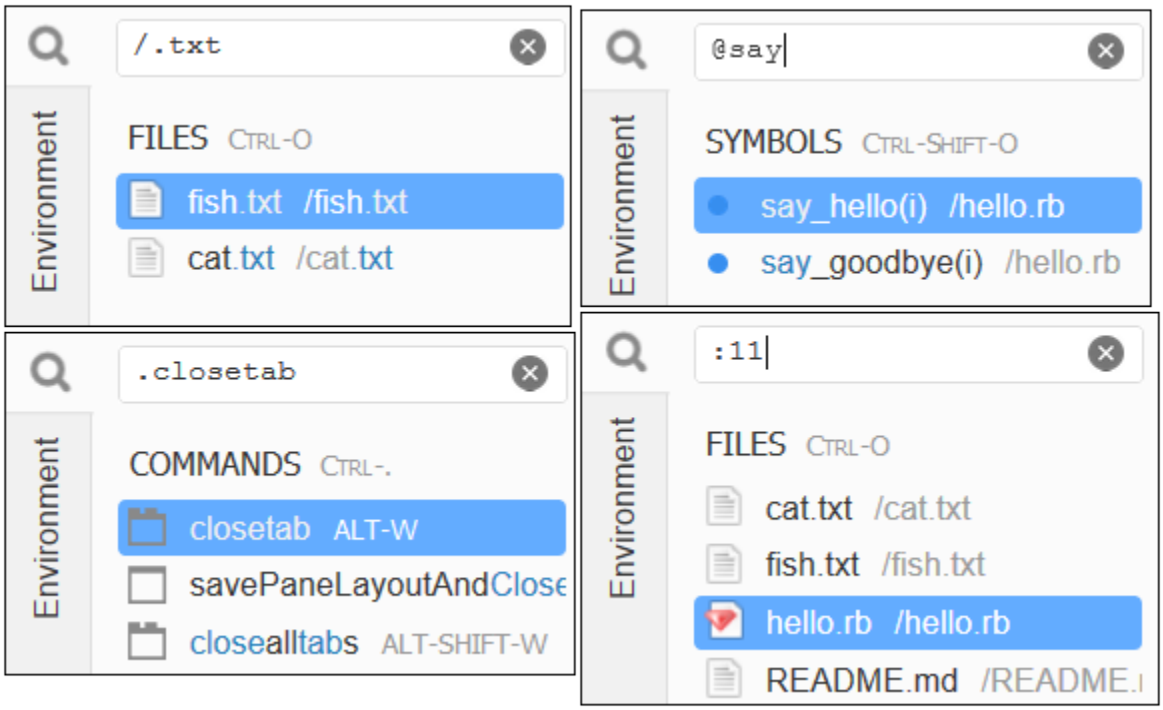
  > **Note**
  >
  > Pencarian file hanya dicakup untuk file tidak tersembunyi dan folder tidak tersembunyi di jendela Lingkungan.

- Ketik simbol at (`@`) diikuti dengan nama simbol. Dalam daftar file yang cocok yang ditampilkan, memilih simbol untuk membukanya di editor. Misalnya, dengan file `hello.rb` terbuka dan aktif di editor, ketik `@hello` daftar `say_hello(i)`, atau ketik `@say` untuk mencantumkan `say_hello(i)` dan `say_goodbye(i)`.

  > **Note**
  >
  > Jika file aktif dalam editor adalah bagian dari proyek bahasa yang di support, pencarian simbol dicakupkan ke proyek saat ini. Jika tidak, pencarian simbol hanya dicakupkan ke file aktif di editor. Untuk informasi selengkapnya, lihat TypeScript Dukungan dan fitur yang ditingkatkan.

- Ketik titik (`.`) diikuti dengan nama perintah. Dalam daftar perintah yang ditampilkan, memilih perintah untuk menjalankannya. Misalnya, mengetik `.closetab` dan kemudian menekan `Enter` menutup tab saat ini di editor. Untuk daftar lengkap metrik yang tersedia, lihat Referensi perintah untuk AWS Cloud9 IDE.

- Ketik titik dua (`:`) diikuti dengan nomor untuk pergi ke nomor baris dalam file aktif di editor. Misalnya, dengan file `hello.rb` terbuka dan aktif di editor, ketik `:11` untuk pergi ke baris 11 dalam file tersebut.




Untuk melihat pengikatan tombol untuk masing-masing tindakan ini berdasarkan mode keyboard dan sistem operasi saat ini, lihat masing-masing perintah Pergi Ke pada menu Pergi di bilah menu.

## Langkah 11: tab Segera

Tab Segera memungkinkan Anda untuk menguji potongan kecil kode. JavaScript Untuk melihat bagaimana tab Segera bekerja, lakukan hal berikut.

1. Buka tab Segera dengan memilih Jendela, Jendela Segera Baru pada bilah menu.
2. Menjalankan beberapa kode di tab Segera. Untuk mencoba ini, ketik kode berikut ke jendela, menekan `Shift-Enter` setelah mengetik baris 1 dan lagi setelah baris 2. Tekan `Enter` setelah baris 3. (Jika Anda menekan `Enter` alih-alih `Shift-Enter` setelah Anda mengetik baris 1 atau baris 2, kode akan berjalan lebih awal dari yang Anda inginkan.)

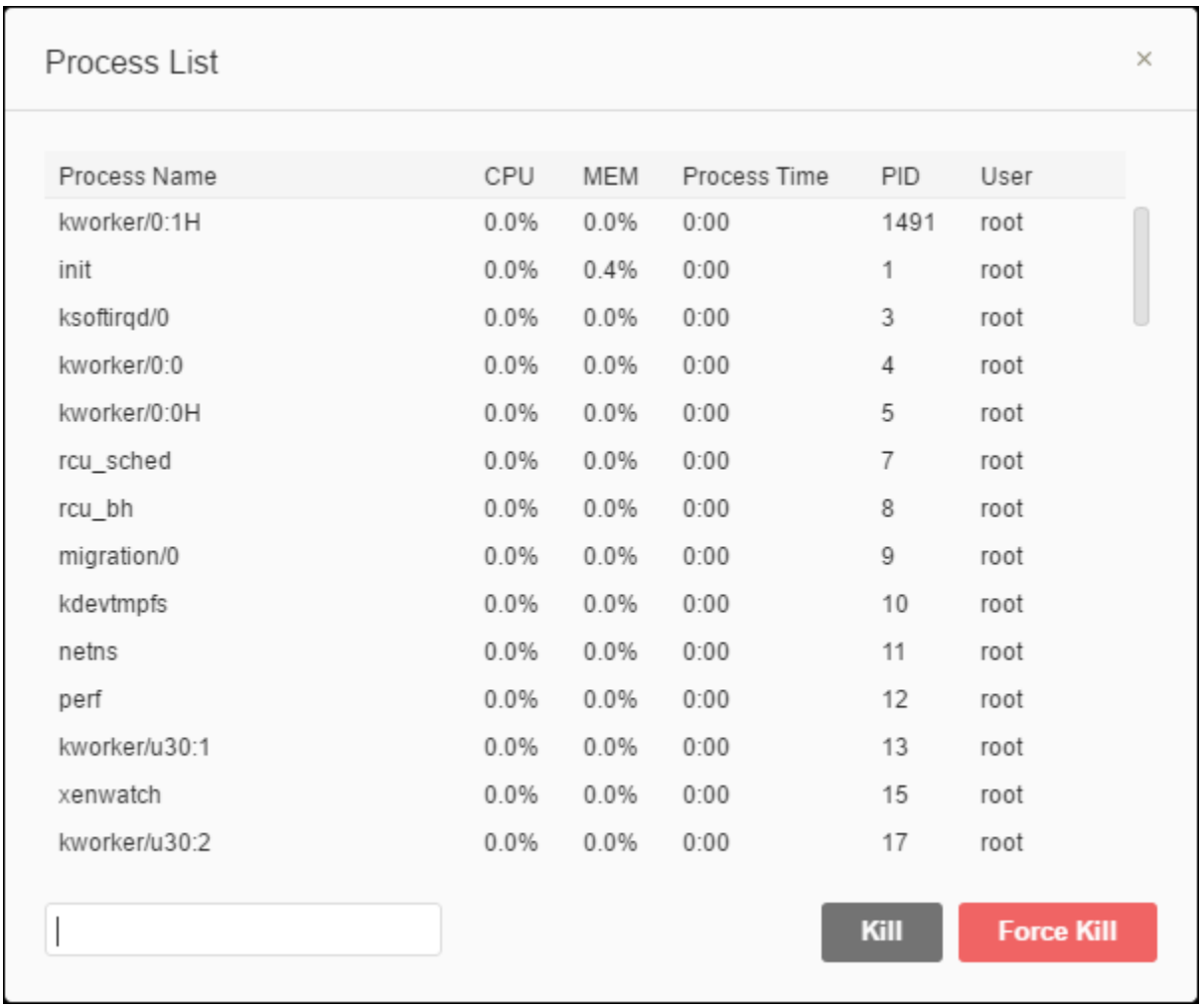
```
for (i = 0; i <= 10; i++) { // Press Shift-Enter after typing this line.
  console.log(i)            // Press Shift-Enter after typing this line.
```

```
}                          // Press Enter after typing this line. The numbers 0 to
 10 will be printed.
```

## Langkah 12: Daftar proses

Daftar Proses menunjukkan semua proses yang sedang berjalan. Anda dapat menghentikan atau bahkan secara paksa menghentikan proses yang tidak ingin Anda jalankan lagi. Untuk melihat bagaimana jendela Daftar Proses bekerja, lakukan hal berikut.

1. Tampilkan Daftar Proses dengan memilih Alat, Daftar Proses pada bilah menu.
2. Cari sebuah proses. Di Daftar Proses, ketik nama proses.
3. Hentikan atau hentikan paksa proses. Dalam daftar proses, memilih prosesnya, lalu memilih Bunuh atau Bunuh Paksa.

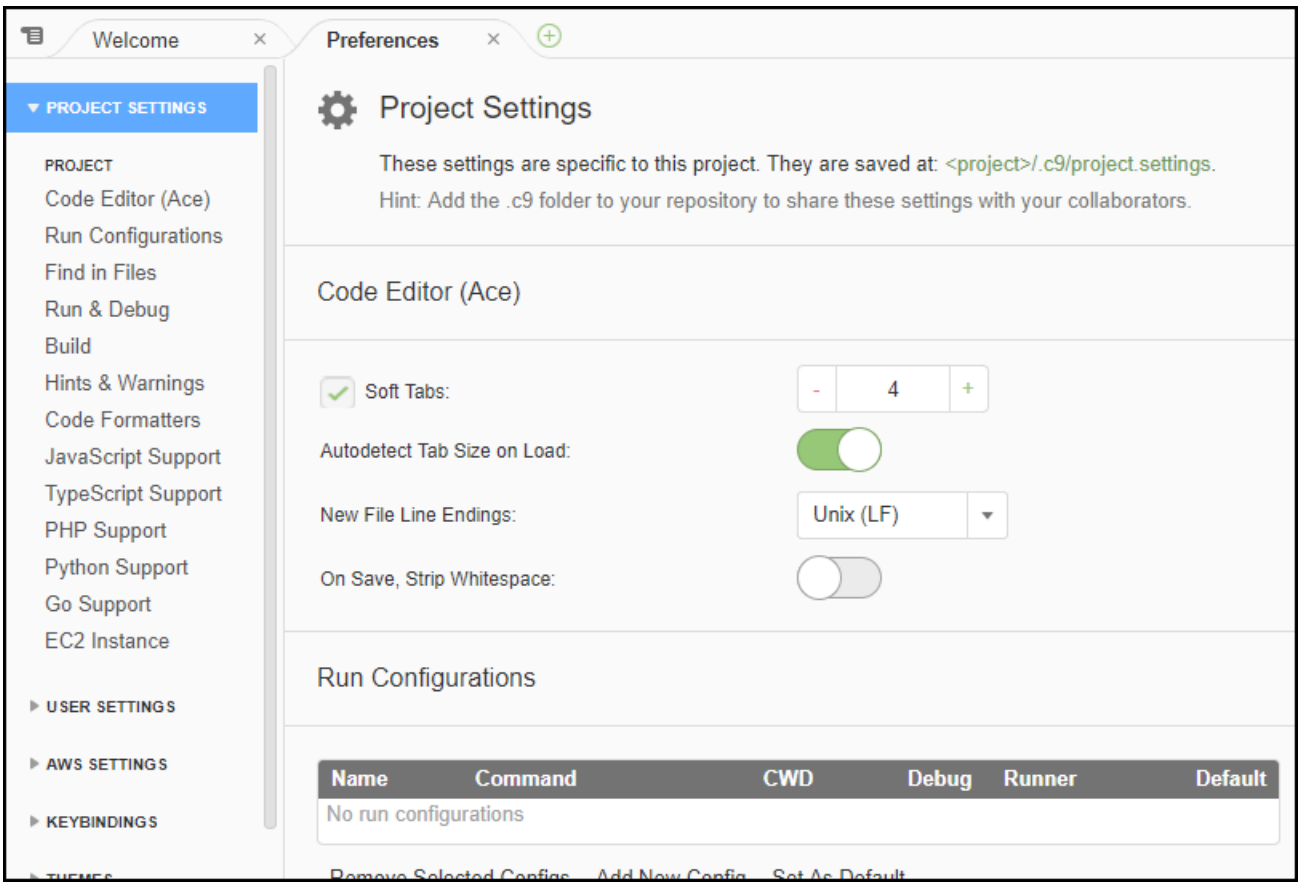



## Langkah 13: Preferensi

Preferensi mencakup pengaturan berikut.

- Pengaturan untuk lingkungan saat ini saja, seperti apakah akan menggunakan tab lunak di editor, jenis file yang akan diabaikan, dan perilaku penyelesaian kode untuk bahasa seperti PHP dan Python.
- Pengaturan pengguna Anda di setiap lingkungan Anda, seperti warna, font, dan perilaku editor.
- Pengikatan tombol Anda, seperti kombinasi kunci pintas yang ingin Anda gunakan untuk bekerja dengan file dan editor.
- Tema keseluruhan IDE.

Untuk menampilkan preferensi, memilih AWS Cloud9, Preferensi pada bilah menu. Sesuatu seperti berikut ditampilkan.




## Langkah 14: Terminal

Anda dapat menjalankan satu atau lebih sesi terminal dalam IDE. Untuk meluncurkan sesi terminal baru, memilih Jendela, Terminal Baru pada bilah menu. Atau, memilih ikon "plus" di samping tab konsol tersebut dan memilih Terminal Baru.

Anda dapat mencoba menjalankan perintah di terminal. Sebagai contoh, di terminal, ketik `echo $PATH` dan kemudian tekan `Enter` untuk mencetak nilai variabel lingkungan `PATH`.

Anda juga dapat mencoba menjalankan perintah tambahan. Misalnya, coba perintah seperti berikut ini.

- **`pwd`** untuk mencetak path ke direktori saat ini.
- **`aws --version`**untuk mencetak informasi versi tentang AWS CLI.
- **`ls -l`** untuk mencetak informasi tentang direktori saat ini.

## Langkah 15: Jendela Debugger

Anda dapat menggunakan jendela Debugger untuk men-debug kode Anda. Misalnya, Anda dapat melangkah melalui menjalankan kode porsi pada satu waktu, melihat nilai-nilai variabel dari waktu ke waktu, dan menjelajahi tumpukan panggilan.

> **Note**
>
> Prosedur ini mirip dengan Langkah 2: Tur Basic IDE dari salah satu Tutorial IDE basic.

Untuk menampilkan atau menyembunyikan jendela Debugger dan tombol Debugger, memilih Jendela, Debugger pada bilah menu.

Untuk tutorial ini, Anda dapat bereksperimen dengan jendela Debugger dan beberapa JavaScript kode dengan melakukan hal berikut.

1. Memeriksa instalasi Node.js di lingkungan Anda dengan menjalankan perintah berikut dalam sesi terminal: **`node --version`**. Jika Node.js diinstal, nomor versi Node.js ditampilkan dalam output, dan Anda dapat melompat ke langkah 3 dalam prosedur ini (“Tulis beberapa JavaScript kode...”).
2. Jika Anda perlu menginstal Node.js, lakukan hal berikut.

   a. Jalankan dua perintah berikut, satu per satu, untuk memastikan lingkungan Anda memiliki pembaruan terbaru dan kemudian unduh Node Version Manager (nvm). (nvm adalah skrip shell Bash sederhana yang berguna untuk menginstal dan mengelola versi Node.js. Untuk informasi selengkapnya, lihat Pengelola Versi Node di GitHub.)

   Untuk Amazon Linux:

   ```
   sudo yum -y update
   curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.0/install.sh |
    bash
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt update
   curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.0/install.sh |
    bash
   ```

   b. Gunakan editor teks untuk memperbarui file profil shell Anda (misalnya, `~/.bashrc`) untuk mengaktifkan nvm untuk memuat. Misalnya, dalam jendela Lingkungan IDE, memilih ikon gear, dan kemudian memilih Tampilkan Rumah di Favorit. Ulangi langkah ini dan memilih Tampilkan File Tersembunyi juga.

   c. Buka file `~/.bashrc`.

   d. Ketik atau tempel kode berikut di akhir file untuk mengaktifkan nvm untuk memuat.

   Untuk Amazon Linux:

   ```
   export NVM_DIR="/home/ec2-user/.nvm"
   [ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  # This loads nvm.
   ```

   Untuk Ubuntu Server:

   ```
   export NVM_DIR="/home/ubuntu/.nvm"
   ```

```
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  # This loads nvm.
```

e. Simpan file tersebut.

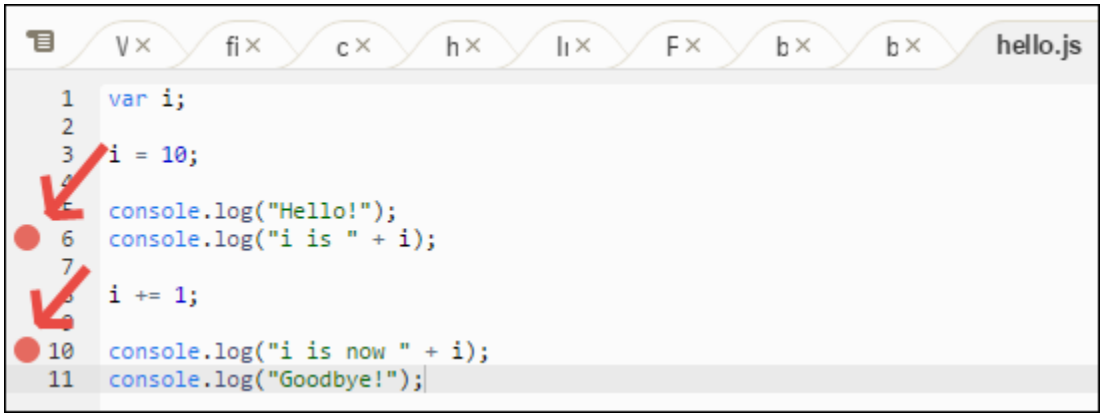
f. Tutup sesi terminal itu dan mulai yang baru. Kemudian jalankan perintah berikut untuk menginstal versi terbaru Node.js.

```
nvm install node
```

3. Tulis beberapa JavaScript kode untuk di-debug. Sebagai contoh, membuat file, menambahkan kode berikut ke file, dan menyimpannya sebagai `hello.js`.

```
var i;

i = 10;

console.log("Hello!");
console.log("i is " + i);

i += 1;

console.log("i is now " + i);
console.log("Goodbye!");
```

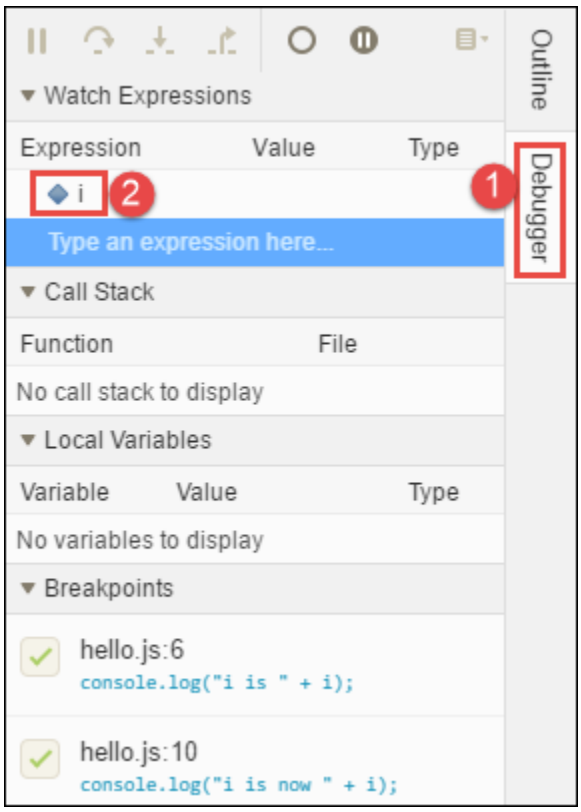
4. Menambahkan beberapa breakpoints ke kode. Misalnya, di selokan, memilih margin di sebelah garis 6 dan 10. Lingkaran merah ditampilkan di samping masing-masing nomor baris ini, sebagai berikut.



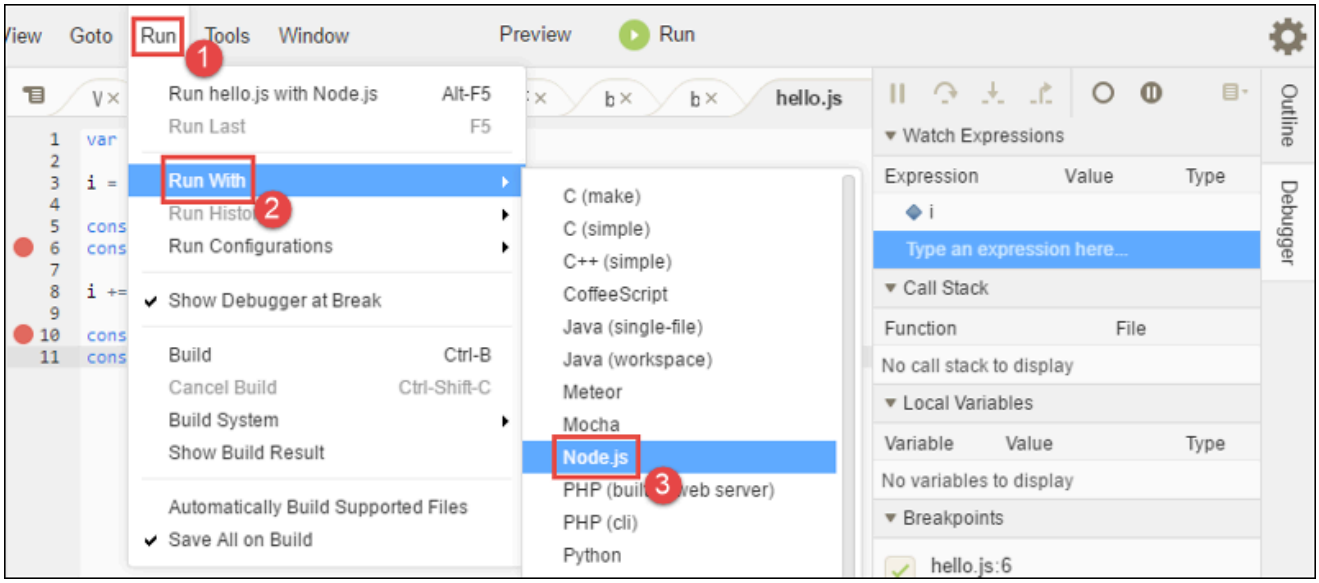

5. Sekarang Anda siap untuk men-debug JavaScript kode. Untuk melakukannya, jalankan yang berikut ini.

a. Untuk menampilkan atau menyembunyikan konten jendela Debugger, memilih tombol Debugger, seperti yang ditunjukkan pada langkah berikutnya.

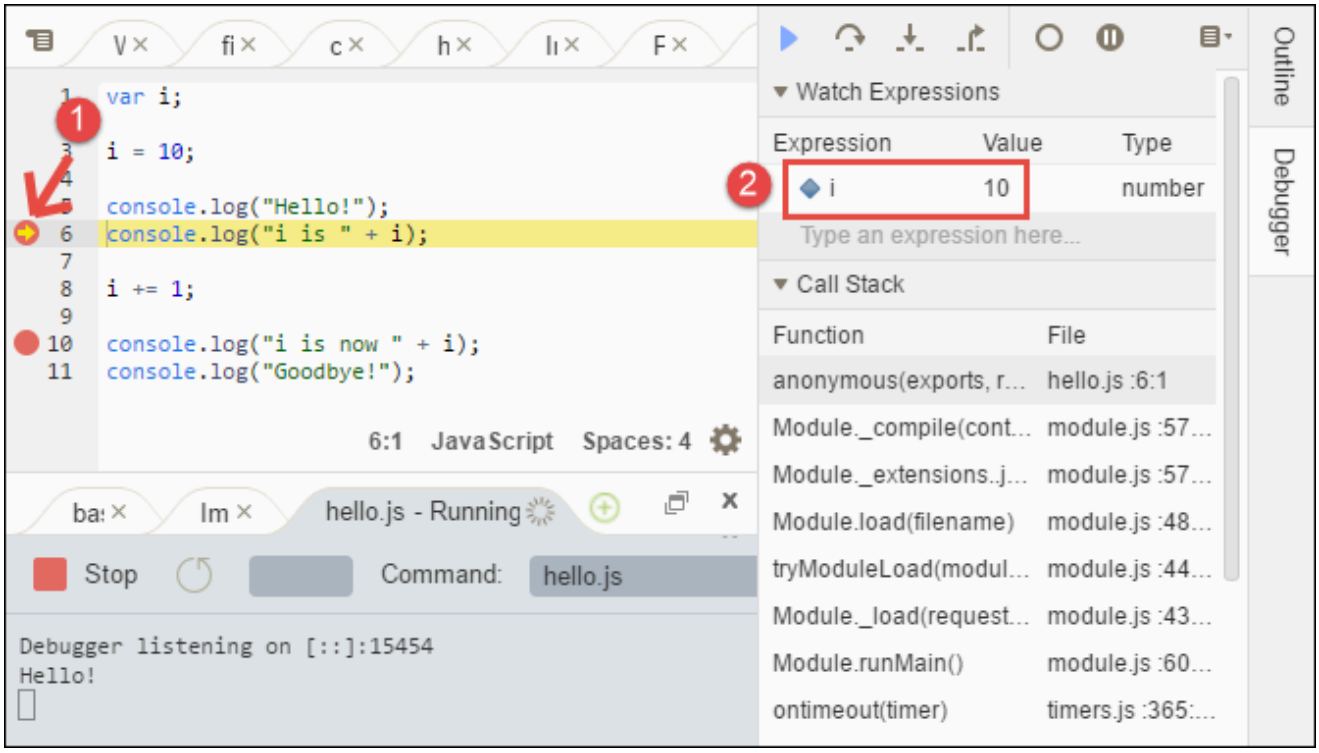
b. Melihat nilai variabel bernama `i` sementara kode berjalan. Di jendela Debugger, untuk Lihat Ekspresi, memilih Ketik ekspresi di sini. Ketik huruf `i`, lalu tekan `Enter`, sebagai berikut.



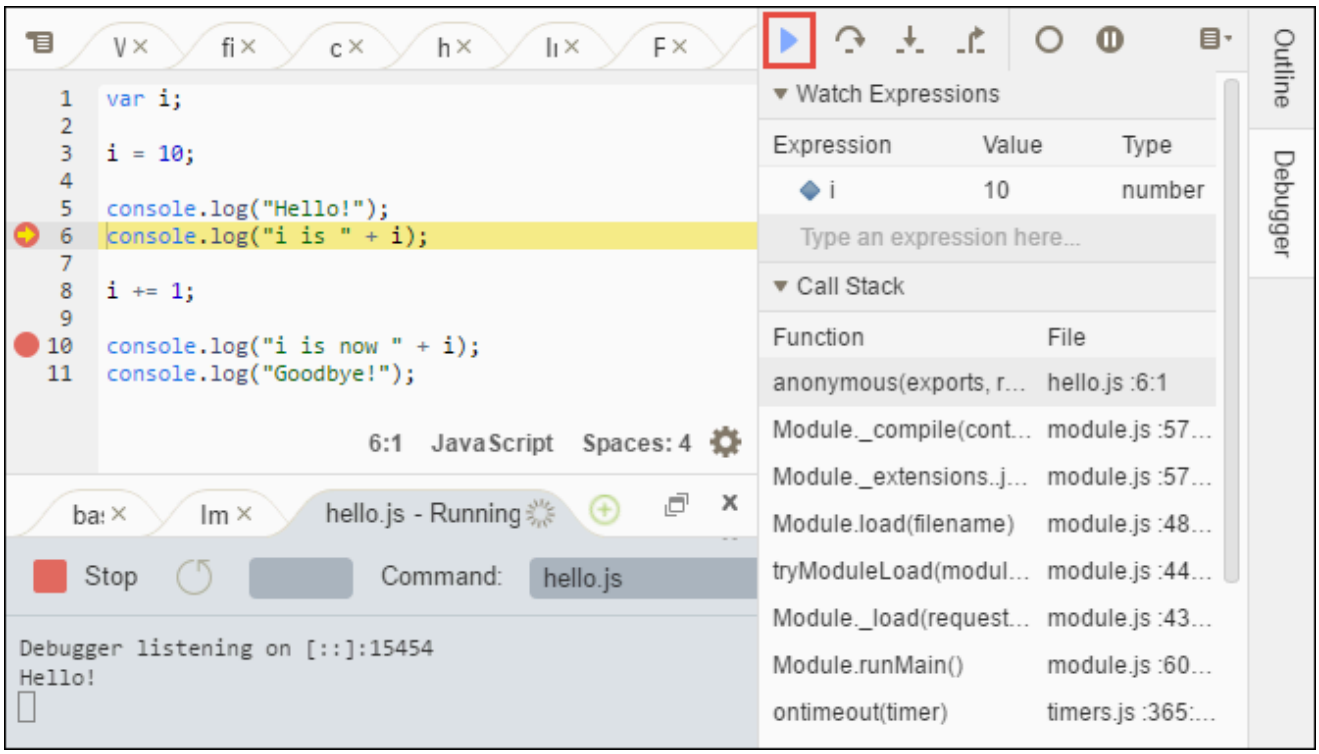

c. Mulai menjalankan kode. Memilih Jalankan, Jalankan Dengan, Node.js, sebagai berikut.




d. Kode berhenti berjalan pada baris 6. Jendela Debugger menunjukkan nilai `i` di Lihat Ekspresi, yang saat ini `10`.

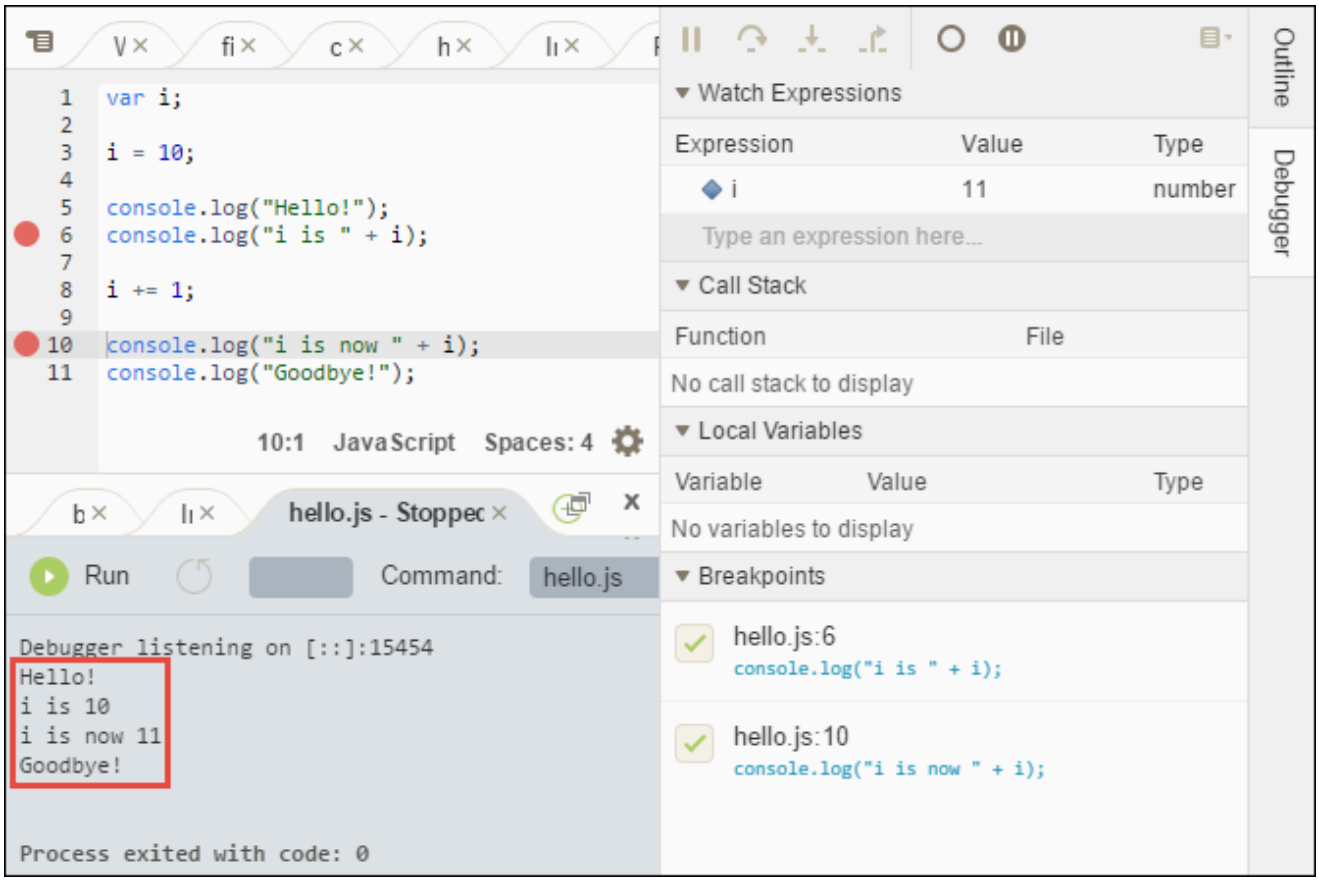



e. Di jendela Debugger, memilih Lanjutkan, yang merupakan ikon panah biru, seperti berikut.

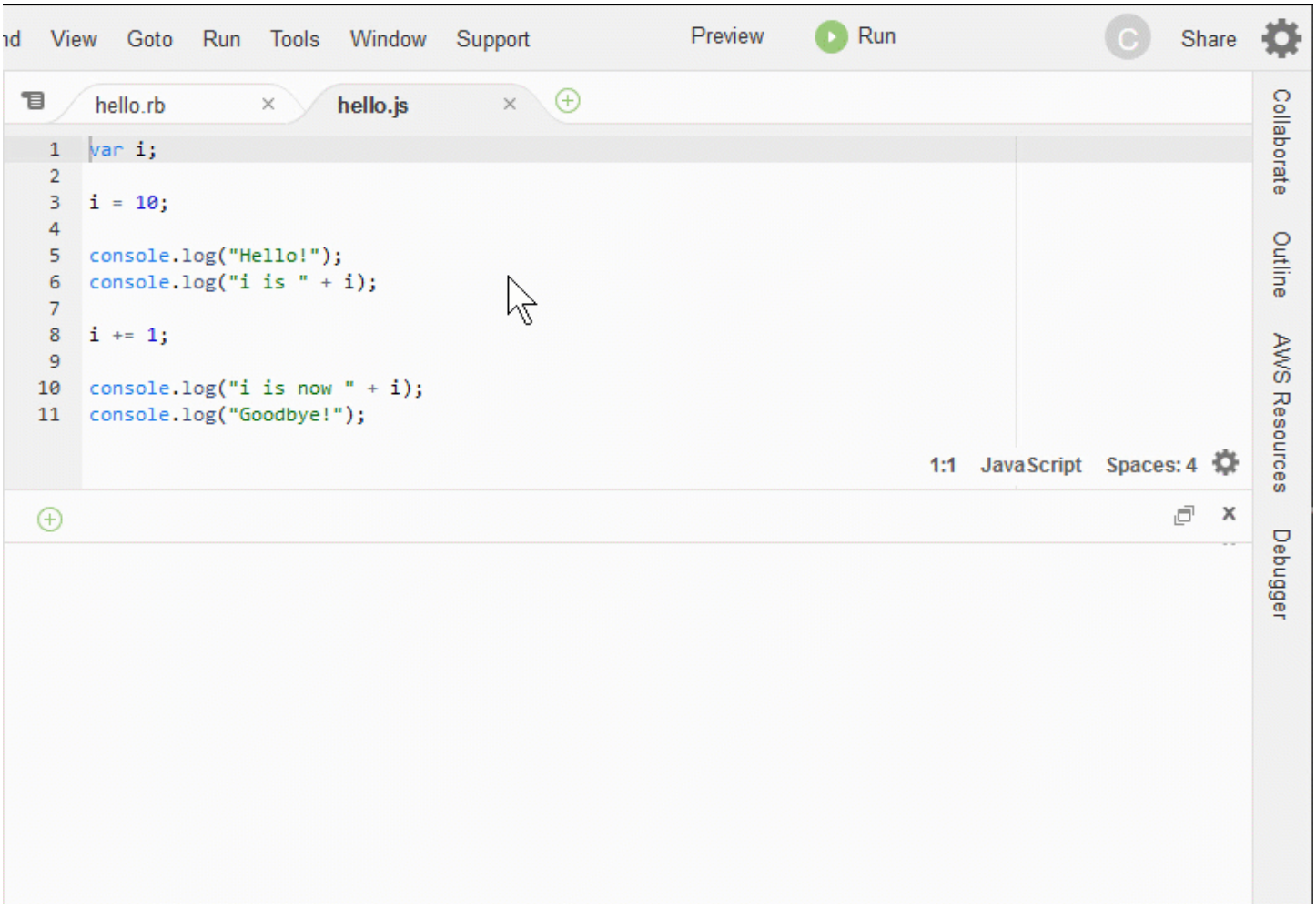



f. Kode berhenti berjalan pada baris 10. Jendela Debugger sekarang menunjukkan nilai `i`, yang saat ini 11.

g. Memilih Lanjutkan lagi. Kode berjalan sampai akhir. Output dicetak ke tab hello.js konsol tersebut, sebagai berikut.




Bandingkan hasil Anda dengan berikut.




## Pikiran akhir

**Warning**

Ingatlah bahwa memiliki lingkungan AWS Cloud9 pengembangan dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk Amazon EC2 jika Anda menggunakan EC2 lingkungan. Untuk informasi selengkapnya, lihat EC2 Harga Amazon. Ada topik tambahan di bagian induk (Bekerja dengan IDE) yang mungkin ingin Anda jelajahi. Namun, ketika Anda selesai tur AWS Cloud9 IDE dan tidak lagi membutuhkan lingkungan, pastikan untuk menghapusnya dan sumber daya terkait, seperti yang dijelaskan dalamMenghapus Lingkungan.

# Dukungan bahasa di AWS Cloud9 IDE

AWS Cloud9 IDE mendukung banyak bahasa pemrograman. Tabel berikut mencantumkan bahasa yang didukung dan tingkat apa.

| Bahasa | Penyorotan sintaks [1] | UI penjalanan [2] | Tampilan garis besar | Petunjuk kode dan linting | Penyelesaian kode | Debugging [3] |
|---|---|---|---|---|---|---|
| C++ | ✓ | ✓ | ✓ | | ✓ [5] | ✓ [4] |
| C# | ✓ | | ✓ | | ✓ [5] | |
| CoffeeScript | ✓ | ✓ | | | | |
| CSS | ✓ | | | | ✓ | |
| Dart | ✓ | | | | | |
| Go | ✓ | ✓ | ✓ | ✓ | ✓ [4] | ✓ [4] |
| Haskell | ✓ | | | | | |
| HTML | ✓ | ✓ | ✓ | | ✓ | |
| Jawa [6] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| JavaScript | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Node.js | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PHP | ✓ | ✓ | ✓ | ✓ | ✓ [7] | ✓ |
| Python | ✓ | ✓ | ✓ | ✓ | ✓ [8] | ✓ |
| Ruby | ✓ | ✓ | ✓ | ✓ | ✓ [5] | |
| Shell script | ✓ | ✓ | ✓ | ✓ | ✓ [5] | |

| Bahasa | Penyorotan sintaks [1] | UI penjalanan [2] | Tampilan garis besar | Petunjuk kode dan linting | Penyelesaian kode | Debugging [3] |
|---|---|---|---|---|---|---|
| TypeScript[9] | ✓ | ✓ | ✓ | ✓ | ✓ | |

Catatan

[1] AWS Cloud9 IDE menyediakan penyorotan sintaks untuk lebih banyak bahasa. Untuk daftar lengkap, di bilah menu IDE, pilih Tampilan, Sintaks.

[2] Anda dapat menjalankan program atau skrip dengan mengklik tombol untuk bahasa yang ditandai dengan ✓, tanpa menggunakan baris perintah. Untuk bahasa yang tidak ditandai dengan ✓ atau tidak ditampilkan pada bilah menu Run, Run With di IDE, Anda dapat membuat runner untuk bahasa tersebut. Untuk instruksi, lihat Membuat Builder atau Runner.

[3] Anda dapat menggunakan alat bawaan IDE untuk men-debug program atau skrip untuk bahasa yang ditandai dengan ✓. Untuk petunjuk, lihat Men-debug Kode Anda.

[4] Fitur ini dalam status eksperimental untuk bahasa ini. Itu tidak sepenuhnya diimplementasikan dan tidak didokumentasikan atau didukung.

[5] Fitur ini hanya mendukung fungsi lokal untuk bahasa ini.

[6] Dukungan yang ditingkatkan untuk fitur Java SE 11 dapat diaktifkan di lingkungan AWS Cloud9 EC2 pengembangan dengan 2 GiB atau lebih memori. Untuk informasi selengkapnya, lihat Peningkatan dukungan untuk pengembangan Java.

[7] Untuk menentukan jalur yang akan digunakan AWS Cloud9 untuk menyelesaikan kode PHP kustom, di AWS Cloud9 IDE nyalakan pengaturan Project, PHP Support, Enable PHP code completion in Preferences, dan kemudian tambahkan path ke kode kustom ke pengaturan Project, PHP Support, PHP Completion Include Paths.

[8] Untuk menentukan jalur yang akan digunakan AWS Cloud9 untuk menyelesaikan kode Python kustom, di AWS Cloud9 IDE nyalakan Proyek, Dukungan Python, Aktifkan pengaturan penyelesaian kode Python di Preferences, dan kemudian tambahkan path ke kode kustom ke Project, Python Support, pengaturan PYTHONPATH.

[9] AWS Cloud9 IDE memberikan dukungan tambahan untuk beberapa bahasa pemrograman, seperti TypeScript (versi 3.7.5 didukung dalam AWS Cloud9 IDE), dalam konteks proyek bahasa. Untuk informasi selengkapnya, lihat Bekerja dengan Proyek Bahasa.

## Versi bahasa pemrograman yang didukung di AWS Cloud9 Integrated Development Environment (IDE)

Tabel di bawah ini menguraikan versi bahasa pemrograman mana yang didukung AMIs secara spesifik di AWS Cloud9 IDE. Ubuntu 18 menggunakan EOL pada tahun 2023 dan sebagai hasilnya versi bahasa pemrograman tidak dapat diperbarui. AWS Cloud9

| Bahasa | Amazon Linux 2023 | Amazon Linux 2 | Ubuntu 18 | Ubuntu 22 |
|---|---|---|---|---|
| Python3 | 3.9 | 3.8 | 3.6 | 3.10 |
| TypeScript | 3.7.5 | 3.7.5 | 3.7.5 | 3.7.5 |
| PHP | 8.2 | 8.2 | 7.2 | 8.1 |
| Ruby | 3.2 | 3.0 | 3.0 | 3.2 |
| Java | 11, 17 | 11 | 11 | 11, 17 |
| Python2 | N/A | 2.7 | N/A | N/A |
| C++* | 23 | 17 | 17 | 23 |
| Go | 1.20 | 1.20 | 1.9 | 1.21 |
| CoffeeScript | 2.7 | 2.7 | 2.7 | 2.7 |

*Anda dapat menjalankan perintah berikut untuk mengkompilasi file C++ menggunakan versi bahasa pemrograman yang ingin Anda gunakan:

```
g++ -std=c++[version-number] "$file" -o "$file.o"
```

# Dukungan bahasa yang ditingkatkan di AWS Cloud9 IDE

AWS Cloud9 memberikan dukungan yang disempurnakan untuk meningkatkan pengalaman pengembangan Anda saat membuat kode dengan bahasa berikut:

- Java: Ekstensi memungkinkan menyediakan fitur seperti penyelesaian kode, linting untuk kesalahan, tindakan khusus konteks, dan opsi debugging.
- TypeScript: Proyek bahasa menawarkan akses ke fitur produktivitas yang ditingkatkan untuk. TypeScript

Topik


- Peningkatan dukungan untuk pengembangan Java
- TypeScript Dukungan dan fitur yang ditingkatkan


## Peningkatan dukungan untuk pengembangan Java

AWS Cloud9 menyediakan dukungan bahasa yang disempurnakan untuk meningkatkan pengalaman pengembangan Anda saat bekerja dengan Java. Fitur produktivitas utama termasuk penyelesaian kode, linting untuk kesalahan, lensa kode, dan opsi debugging seperti breakpoint dan stepping.

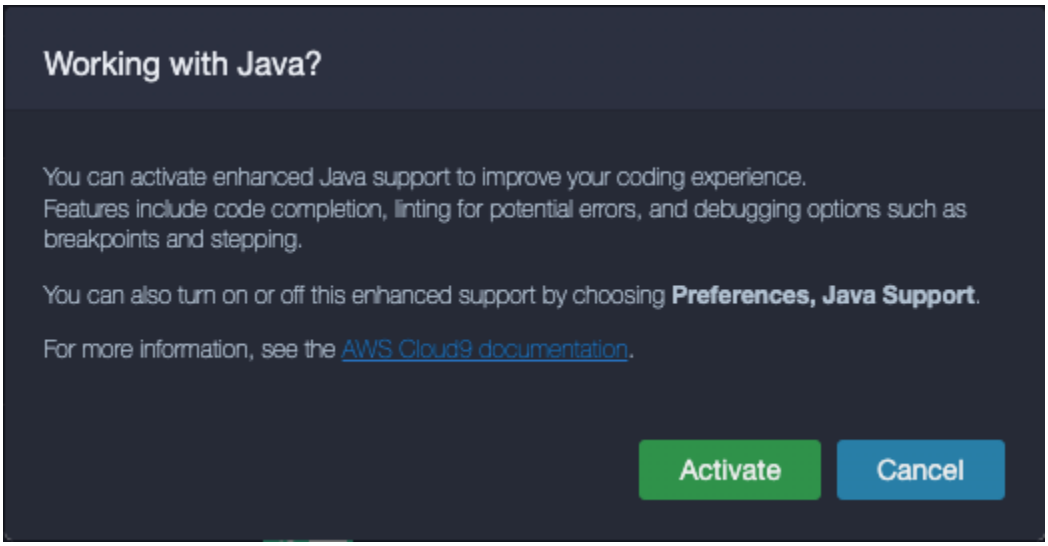
> **Important**
>
> Fitur produktivitas yang disempurnakan hanya tersedia untuk lingkungan AWS Cloud9 pengembangan yang terhubung ke EC2 instans Amazon.
> Selain itu, untuk memastikan pengalaman IDE yang optimal saat menggunakan dukungan bahasa yang disempurnakan untuk Java, instans EC2 komputasi Amazon yang mendukung AWS Cloud9 lingkungan Anda memerlukan 2 GiB atau lebih memori. Jika AWS Cloud9 mendeteksi bahwa instans EC2 komputasi Anda tidak memiliki RAM yang cukup, Anda tidak ditawari opsi untuk mengaktifkan fitur yang disempurnakan untuk Java.

### Mengaktifkan dan menyesuaikan dukungan Java yang disempurnakan

Opsi untuk mengaktifkan dukungan yang disempurnakan untuk Java secara otomatis ditampilkan jika kondisi berikut terpenuhi:

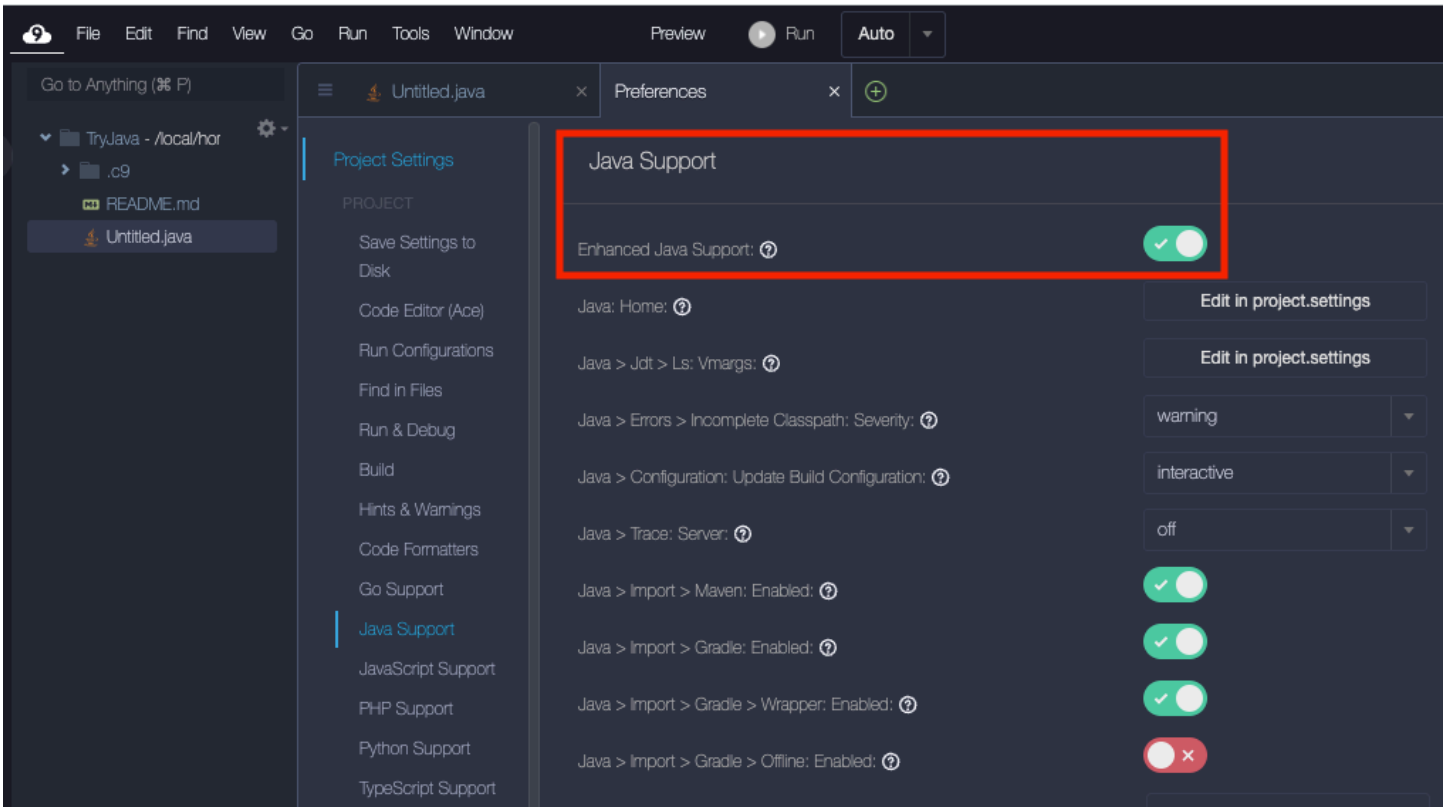
- AWS Cloud9 Lingkungan Anda terhubung ke EC2 instans Amazon dengan memori 2 GiB atau lebih.
- Anda bekerja dengan file yang terkait dengan pengembangan Java. AWS Cloud9 memeriksa nama dan ekstensi file berikut:`*.java`, `*.gradle` (terkait dengan alat build Gradle), dan `pom.xml` (terkait dengan alat build Apache Maven).
- Anda bekerja di AWS Cloud9 lingkungan yang dibuat setelah 11 Desember 2020. Saat ini, tidak mungkin menggunakan fitur produktivitas Java di lingkungan pengembangan yang dibuat sebelum tanggal ini.

Jika kondisi ini terpenuhi, kotak dialog akan ditampilkan untuk menanyakan apakah Anda ingin mengaktifkan fitur produktivitas tambahan untuk pengkodean dan debugging Java. Jika Anda memilih Aktifkan, Anda dapat mulai menggunakan fitur di IDE.




> **Note**
>
> EC2 Instans Amazon yang diluncurkan saat Anda membuat AWS Cloud9 lingkungan telah menginstal Amazon Coretto 11. Amazon Coretto adalah distribusi tanpa biaya, multiplatform, siap produksi dari Open Java Development Kit (OpenJDK). Ini berarti Anda dapat mulai mengembangkan dan menjalankan aplikasi Java di AWS Cloud9 out-of-the-box.

Anda juga dapat secara manual mengaktifkan dan menonaktifkan bahasa yang disempurnakan dan dukungan debugging menggunakan antarmuka. AWS Cloud9 Pilih Preferensi, Dukungan Java, Dukungan Java yang Ditingkatkan.




Dukungan yang disempurnakan untuk pengembangan Java di AWS Cloud9 disediakan oleh dua ekstensi ke IDE:

- Dukungan Bahasa untuk Java (TM) oleh Red Hat
- Debugger untuk Java

AWS Cloud9 Antarmuka memberi Anda akses ke berbagai pengaturan yang menyesuaikan kinerja ekstensi ini. Untuk mengubah pengaturan ekstensi, pilih Preferensi, Dukungan Java.

Untuk informasi rinci tentang pengaturan ini, lihat ReadMe halaman versi yang diinstal di repositori ekstensi GitHub :

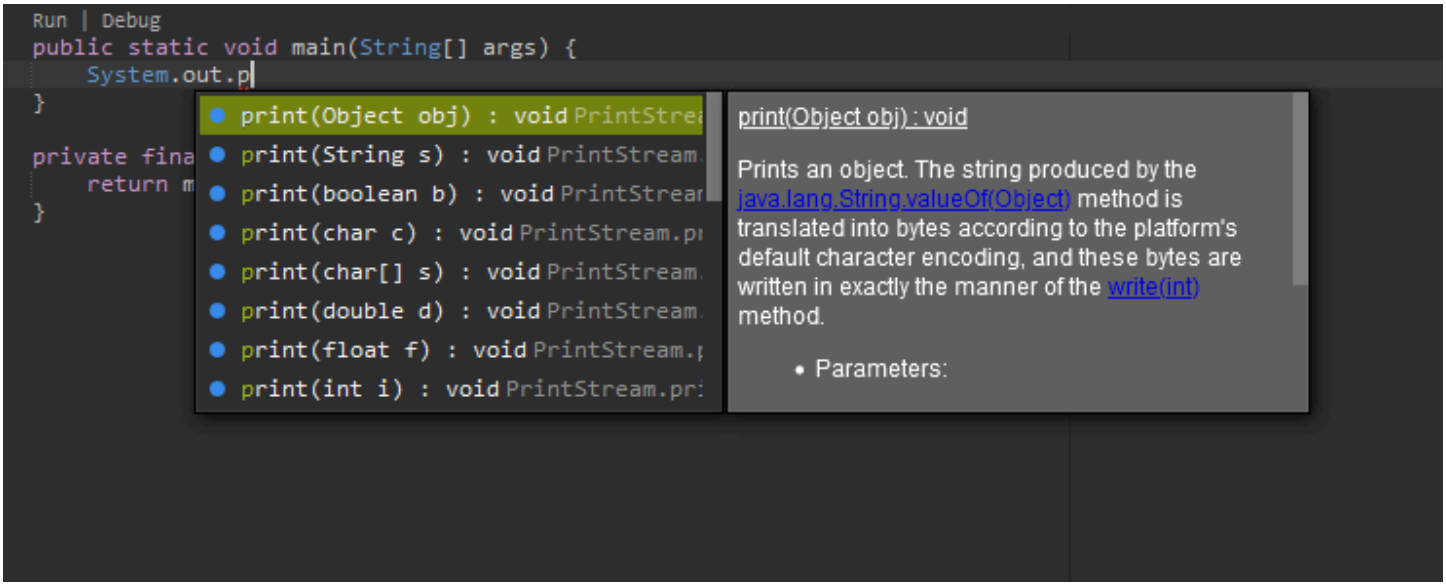
- Dukungan Bahasa untuk Java (TM) oleh Red Hat
- Debugger untuk Java

## Sorotan fitur

Setelah Anda mengaktifkan dukungan Java yang disempurnakan, Anda dapat menggunakan berbagai fitur peningkatan produktivitas.

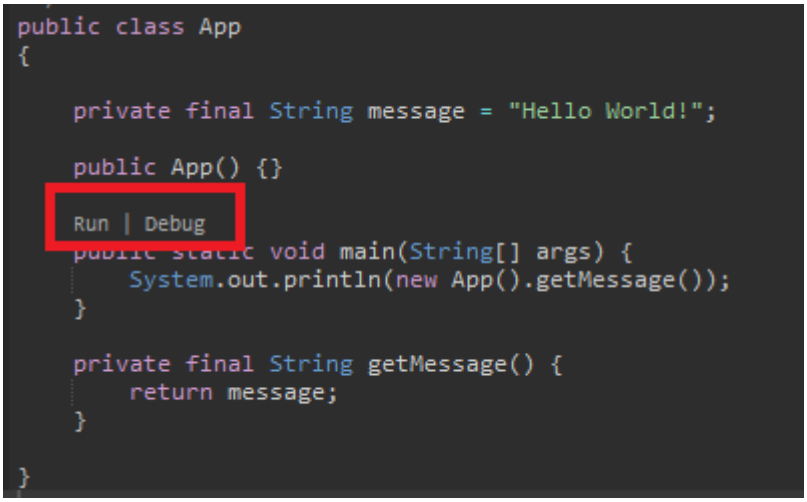
### Penyelesaian kode

Dengan penyelesaian kode, editor membuat saran sadar konteks berdasarkan kode yang Anda ketik. Misalnya, jika Anda mengetik operator titik (“.”) setelah nama objek, editor akan menampilkan metode atau properti yang tersedia untuk objek tersebut.




### Lensa kode

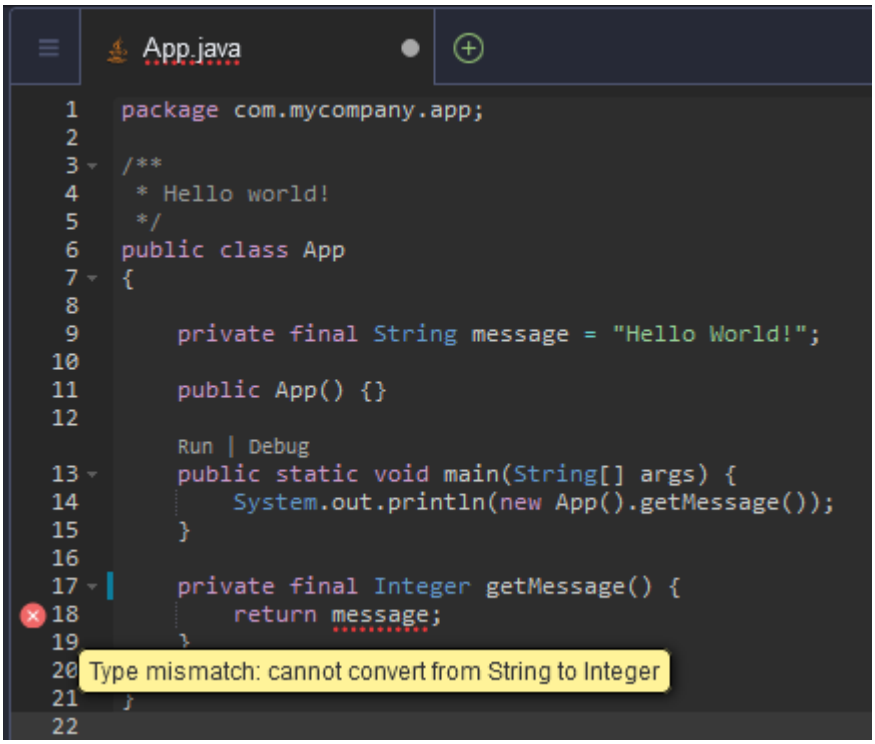
Lensa kode memungkinkan Anda mengakses tindakan spesifik konteks secara langsung di kode sumber. Untuk pengembangan Java, lensa kode memfasilitasi pengujian unit dengan memungkinkan Anda menjalankan dan men-debug metode tertentu.




### Linting kode

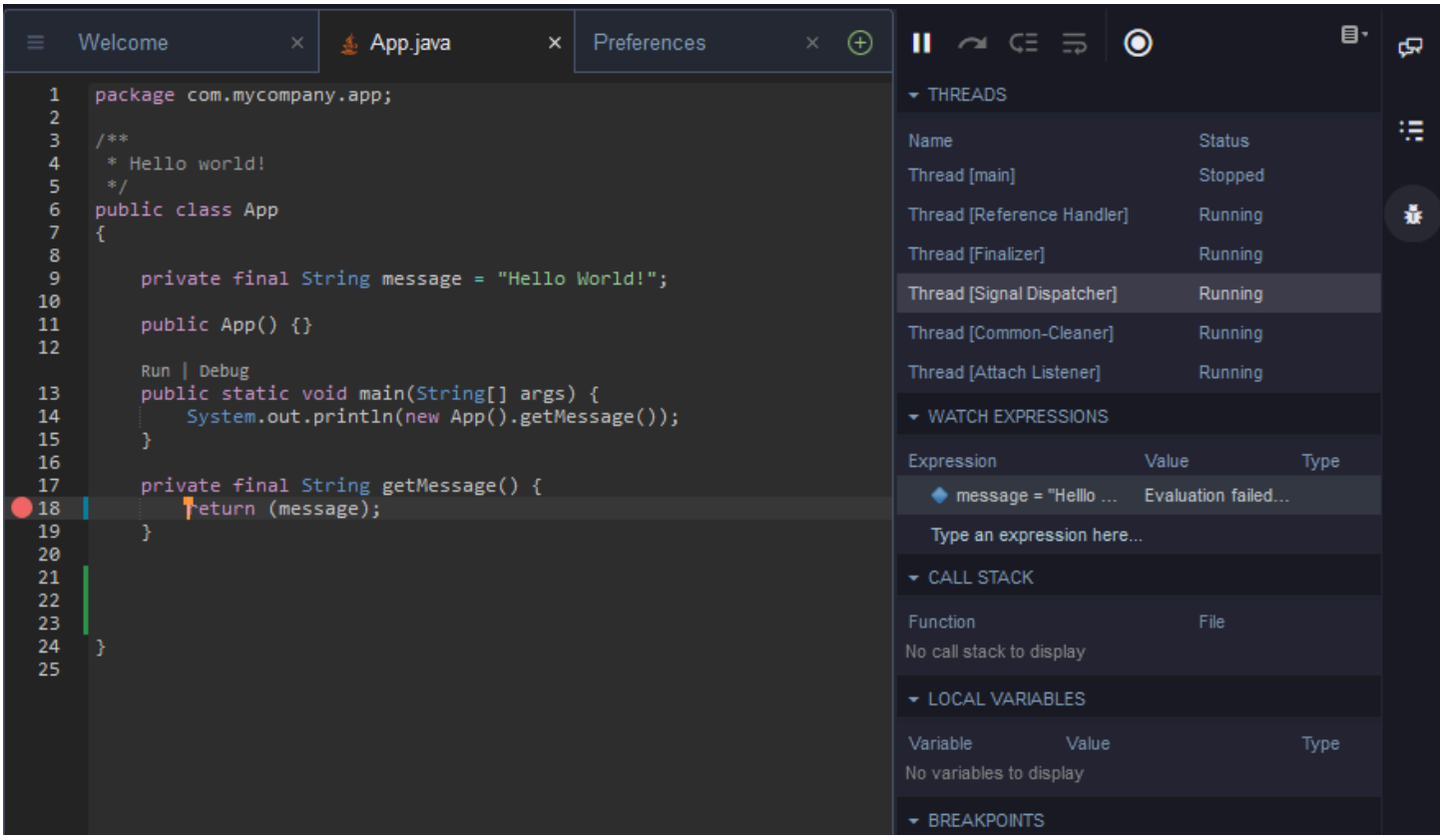
Code linting menjelaskan bagaimana editor menyoroti potensi kesalahan dalam kode Anda bahkan sebelum Anda membuatnya. Misalnya, alat linting memanggil jika Anda mencoba menggunakan

variabel yang tidak diinisialisasi atau mencoba menetapkan nilai ke variabel yang mengharapkan tipe yang berbeda.




Opsi debugging

Anda dapat menerapkan breakpoint dan ekspresi tontonan. Tetapkan breakpoint Anda dalam kode sumber dan tampilkan panel debugger untuk menentukan kondisi yang relevan.

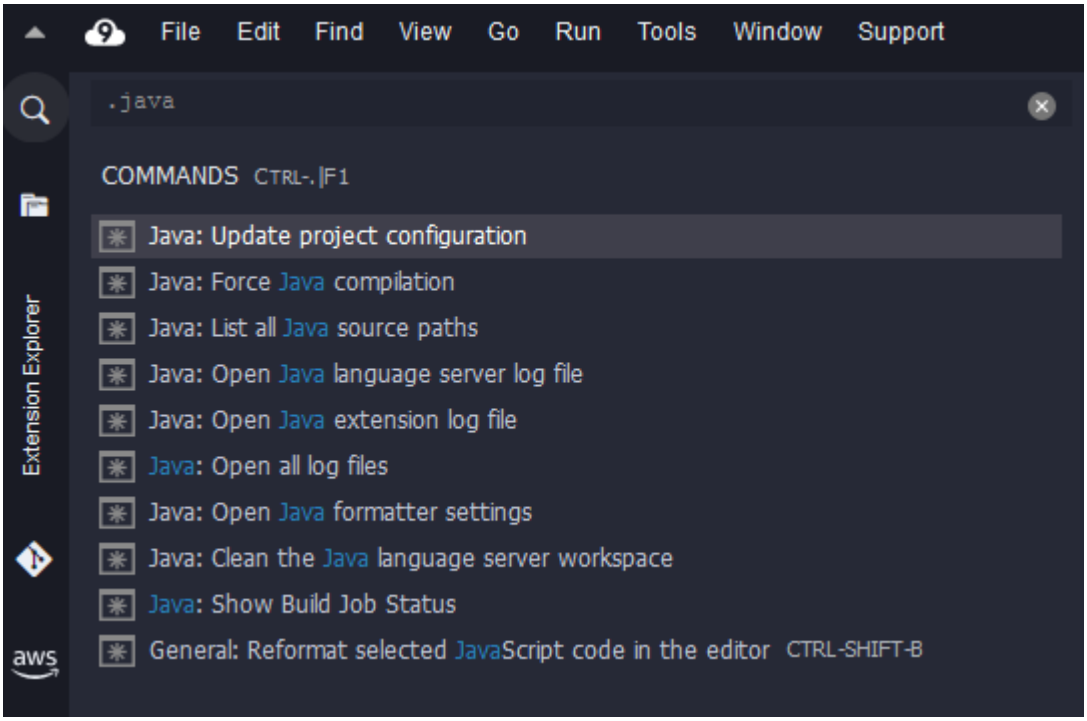



Debugging menggunakan file konfigurasi

Anda juga dapat mengontrol konfigurasi debugging Anda dengan menggunakan konfigurasi peluncuran dan tugas yang AWS Cloud9 mendukung melalui `launch.json` dan file `tasks.json` konfigurasi. Untuk contoh konfigurasi peluncuran dan bagaimana mereka dapat digunakan, lihat konfigurasi debug Java.

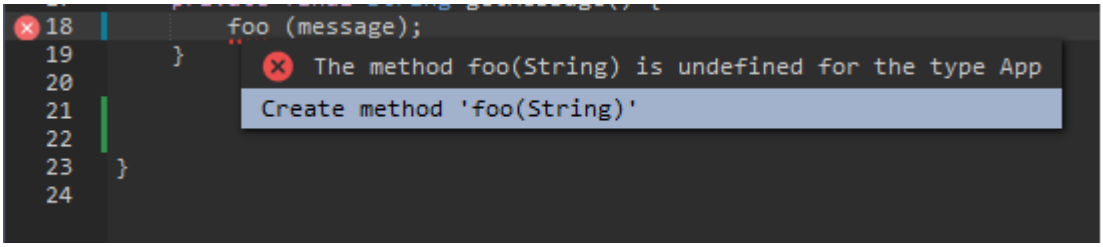
Perintah Java

Anda dapat menjalankan perintah dari panel AWS Cloud9 perintah dengan menekan Ctrl+. atau F1. Kemudian filter perintah yang relevan dengan memasukkan “java”.



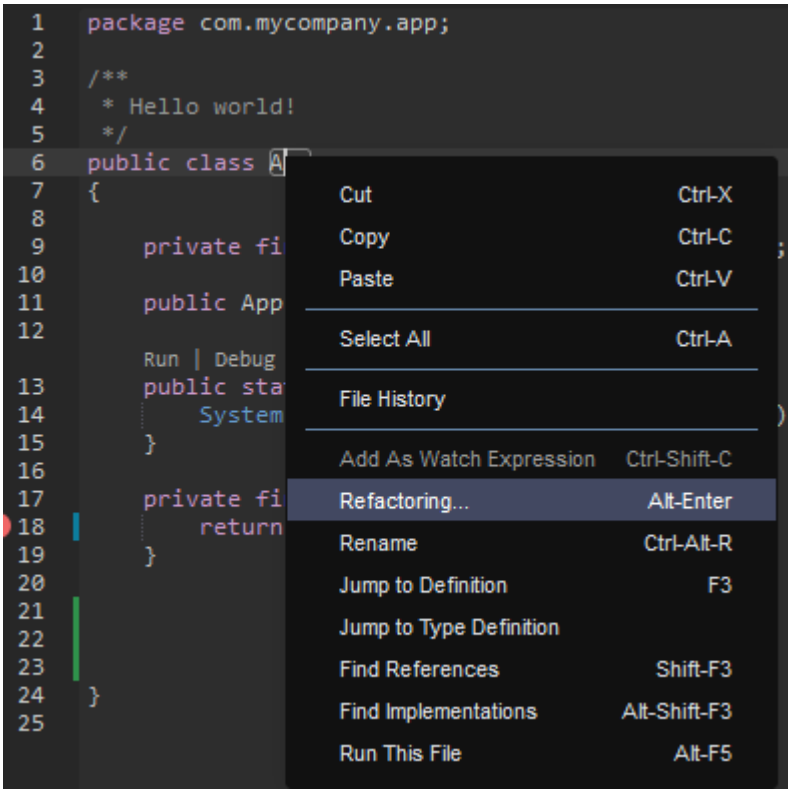

Perbaikan cepat

Dengan perbaikan cepat, Anda dapat mengatasi kesalahan yang disebabkan oleh penggunaan variabel yang tidak dideklarasikan atau metode yang tidak ditentukan dengan membuat rintisan untuk elemen yang hilang.



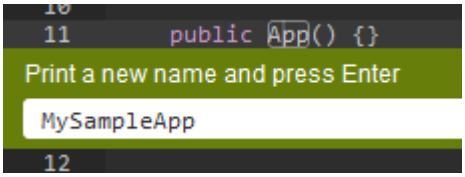

Refactoring

Refactoring memungkinkan Anda untuk merestrukturisasi kode Anda tanpa mengubah perilakunya. Untuk mengakses opsi seperti mengatur impor atau membuat konstruktor, buka menu konteks (klik kanan) untuk item tersebut dan pilih Refactoring.




Renaming

Mengganti nama adalah fitur refactoring yang memungkinkan Anda untuk dengan mudah memodifikasi nama variabel, fungsi, dan kelas yang dipilih di mana pun mereka muncul dalam kode dengan satu tindakan. Untuk mengubah nama, buka menu konteks (klik kanan) untuk item tersebut dan pilih Ganti nama. Mengganti nama memengaruhi setiap instance nama dalam kode Anda.




## Alat opsional untuk pengembangan Java

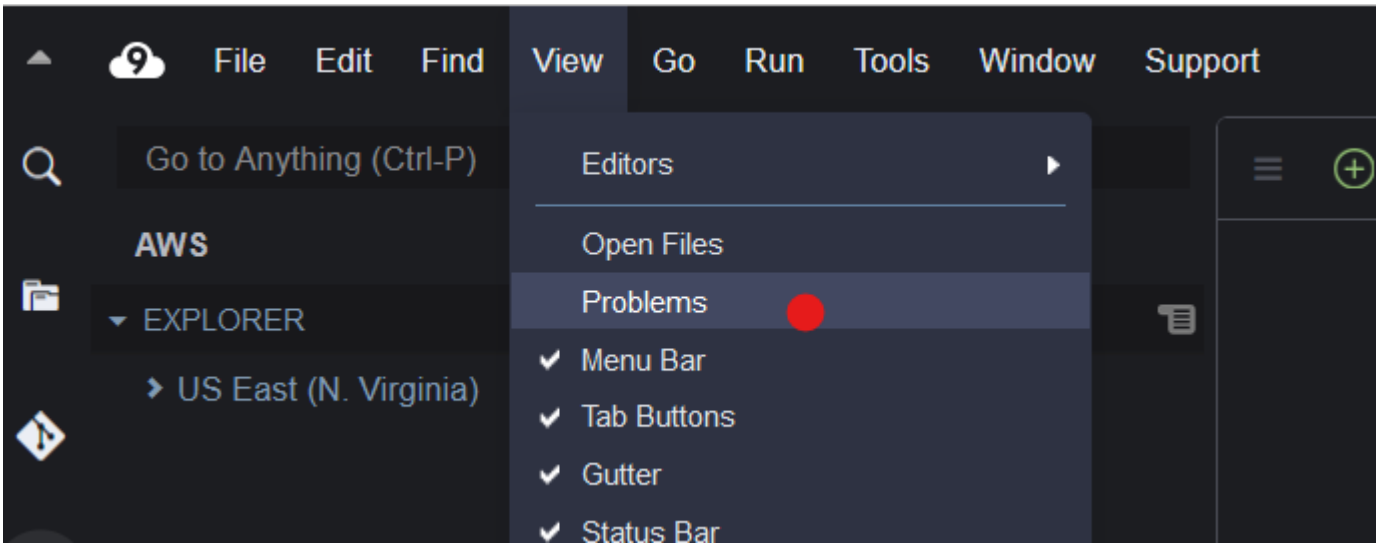
Ekstensi yang menyediakan dukungan Java yang disempurnakan mencakup fitur yang memungkinkan Anda mengintegrasikan alat otomatisasi Gradle dan Maven ke dalam pengembangan proyek Anda. Alat-alat ini tidak diinstal sebelumnya di lingkungan AWS Cloud9 pengembangan Anda. Untuk informasi selengkapnya tentang menginstal dan menggunakan alat build opsional ini, lihat sumber daya berikut:

- Gradle: Panduan memulai

- Maven: Maven dalam 5 menit

### Masalah tab untuk ekstensi Java

Anda dapat melihat dan memecahkan masalah dengan proyek java Anda dalam AWS Cloud9 lingkungan Anda di tab Masalah IDE. AWS Cloud9 Untuk menampilkan tab Masalah dari AWS Cloud9 IDE, pilih Lihat dan pilih Masalah dari bilah menu.




Anda juga dapat membuka tab Masalah dengan memilih ikon + di dalam konsol dan memilih Open Problems. Ketika Anda memilih masalah dari tab, itu membuka file yang terpengaruh dan menampilkan detail masalah.

## TypeScript Dukungan dan fitur yang ditingkatkan

AWS Cloud9 IDE memungkinkan Anda menggunakan proyek bahasa untuk mengakses fitur produktivitas yang ditingkatkan TypeScript. Proyek bahasa adalah kumpulan file, folder, dan pengaturan terkait di IDE untuk lingkungan AWS Cloud9 pengembangan.

Untuk menggunakan IDE untuk membuat proyek bahasa di lingkungan Anda, lihat Buat Proyek Bahasa.

### Fitur produktivitas proyek yang tersedia

AWS Cloud9 IDE menyediakan fitur produktivitas proyek berikut untuk TypeScript.

Pelengkapan otomatis

Saat Anda mengetik di dalam file di editor, daftar simbol ditampilkan di titik penyisipan untuk konteks tersebut, jika ada simbol yang tersedia di sana.

Untuk menyisipkan simbol dari daftar pada titik penyisipan, jika simbol belum dipilih, pilih dengan menggunakan tombol panah atas atau bawah, lalu tekan `Tab`.

Sebelum Anda menekan `Tab`, Anda mungkin melihat tip layar yang berisi informasi tentang simbol yang Anda pilih, jika informasi tersedia.

Untuk menutup daftar tanpa memasukkan simbol, tekan `Esc`.

Ikon selokan

Ikon mungkin muncul di selokan untuk file yang aktif. Ikon ini menyoroti kemungkinan masalah seperti peringatan dan kesalahan dalam kode sebelum Anda menjalankannya.

Untuk informasi selengkapnya tentang suatu masalah, hentikan penunjuk Anda pada ikon masalah tersebut.

Perbaikan Cepat

Di file aktif di editor, Anda dapat menampilkan informasi tentang kesalahan dan peringatan pengkodean, dengan kemungkinan perbaikan yang dapat Anda terapkan secara otomatis ke kode itu. Untuk menampilkan informasi kesalahan atau peringatan dan perbaikan yang mungkin, pilih bagian mana pun dari kode yang memiliki garis bawah putus-putus merah (untuk kesalahan), atau garis bawah putus-putus abu-abu (untuk peringatan). Atau, dengan kursor terletak pada kode yang memiliki garis bawah putus-putus berwarna merah atau abu-abu, tekan `Option-Enter` (untuk macOS), atau `Alt-Enter` (untuk Linux atau Windows). Untuk menerapkan perbaikan yang diusulkan, pilih perbaikan dalam daftar, atau gunakan tombol panah untuk memilih perbaikan lalu tekan `Enter`. Untuk mengaktifkan atau menonaktifkan pemilihan perbaikan cepat dengan klik mouse, pilih AWS Cloud9, Preferensi, Pengaturan Pengguna, Bahasa, Petunjuk & Peringatan, Tampilkan Perbaikan Cepat yang Tersedia saat Klik.

Temukan Referensi

Dalam file aktif di editor, Anda dapat menampilkan semua referensi ke simbol pada titik penyisipan, jika IDE memiliki akses ke referensi tersebut.

Untuk melakukan ini, pada titik penyisipan di mana saja di dalam simbol, jalankan perintah **`Find References`** . Sebagai contoh:

- Klik kanan pada titik penyisipan, lalu pilih Temukan Referensi.
- Pada bilah menu, pilih Pergi, Temukan Referensi.
- Tekan `Shift-F3` secara default untuk macOS, Windows, atau Linux.

Jika referensi tersedia, panel terbuka di atas file aktif, di samping simbol itu. Panel berisi daftar file tempat simbol direferensikan. Panel menampilkan referensi pertama dalam daftar. Untuk menampilkan referensi yang berbeda, pilih referensi tersebut dalam daftar.

Untuk menutup panel, pilih ikon tutup (X) di panel, atau tekan `Esc`.

Perintah **`Find References`** mungkin dinonaktifkan, atau mungkin tidak berfungsi seperti yang diharapkan, dalam kondisi berikut:

- Tidak ada referensi ke simbol tersebut di proyek file aktif.
- IDE tidak dapat menemukan beberapa atau semua referensi simbol itu dalam proyek file yang aktif.
- IDE tidak memiliki akses ke satu atau lebih lokasi di mana simbol itu direferensikan dalam proyek file aktif.

Pergi ke Definisi

Dalam file aktif di editor, Anda dapat pergi dari simbol ke tempat simbol tersebut didefinisikan, jika IDE memiliki akses ke definisi tersebut.

Untuk melakukan ini, pada titik penyisipan di mana saja di dalam simbol, jalankan perintah **`Jump to Definition`** . Sebagai contoh:

- Klik kanan pada titik penyisipan, lalu pilih Lompat ke Definisi.
- Pada bilah menu, pilih Pergi, Lompat ke Definisi.
- Tekan `F3` secara default untuk macOS, Windows, atau Linux.

Jika definisi tersedia, titik penyisipan beralih ke definisi tersebut, meskipun definisi tersebut berada dalam file terpisah.

Perintah **`Jump to Definition`** mungkin dinonaktifkan, atau mungkin tidak berfungsi seperti yang diharapkan, dalam kondisi berikut:

- Simbol tersebut merupakan simbol primitif untuk bahasa tersebut.

- IDE tidak dapat menemukan lokasi definisi di proyek file aktif.
- IDE tidak memiliki akses ke lokasi definisi di proyek file aktif.

Pergi ke Simbol

Anda dapat pergi ke simbol tertentu dalam sebuah proyek, sebagai berikut.

1. Jadikan salah satu file dalam proyek aktif dengan membukanya di editor. Jika file sudah terbuka, pilih tabnya di editor untuk membuat file tersebut aktif.
2. Jalankan perintah **`Go to Symbol`** . Sebagai contoh:

   - Pilih tombol jendela Pergi (ikon kaca pembesar). Di kotak Pergi ke Apapun, ketik `@`, lalu mulailah mengetik simbol.
   - Pada bilah menu, pilih Pergi, Pergi ke Simbol. Di jendela Pergi, mulailah mengetik simbol setelah @.
   - Tekan `Command-2` atau `Command-Shift-O` secara default untuk macOS, atau `Ctrl-Shift-O` secara default untuk Windows atau Linux. Di jendela Pergi, mulailah mengetik simbol setelah @.

   Misalnya, untuk menemukan semua simbol dalam proyek dengan nama `toString`, mulailah mengetik `@toString` (atau mulai mengetik `toString` setelah @, jika @ sudah ditampilkan).
3. Jika Anda melihat simbol yang Anda inginkan dalam daftar Simbol, pilih dengan mengkliknya. Atau gunakan tombol panah atas atau panah bawah untuk memilihnya, lalu tekan `Enter`. Titik penyisipan kemudian beralih ke simbol itu.

Jika simbol yang ingin Anda tuju tidak ada dalam proyek file aktif, prosedur ini mungkin tidak berfungsi seperti yang diharapkan.

## Buat Proyek Bahasa

Gunakan prosedur berikut untuk membuat proyek bahasa yang akan bekerja dengan fitur produktivitas proyek yang didukung di AWS Cloud9 IDE.

> **Note**
>
> Kami merekomendasikan Anda menggunakan fitur produktivitas proyek yang didukung pada file yang merupakan bagian dari proyek bahasa. Meskipun Anda dapat menggunakan

beberapa fitur produktivitas proyek yang didukung pada file yang bukan bagian dari proyek, fitur tersebut mungkin berperilaku dengan hasil yang tidak disangka.
Misalnya, Anda mungkin menggunakan IDE untuk mencari referensi dan definisi dari dalam file di tingkat root lingkungan yang bukan bagian dari proyek. IDE kemudian mungkin hanya mencari di seluruh file pada tingkat root yang sama. Ini mungkin mengakibatkan tidak ada referensi atau definisi yang ditemukan, meskipun referensi atau definisi tersebut sebenarnya ada dalam proyek bahasa di tempat lain di lingkungan yang sama.

Buat Proyek TypeScript Bahasa

1. Pastikan Anda telah TypeScript menginstal di lingkungan. Untuk informasi selengkapnya, lihat Langkah 1: Pasang alat yang diperlukan di TypeScript tutorial untuk AWS Cloud9.
2. Dari sesi terminal di IDE untuk lingkungan, alihkan ke direktori tempat Anda ingin membuat proyek. Jika direktori tidak ada, buatlah lalu alihkan ke sana. Sebagai contoh, perintah berikut membuat direktori dengan nama `my-demo-project` di root lingkungan (dalam `~/environment`), lalu beralih ke direktori tersebut.

   ```
   mkdir ~/environment/my-demo-project
   cd ~/environment/my-demo-project
   ```

3. Di root direktori tempat Anda ingin membuat proyek, jalankan TypeScript kompiler dengan **`--init`**opsi.

   ```
   tsc --init
   ```

   Jika perintah ini berhasil, TypeScript compiler membuat `tsconfig.json` file di root direktori untuk proyek. Anda dapat menggunakan file ini untuk menentukan berbagai pengaturan proyek, seperti opsi TypeScript kompiler dan file tertentu untuk disertakan atau dikecualikan dari proyek.

   Untuk informasi selengkapnya tentang file `tsconfig.json`, lihat berikut ini:

   - tsconfig.json Ikhtisar di situs web. TypeScript
   - Skema tsconfig.json di situs web json.schemastore.org.

# Referensi perintah bilah menu untuk AWS Cloud9 IDE

Daftar berikut menjelaskan perintah menu bar default di AWS Cloud9 IDE. Jika bilah menu tidak terlihat, pilih bilah tipis di sepanjang tepi atas IDE untuk menampilkannya.


- AWS Cloud9 menu
- Menu berkas
- Edit menu
- Temukan menu
- Lihat menu
- Buka menu
- Jalankan menu
- Menu alat
- Menu jendela
- Menu Support
- Menu pratinjau
- Perintah bilah menu lainnya


## AWS Cloud9 menu

| Perintah | Deskripsi |
| --- | --- |
| Preferensi | Lakukan salah satu hal berikut ini:<br>• Membuka tab Preferensi jika belum terbuka.<br>• Menjadikan tab Preferensi aktif jika terbuka tetapi tidak aktif.<br>• Menyembunyikan tab Preferensi jika aktif.<br><br>Lihat Bekerja dengan Pengaturan Proyek, Bekerja dengan Pengaturan Pengguna, Bekerja dengan Keybindings, Bekerja dengan Tema, dan Bekerja dengan Skrip Inisialisasi. |

| Perintah | Deskripsi |
| --- | --- |
| Pergi ke Dashboard Anda | Buka AWS Cloud9 konsol di tab browser web terpisah. Lihat Membuat Lingkungan, Membuka Lingkungan, Mengubah Pengaturan Lingkungan, dan Menghapus Lingkungan. |
| Halaman Selamat Datang | Membuka tab Selamat Datang. |
| Buka Pengaturan Proyek Anda | Membuka file `project.settings` untuk lingkungan saat ini. Lihat Bekerja dengan Pengaturan Proyek. |
| Buka Pengaturan Pengguna Anda | Membuka file `user.settings` untuk pengguna saat ini. Lihat Bekerja dengan Pengaturan Pengguna. |
| Buka Keymap Anda | Membuka file `keybindings.settings` untuk pengguna saat ini. Lihat Bekerja dengan Keybinding. |
| Buka Skrip Init Anda | Membuka file `init.js` untuk pengguna saat ini. Lihat Bekerja dengan Skrip Inisialisasi. |
| Buka Stylesheet Anda | Membuka file `styles.css` untuk pengguna saat ini. Lihat Bekerja dengan Tema. |

## Menu berkas

| Perintah | Deskripsi |
| --- | --- |
| File Baru | Membuat file baru. |
| Baru Dari Template | Membuat file baru, berdasarkan templat file yang dipilih. |
| Buka | Menampilkan dan membuka jendela Navigasi. |

| Perintah | Deskripsi |
| --- | --- |
| Buka Terbaru | Membuka file yang dipilih. |
| Simpan | Menyimpan file saat ini. |
| Simpan Sebagai | Menyimpan file saat ini dengan nama file, lokasi, atau keduanya yang berbeda. |
| Simpan Semua | Menyimpan semua file yang belum disimpan. |
| Kembali ke Disimpan | Membuang perubahan untuk file saat ini sejak terakhir disimpan. |
| Kembalikan Semua ke Tersimpan | Membuang perubahan untuk semua file yang belum disimpan sejak terakhir disimpan. |
| Tampilkan Riwayat Revisi File | Melihat dan mengelola perubahan pada file saat ini di editor. Lihat Bekerja dengan Revisi File. |
| Unggah File Lokal | Menampilkan kotak dialog Unggah File, yang memungkinkan Anda menyeret file dari komputer lokal ke lingkungan. |
| Unduh Proyek | Menggabungkan file di lingkungan menjadi file .zip, yang dapat Anda unduh ke komputer lokal Anda. |
| Akhir Baris | Menggunakan akhiran baris Windows (carriage return plus umpan baris) atau Unix (hanya umpan baris). |
| Tutup File | Menutup file saat ini. |
| Tutup Semua File | Menutup semua file terbuka. |

# Edit menu

| Perintah | Deskripsi |
| --- | --- |
| Batalkan | Mengurungkan tindakan terakhir. |
| Ulangi | Mengulangi tindakan terakhir yang dibatalkan. |
| Potong | Memindahkan seleksi ke clipboard. |
| Salin | Menyalin seleksi ke clipboard. |
| Tempel | Menyalin isi clipboard ke titik seleksi. |
| Mode Keyboard | Serangkaian keybinding untuk digunakan, seperti `Default`, `Vim`, `Emacs`, atau `Sublime`. Lihat Bekerja dengan Keybinding. |
| Seleksi, Pilih Semua | Memilih semua konten yang dapat dipilih. |
| Seleksi, Dibagi Menjadi Garis | Menambahkan kursor pada akhir baris saat ini. |
| Seleksi, Seleksi Tunggal | Mengosongkan semua seleksi sebelumnya. |
| Seleksi, Beberapa Pilihan, Tambahkan Kursor Naik | Menambahkan kursor satu baris di atas kursor aktif. Jika kursor sudah ditambahkan, menambahkan kursor lain di atasnya. |
| Seleksi, Beberapa Pilihan, Tambahkan Kursor Bawah | Menambahkan kursor satu baris di bawah kursor aktif. Jika kursor sudah ditambahkan, menambahkan kursor lain di bawahnya. |
| Seleksi, Beberapa Pilihan, Pindahkan Kursor Aktif Ke Atas | Menambahkan kursor satu baris kedua di atas kursor aktif. Jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke atas satu baris. |
| Seleksi, Beberapa Pilihan, Pindahkan Kursor Aktif Ke Bawah | Menambahkan kursor satu baris kedua di bawah kursor aktif. Jika kursor kedua sudah |

| Perintah | Deskripsi |
| --- | --- |
| | ditambahkan, memindahkan kursor kedua ke bawah satu baris. |
| Seleksi, Beberapa Pilihan, Tambahkan Pencocokan Seleksi Berikutnya | Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi. |
| Seleksi, Beberapa Pilihan, Tambahkan Pencocokan Seleksi Sebelumnya | Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi. |
| Seleksi, Beberapa Pilihan, Gabungkan Rentang Pilihan | Menambahkan kursor pada akhir baris saat ini. |
| Seleksi, Pilih Word Right | Menyertakan kata berikutnya ke kanan kursor dalam seleksi. |
| Seleksi, Pilih Kata Kiri | Menyertakan kata berikutnya ke kiri kursor dalam seleksi. |
| Seleksi, Pilih ke Akhir Baris | Menyertakan dari kursor ke akhir baris saat ini dalam seleksi |
| Seleksi, Pilih untuk Mulai Baris | Menyertakan dari awal baris saat ini hingga kursor dalam seleksi. |
| Seleksi, Pilih untuk Dokumen Akhir | Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi. |
| Seleksi, Pilih untuk Mulai Dokumen | Menyertakan dari kursor naik ke awal file saat ini dalam seleksi. |
| Garis, Indentasi | Meng-indent seleksi satu tab. |
| Garis, Outdent | Meng-outdent seleksi satu tab. |
| Garis, Pindahkan Line Up | Memindahkan seleksi ke atas satu baris. |
| Garis, Pindahkan Garis Ke Bawah | Memindahkan seleksi ke bawah satu baris. |

| Perintah | Deskripsi |
| --- | --- |
| Baris, Salin Baris Naik | Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas. |
| Baris, Salin Garis Bawah | Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah. |
| Garis, Hapus Garis | Menghapus isi dari baris saat ini. |
| Baris, Hapus ke Ujung Garis | Menghapus dari kursor ke akhir baris saat ini. |
| Baris, Hapus ke Garis Mulai | Menghapus dari awal baris saat ini hingga kursor. |
| Garis, Garis Terpisah | Memindahkan isi kursor ke akhir baris, ke barisnya sendiri. |
| Teks, Hapus Kata Kanan | Menghapus kata di sebelah kanan kursor. |
| Teks, Hapus Kata Kiri | Menghapus kata di sebelah kiri kursor. |
| Teks, Sejajarkan | Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar. |
| Teks, Surat Transpose | Menranspos seleksi. |
| Teks, ke huruf besar | Mengubah seleksi ke semua huruf besar. |
| Teks, Untuk Hurse Kecil | Mengubah seleksi ke semua huruf kecil. |
| Komentar, Beralih Komentar | Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada. |
| Kode Lipat, Alihkan Lipat | Melipat kode, atau menghapus lipatan kode jika sudah ada. |
| Kode Lipat, Buka | Membuka lipatan kode yang dipilih. |

| Perintah | Deskripsi |
| --- | --- |
| Kode Lipat, Lipat Lainnya | Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini. |
| Kode Lipat, Lipat Semua | Melipat semua elemen yang mungkin dapat dilipat. |
| Kode Lipat, Buka Semua | Membuka lipatan dari pelipatan kode untuk seluruh file. |
| Pemformatan Kode, Terapkan Pemformatan Kode | Format ulang JavaScript kode yang dipilih. |
| Pemformatan Kode, Bahasa Terbuka & Preferensi Pemformatan | Membuka bagian Pengaturan Proyek dari tab Preferensi untuk pengaturan bahasa pemrograman. |

## Temukan menu

Untuk informasi selengkapnya, lihat Menemukan dan Mengganti Teks.

| Perintah | Deskripsi |
| --- | --- |
| Temukan | Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi Temukan. |
| Temukan Berikutnya | Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir. |
| Temukan Sebelumnya | Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir. |

| Perintah | Deskripsi |
| --- | --- |
| Ganti | Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi Ganti Dengan. |
| Ganti Berikutnya | Mengganti kecocokan berikutnya untuk Temukan dengan Ganti Dengan di bilah temukan dan ganti untuk dokumen saat ini. |
| Ganti Sebelumnya | Mengganti kecocokan sebelumnya untuk Temukan dengan Ganti Dengan di bilah temukan dan ganti untuk dokumen saat ini. |
| Ganti Semua | Mengganti semua kecocokan untuk Temukan dengan Ganti Dengan di bilah temukan dan ganti untuk dokumen saat ini. |
| Temukan di File | Menampilkan bilah temukan dan ganti untuk banyak file. |

## Lihat menu

| Perintah | Deskripsi |
| --- | --- |
| Editor | Menampilkan editor yang dipilih. |
| Buka File | Menampilkan daftar Buka File di jendela Lingkungan, atau menyembunyikan jika ditampilkan. |
| Masalah | Tampilkan masalah apa pun dalam proyek Java untuk lingkungan di panel Masalah di terminal. Anda dapat memilih masalah untuk membuka file target. |

| Perintah | Deskripsi |
| --- | --- |
| Bilah Menu | Menampilkan bilah menu, atau menyembun yikan jika ditampilkan. |
| Tombol Tab | Menampilkan tab, atau menyembunyikan jika ditampilkan. |
| Selokan | Menampilkan gutter, atau menyembunyikan jika ditampilkan. |
| Bilah Status | Menampilkan bilah status, atau menyembun yikan jika ditampilkan. |
| Konsol | Menampilkan jendela Konsol, atau menyembun yikan jika ditampilkan. |
| Tata Letak, Tunggal | Menampilkan satu panel. |
| Tata Letak, Pemisahan Vertikal | Menampilkan dua panel, atas dan bawah. |
| Tata Letak, Pemisahan Horisontal | Menampilkan dua panel, berdampingan. |
| Tata Letak, Cross Split | Menampilkan empat panel dengan ukuran yang sama. |
| Tata letak, Split 1:2 | Menampilkan satu panel di sebelah kiri dan dua panel di sebelah kanan. |
| Tata letak, Split 2:1 | Menampilkan dua panel di sebelah kiri dan satu panel di sebelah kanan. |
| Ukuran Font, Tingkatkan Ukuran Font | Meningkatkan ukuran huruf. |
| Ukuran Font, Kurangi Ukuran Font | Mengurangi ukuran huruf. |
| Sintaksis | Menampilkan jenis sintaks untuk dokumen saat ini. |
| Tema | Menampilkan jenis tema IDE. |

| Perintah | Deskripsi |
| --- | --- |
| Garis Bungkus | Membungkus kata-kata ke tepi panel saat ini, atau menghentikan membungkus kata-kata jika sudah membungkus. |
| Bungkus Untuk Mencetak Margin | Membungkus kata-kata ke tepi margin cetak saat ini, atau menghentikan membungkus kata-kata jika sudah membungkus. |

## Buka menu

| Perintah | Deskripsi |
| --- | --- |
| Pergi Ke Apa Saja | Menampilkan jendela Pergi di mode Pergi ke Apa Pun. |
| Pergi ke Simbol | Menampilkan jendela Pergi di mode Pergi ke Simbol. |
| Pergi ke File | Menampilkan jendela Pergi di mode Pergi ke File. |
| Pergi ke Command | Menampilkan jendela Pergi di mode Pergi ke Perintah. |
| Pergi ke Baris | Menampilkan jendela Pergi di mode Pergi ke Baris. |
| Kesalahan Berikutnya | Pergi ke kesalahan berikutnya. |
| Kesalahan Sebelumnya | Pergi ke kesalahan sebelumnya. |
| Kata Benar | Pergi satu kata ke kanan. |
| Kata Kiri | Pergi satu kata ke kiri. |
| Garis Akhir | Pergi ke akhir baris saat ini. |

| Perintah | Deskripsi |
| --- | --- |
| Garis Mulai | Pergi ke awal baris saat ini. |
| Lompat ke Definisi | Membuka definisi variabel atau fungsi di kursor. |
| Lompat ke Penjepit yang Cocokkan | Pergi ke simbol yang cocok dalam cakupan saat ini. |
| Gulir ke Seleksi | Menggulir seleksi ke tampilan yang lebih baik. |

## Jalankan menu

| Perintah | Deskripsi |
| --- | --- |
| Jalankan . | Menjalankan atau men-debug aplikasi saat ini. |
| Jalankan Terakhir | Menjalankan atau men-debug file yang dijalankan terakhir. |
| Jalankan Dengan | Menjalankan atau men-debug menggunak an runner yang dipilih. Lihat Bekerja dengan Builder, Runner, dan Debugger. |
| Jalankan Sejarah | Melihat riwayat penjalanan. |
| Jalankan Konfigurasi | Memilih konfigurasi penjalanan untuk menjalankan atau men-debug, atau membuat atau mengelola konfigurasi penjalanan. Lihat Bekerja dengan Builder, Runner, dan Debugger. |
| Tampilkan Debugger saat Istirahat | Ketika menjalankan kode mencapai breakpoint, menampilkan jendela Debugger. |
| Membangun | Membangun file saat ini. |
| Batalkan Build | Berhenti membangun file saat ini. |

| Perintah | Deskripsi |
| --- | --- |
| Membangun Sistem | Membangun menggunakan sistem membangun yang dipilih. |
| Tampilkan Hasil Build | Menampilkan hasil membangun terkait. |
| Secara Otomatis Membangun File yang Didukung | Membangun file yang didukung secara otomatis. |
| Simpan Semua di Build | Saat membangun, simpan semua file terkait yang belum disimpan. |

## Menu alat

| Perintah | Deskripsi |
| --- | --- |
| Strip Ruang Trailing | Memotong spasi di akhir baris. |
| Pratinjau, Pratinjau File | Mempratinjau dokumen saat ini di tab pratinjau. |
| Pratinjau, Pratinjau Aplikasi yang Berjalan | Mempratinjau aplikasi saat ini di tab peramban web terpisah. |
| Pratinjau, Konfigurasikan URL Pratinjau | Membuka bagian Pengaturan Proyek dari tab Preferensi ke kotak Jalankan & Debug, URL Pratinjau. |
| Pratinjau, Tampilkan Server Aktif | Menampilkan daftar alamat server aktif yang tersedia di kotak dialog Daftar Proses. |
| Daftar Proses | Menampilkan kotak dialog Daftar Proses. |
| Tampilkan Pelengkapan Otomatis | Menampilkan menu konteks penyelesaian kode. |
| Ubah nama Variabel | Memulai memfaktor ulang (refactor) ganti nama untuk seleksi. |

| Perintah | Deskripsi |
| --- | --- |
| Alihkan Perekaman Makro | Memulai perekaman penekanan tombol, menghentikannya jika sudah merekam. |
| Mainkan Makro | Memainkan penekanan tombol yang direkam sebelumnya. |

## Menu jendela

| Perintah | Deskripsi |
| --- | --- |
| Go | Menampilkan jendela Pergi, atau menyembun yikan jika ditampilkan. |
| Terminal Baru | Membuka tab Terminal baru. |
| Jendela Segera Baru | Membuka tab Segera baru. |
| Bagikan | Menampilkan kotak dialog Bagikan lingkungan ini. |
| Pemasang | Menampilkan kotak dialog Penginstal AWS Cloud9 . |
| Berkolaborasi | Menampilkan jendela Kolaborasikan, atau menyembunyikan jika ditampilkan. |
| Garis Besar | Menampilkan jendela Outline, atau menyembun yikan jika ditampilkan. |
| AWS Sumber Daya | Menampilkan jendela  Sumber Daya AWS , atau menyembunyikan jika ditampilkan. |
| Lingkungan | Menampilkan jendela Lingkungan, atau menyembunyikan jika ditampilkan. |

| Perintah | Deskripsi |
| --- | --- |
| Debugger | Menampilkan jendela Debugger, atau menyembunyikan jika ditampilkan. |
| Navigasi, Tab ke Kanan | Pergi satu tab ke kanan. |
| Navigasi, Tab ke Kiri | Pergi satu tab ke kiri. |
| Navigasi, Tab Berikutnya dalam Sejarah | Pergi ke tab berikutnya. |
| Navigasi, Tab Sebelumnya dalam Sejarah | Pergi ke tab sebelumnya. |
| Navigasi, Pindahkan Tab ke Kanan | Memindahkan tab saat ini ke kanan. Jika tab sudah berada di paling kanan, membuat tab terpisah di sana. |
| Navigasi, Pindahkan Tab ke Kiri | Memindahkan tab saat ini ke kiri. Jika tab sudah berada di paling kiri, membuat tab terpisah di sana. |
| Navigasi, Pindahkan Tab ke Atas | Memindahkan tab saat ini ke atas satu panel. Jika tab sudah berada di paling atas, membuat tab terpisah di sana. |
| Navigasi, Pindahkan Tab ke Bawah | Memindahkan tab saat ini ke bawah satu panel. Jika tab sudah berada di paling bawah, membuat tab terpisah di sana. |
| Navigasi, Pergi ke Panel ke Kanan | Pergi satu panel ke kanan. |
| Navigasi, Pergi ke Panel ke Kiri | Pergi satu panel ke kiri. |
| Navigasi, Pergi ke Pane to Up | Pergi satu panel ke atas. |
| Navigasi, Pergi ke Panel ke Bawah | Pergi satu panel ke bawah. |
| Navigasi, Beralih Antara Editor dan Terminal | Beralih antara editor dan tab Terminal. |
| Navigasi, Panel Berikutnya dalam Sejarah | Pergi ke panel berikutnya. |

| Perintah | Deskripsi |
| --- | --- |
| Navigasi, Panel Sebelumnya dalam Sejarah | Pergi ke panel sebelumnya. |
| Tata Letak Tersimpan, Simpan | Menyimpan tata letak saat ini. Untuk beralih ke tata letak ini nanti, pilih Tata Letak Tersimpan, LAYOUT-ID. |
| Tata Letak Tersimpan, Simpan dan Tutup Semua | Menyimpan tata letak saat ini, dan kemudian menutup semua tab dan panel. |
| Tata Letak Tersimpan, Tampilkan Tata Letak Tersimpan di Pohon File | Menampilkan semua tata letak tersimpan di jendela Lingkungan. |
| Tab, Tutup Panel | Menutup panel saat ini. |
| Tab, Tutup Semua Tab Di Semua Panel | Menutup semua tab yang terbuka di semua panel. |
| Tab, Tutup Semua Tapi Tab Saat Ini | Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini. |
| Tab, Split Pane dalam Dua Baris | Memisahkan panel saat ini menjadi dua panel, atas dan bawah. |
| Tab, Split Pane dalam Dua Kolom | Memisahkan panel saat ini menjadi dua panel, kiri dan kanan. |
| Preset, IDE Lengkap | Mengalihkan ke mode IDE penuh. |
| Preset, Editor Minimal | Mengalihkan ke mode editor minimal. |
| Preset, Mode Luhur | Mengalihkan ke mode Sublim. |

## Menu Support

| Perintah | Deskripsi |
| --- | --- |
| Halaman Selamat Datang | Membuka tab Selamat Datang. |

| Perintah | Deskripsi |
| --- | --- |
| Dapatkan Bantuan (Komunitas) | Membuka situs web komunitas AWS Cloud9 online di tab browser web terpisah. |
| Baca Dokumentasi | Membuka Panduan Pengguna AWS Cloud9 di tab peramban web terpisah. |

## Menu pratinjau

| Perintah | Deskripsi |
| --- | --- |
| File Pratinjau | Mempratinjau dokumen saat ini di tab pratinjau. |
| Pratinjau Aplikasi Berjalan | Mempratinjau aplikasi saat ini di tab peramban web terpisah. |
| Konfigurasikan URL Pratinjau | Membuka bagian Pengaturan Proyek dari tab Preferensi ke kotak Jalankan & Debug, URL Pratinjau. |
| Tampilkan Server Aktif | Menampilkan daftar alamat server aktif yang tersedia di kotak dialog Daftar Proses. |

## Perintah bilah menu lainnya

| Perintah | Deskripsi |
| --- | --- |
| Jalankan . | Menjalankan atau men-debug aplikasi saat ini. |
| Bagikan | Membuka kotak dialog Bagikan lingkungan ini. |
| Preferensi (ikon roda gigi) | Membuka tab Preferensi. |

# Menemukan dan Mengganti Teks di AWS Cloud9 IDE

Anda dapat menggunakan bilah find dan replace di AWS Cloud9 Integrated Development Environment (IDE) untuk menemukan dan mengganti teks dalam satu file atau beberapa file.


- Temukan Teks dalam Satu File
- Ganti Teks dalam Satu File
- Temukan Teks dalam Beberapa File
- Ganti Teks di Beberapa File
- Temukan dan Ganti Opsi


## Menemukan Teks di File Tunggal

1. Buka file yang ingin Anda temukan teksnya. Jika file sudah terbuka, pilih tab file untuk mengaktifkan file tersebut.
2. Pada bilah menu, pilih Temukan, Temukan.
3. Dalam bilah temukan dan ganti, untuk Temukan, ketikkan teks yang ingin Anda temukan.
4. Untuk menentukan opsi temukan tambahan, lihat Opsi Menemukan dan Mengganti.
5. Jika ada kecocokan, 0 dari 0 di kotak Temukan berubah menjadi angka bukan nol. Jika ada kecocokan, editor pergi ke kecocokan pertama. Jika ada lebih dari satu kecocokan, untuk menuju ke kecocokan berikutnya, pilih panah kanan di kotak Temukan atau pilih Temukan, Temukan Berikutnya di bilah menu. Untuk menuju ke kecocokan sebelumnya, pilih panah kiri di kotak Temukan atau pilih Temukan, Temukan Sebelumnya di bilah menu.

## Mengganti Teks di File Tunggal

1. Buka file yang ingin Anda ganti teksnya. Jika file sudah terbuka, pilih tab file untuk mengaktifkan file tersebut.
2. Pada bilah menu, pilih Temukan, Ganti.
3. Dalam bilah temukan dan ganti, untuk Temukan, ketikkan teks yang ingin Anda temukan.
4. Untuk Ganti Dengan, ketikkan teks yang ingin Anda ganti teksnya dengan di Temukan.
5. Untuk menentukan opsi temukan dan ganti tambahan, lihat Opsi Menemukan dan Mengganti.
6. Jika ada kecocokan, 0 dari 0 di kotak Temukan berubah menjadi angka bukan nol. Jika ada kecocokan, editor pergi ke kecocokan pertama. Jika ada lebih dari satu kecocokan, untuk menuju

ke kecocokan berikutnya, pilih panah kanan di kotak Temukan atau pilih Temukan, Temukan Berikutnya di bilah menu. Untuk menuju ke kecocokan sebelumnya, pilih panah kiri di kotak Temukan atau pilih Temukan, Temukan Sebelumnya di bilah menu.

7. Untuk mengganti kecocokan saat ini dengan teks di Ganti Dengan dan kemudian pergi ke kecocokan berikutnya, pilih Ganti. Untuk mengganti semua kecocokan dengan teks di Ganti Dengan, pilih Ganti Semua.

## Menemukan Teks di Beberapa File

1. Pada bilah menu, pilih Temukan, Temukan di File.
2. Dalam bilah temukan dan ganti, untuk Temukan, ketikkan teks yang ingin Anda temukan.
3. Untuk menentukan opsi temukan tambahan, lihat Opsi Menemukan dan Mengganti.
4. Dalam kotak di sebelah kanan tombol Temukan (kotak dengan `*.*, -.*`), ketikkan kumpulan file apa pun untuk disertakan atau dikecualikan dalam pencarian. Sebagai contoh:
   - Kosong, `*`, atau `*.*`: Temukan semua file.
   - `my-file.txt`: Menemukan hanya file bernama `my-file.txt`.
   - `my*`: Menemukan hanya file dengan nama file yang dimulai dengan `my`.
   - `my*.txt`: Menemukan hanya file dengan nama file yang dimulai dengan `my` dan yang memiliki ekstensi file `.txt`.
   - `my*.htm*`: Menemukan semua file dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`.
   - `my*.htm, my*.html`: Menemukan semua file dengan nama file yang dimulai dengan `my` dan ekstensi file `.htm` atau `.html`.
   - `-my-file.txt`: Jangan mencari file bernama `my-file.txt`.
   - `-my*`: Jangan mencari file apa pun yang dimulai dengan `my`.
   - `-my*.htm*`: Jangan mencari file apa pun dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`.
   - `my*.htm*, -my*.html`: Mencari semua file dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`. Namun, jangan mencari file apa pun dengan nama file yang dimulai dengan `my` dan ekstensi file `.html`.
5. Dalam daftar tarik-turun di samping kotak sebelumnya, pilih salah satu dari berikut ini untuk lebih membatasi penemuan hanya ke lokasi tertentu:
   - Lingkungan: Menemukan hanya file di jendela Lingkungan.

- Proyek (tidak termasuk .gitignore'd): Menemukan file apa pun di lingkungan, kecuali untuk file atau jenis file yang tercantum dalam file `.gitignore` di lingkungan, jika file `.gitignore` ada.
- Seleksi:: Menemukan hanya file yang saat ini dipilih di jendela Lingkungan.

> **Note**
>
> Untuk lebih membatasi pencarian hanya pada satu folder, pilih folder di jendela Lingkungan dan kemudian pilih Seleksi. Atau, Anda dapat mengeklik kanan folder di jendela Lingkungan, dan kemudian memilih Cari di Folder Ini pada menu konteks.

- Favorit: Menemukan hanya file dalam daftar Favorit di jendela Lingkungan.
- File Aktif: Menemukan hanya file aktif.
- File Terbuka: Menemukan hanya file dalam daftar File Terbuka di jendela Lingkungan.

6. Pilih Temukan.
7. Untuk pergi ke file yang berisi kecocokan, klik dua kali nama file pada tab Hasil Pencarian. Untuk pergi ke kecocokan tertentu, klik dua kali kecocokan di tab Hasil Pencarian.

## Mengganti Teks di Beberapa File

1. Pada bilah menu, pilih Temukan, Temukan di File.
2. Dalam bilah temukan dan ganti, untuk Temukan, ketikkan teks yang ingin Anda temukan.
3. Untuk menentukan opsi temukan tambahan, lihat Opsi Menemukan dan Mengganti.
4. Dalam kotak di sebelah kanan tombol Temukan (kotak dengan `*.*, -.*`), ketikkan kumpulan file apa pun untuk disertakan atau dikecualikan dalam pencarian. Sebagai contoh:
   - Kosong, `*`, atau `*.*`: Semua file.
   - `my-file.txt`: Hanya file bernama `my-file.txt`.
   - `my*`: Hanya file dengan nama file yang dimulai dengan `my`.
   - `my*.txt`: Hanya file dengan nama file yang dimulai dengan `my` dan yang memiliki ekstensi file `.txt`.
   - `my*.htm*`: Semua file dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`.
   - `my*.htm, my*.html`: Semua file dengan nama file yang dimulai dengan `my` dan ekstensi file `.htm` atau `.html`.

- `-my-file.txt`: Jangan mencari file bernama `my-file.txt`.
- `-my*`: Jangan mencari file apa pun yang dimulai dengan `my`.
- `-my*.htm*`: Jangan mencari file apa pun dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`.
- `my*.htm*, -my*.html`: Mencari semua file dengan nama file yang dimulai dengan `my` dan ekstensi file yang dimulai dengan `.htm`. Namun, jangan mencari file apa pun dengan nama file yang dimulai dengan `my` dan ekstensi file `.html`.

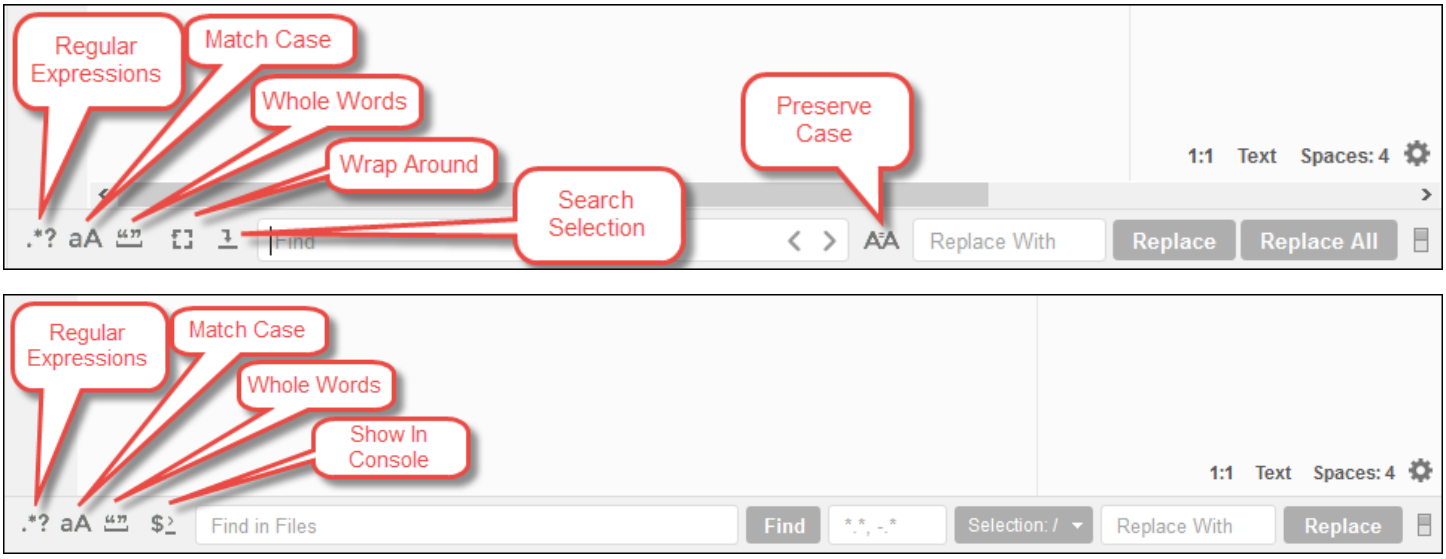
5. Dalam daftar tarik-turun di samping kotak sebelumnya, pilih salah satu dari berikut ini untuk lebih membatasi penemuan hanya ke lokasi tertentu:
   - Lingkungan: Hanya file di jendela Lingkungan.
   - Proyek (tidak termasuk .gitignore'd): File apa pun di lingkungan, kecuali untuk file atau jenis file yang tercantum dalam file `.gitignore` di lingkungan, jika file `.gitignore` ada.
   - Seleksi: /: Hanya file yang saat ini dipilih.
   - Favorit: Hanya file dalam daftar Favorit di jendela Lingkungan.
   - File Aktif: Hanya file aktif.
   - File Terbuka: Hanya file dalam daftar File Terbuka di jendela Lingkungan.
6. Untuk Ganti Dengan, ketikkan teks yang ingin Anda ganti dengan Temukan.
7. Pilih Ganti.

> **Note**
>
> Operasi penggantian terjadi segera di semua file dalam cakupan. Operasi ini tidak dapat dibatalkan dengan mudah. Jika Anda ingin melihat apa yang akan diubah sebelum Anda memulai operasi penggantian, pilih Temukan sebagai gantinya.

8. Untuk pergi ke file yang berisi pengganti, klik dua kali nama file di tab Hasil Pencarian. Untuk pergi ke pengganti tertentu, klik dua kali pengganti di panel Hasil Pencarian.

## Opsi Menemukan dan Mengganti

Pilih salah satu tombol berikut pada bilah temukan dan ganti untuk memodifikasi operasi temukan dan ganti.




- Ekspresi Reguler: Menemukan teks yang cocok dengan ekspresi reguler yang ditentukan di Temukan atau Temukan di File. Lihat Menulis pola ekspresi reguler dalam topik Ekspresi JavaScript Reguler di Jaringan Pengembang Mozilla.
- Cocokkan Huruf: Menemukan teks yang cocok dengan huruf besar/kecil yang ditentukan di Temukan atau Temukan di File.
- Seluruh Kata: Menggunakan aturan karakter kata standar untuk menemukan teks di Temukan atau Temukan di File.
- Membungkus Sekitar: Untuk satu file saja, jangan berhenti di akhir atau awal file saat menuju kecocokan berikutnya atau sebelumnya.
- Cari Seleksi: Untuk satu file saja, menemukan hanya dalam seleksi.
- Tampilkan di Konsol: Untuk beberapa file, menampilkan tab Hasil Pencarian di Konsol, bukan di panel aktif.
- Pertahankan Huruf: Untuk satu file saja, mempertahankan huruf besar/kecil seperti yang berlaku saat mengganti teks.

## Mempratinjau file di IDE AWS Cloud9

Anda dapat menggunakan AWS Cloud9 IDE untuk melihat pratinjau file di lingkungan AWS Cloud9 pengembangan dari dalam IDE.

- Buka file untuk pratinjau
- Muat ulang pratinjau file

- Ubah jenis pratinjau file
- Buka pratinjau file di tab browser web terpisah
- Beralih ke pratinjau file yang berbeda

## Membuka file untuk pratinjau

Pilih salah satu opsi berikut di AWS Cloud9 IDE untuk membuka tab pratinjau file di lingkungan:

- Di jendela Lingkungan, buka menu konteks (klik kanan) untuk file yang ingin Anda pratinjau, lalu pilih Pratinjau.

  > **Note**
  >
  > Meskipun Anda dapat menggunakan pendekatan ini untuk mempratinjau file apa pun, pratinjau berfungsi paling baik dengan file yang memiliki ekstensi file berikut:
  >
  > - `.htm`
  > - `.html`
  > - `.pdf`
  > - `.svg`
  > - `.xhtml`
  > - File apa pun yang berisi konten dalam format Markdown.

- Buka file dengan salah satu ekstensi file berikut:
  - `.pdf`
  - `.svg`
- Dengan file yang ingin Anda pratinjau sudah terbuka dan aktif, pada bilah menu, pilih Pratinjau, Pratinjau File FILE_NAME. Atau pilih Alat, Pratinjau, Pratinjau File FILE_NAME, di mana FILE_NAME adalah nama file yang ingin Anda pratinjau.

  > **Note**
  >
  > Perintah ini hanya bekerja dengan jenis file berikut:
  >
  > - `.htm`
  > - `.html`
  > - `.markdown`

- `.md`
- `.pdf`
- `.svg`
- `.txt`: Pratinjau berfungsi paling baik jika konten file dalam format Markdown.
- `.xhtml`: Pratinjau berfungsi paling baik jika file berisi atau merujuk informasi presentasi konten.

> **Note**
>
> Menu Pengaturan Pratinjau di tab pratinjau file saat ini tidak berfungsi dan memilih salah satu perintah menunya tidak akan berpengaruh.

## Memuat ulang pratinjau file

Pada tab pratinjau file, pilih tombol Refresh (panah melingkar).

## Mengubah jenis pratinjau file

Pada tab pratinjau file, pilih salah satu dari berikut ini dari daftar jenis pratinjau:

- Peramban: Mempratinjau file dalam format peramban web, hanya untuk jenis file berikut:
  - `.htm`
  - `.html`
  - `.pdf`
  - `.svg`
  - `.xhtml`: Pratinjau berfungsi paling baik jika file berisi atau merujuk informasi presentasi konten.
- Konten Mentah (UTF-8): Mempratinjau konten asli file dalam format Unicode Transformation Format 8-bit (UTF-8). Ini mungkin menampilkan konten yang tidak diharapkan untuk beberapa jenis file.
- Markdown: Mempratinjau file apa pun yang berisi format Markdown. Mencoba untuk melihat pratinjau jenis file lainnya, tetapi mungkin menampilkan konten yang tidak diharapkan.

### Membuka pratinjau file di tab peramban web terpisah

Pada tab pratinjau file, pilih Pop Out Ke Jendela Baru.

### Beralih ke pratinjau file berbeda

Pada tab pratinjau file, ketikkan jalur ke jalur file yang berbeda di bilah alamat. Bilah alamat terletak di antara tombol Refresh dan daftar jenis pratinjau.

## Mempratinjau aplikasi yang sedang berjalan di IDE AWS Cloud9

Anda dapat menggunakan AWS Cloud9 Integrated Development Environment (IDE) untuk melihat pratinjau aplikasi yang sedang berjalan dari dalam IDE.

### Menjalankan aplikasi

Sebelum Anda dapat melihat pratinjau aplikasi Anda dari dalam IDE, aplikasi Anda harus berjalan di lingkungan AWS Cloud9 pengembangan. Itu harus menggunakan HTTP melalui port berikut:

- `8080`
- `8081`
- `8082`

Semua port di atas harus menggunakan alamat IP `127.0.0.1localhost`, atau`0.0.0.0`.

> **Note**
>
> Anda tidak diharuskan untuk menjalankan aplikasi Anda menggunakan HTTP melalui port `80808081`,, atau `8082` dengan alamat IP`127.0.0.1`,`localhost`, atau`0.0.0.0`. Namun, jika Anda tidak melakukannya, Anda tidak dapat melihat pratinjau aplikasi yang sedang berjalan dari dalam IDE.

> **Note**
>
> Aplikasi pratinjau dijalankan dalam IDE dan dimuat di dalam elemen iframe. Beberapa server aplikasi mungkin secara default memblokir permintaan yang berasal dari elemen

iframe, seperti X-Frame-Options header. Jika aplikasi pratinjau Anda tidak ditampilkan di tab pratinjau, pastikan server aplikasi Anda tidak melarang menampilkan konten dalam iframe.

Untuk menulis kode untuk menjalankan aplikasi Anda pada port dan alamat IP tertentu, lihat dokumentasi aplikasi Anda.

Untuk menjalankan aplikasi Anda, lihat Menjalankan Kode Anda.

Untuk menguji perilaku ini, tambahkan JavaScript kode berikut ke file yang diberi nama `server.js` di root lingkungan Anda. Kode ini menjalankan server menggunakan file yang bernama Node.js.

**Note**

Dalam contoh berikut, `text/html` adalah konten `Content-Type` yang dikembalikan. Untuk mengembalikan konten dalam format yang berbeda, tentukan yang berbeda`Content-Type`. Misalnya, Anda dapat menentukan `text/css` untuk format file CSS.

```
var http = require('http');
var fs = require('fs');
var url = require('url');

http.createServer( function (request, response) {
  var pathname = url.parse(request.url).pathname;
  console.log("Trying to find '" + pathname.substr(1) + "'...");

  fs.readFile(pathname.substr(1), function (err, data) {
    if (err) {
      response.writeHead(404, {'Content-Type': 'text/html'});
      response.write("ERROR: Cannot find '" + pathname.substr(1) + "'.");
      console.log("ERROR: Cannot find '" + pathname.substr(1) + "'.");
    } else {
      console.log("Found '" + pathname.substr(1) + "'.");
      response.writeHead(200, {'Content-Type': 'text/html'});
      response.write(data.toString());
    }
    response.end();
  });
}).listen(8080, 'localhost'); // Or 8081 or 8082 instead of 8080. Or '127.0.0.1'
 instead of 'localhost'.
```

Di root lingkungan Anda, Anda dapat menambahkan kode Python berikut ke file dengan nama seperti. `server.py` Dalam contoh berikut, server dijalankan menggunakan Python.

```
import os
import http.server
import socketserver

ip = 'localhost' # Or '127.0.0.1' instead of 'localhost'.
port = '8080' # Or '8081' or '8082' instead of '8080'.
Handler = http.server.SimpleHTTPRequestHandler
httpd = socketserver.TCPServer((ip, int(port)), Handler)
httpd.serve_forever()
```

Di root lingkungan Anda, tambahkan kode HTML berikut ke file yang diberi nama`index.html`.

```
<html>
  <head>
    <title>Hello Home Page</title>
  </head>
  <body>
    <p style="font-family:Arial;color:blue">Hello, World!</p>
  </body>
</html>
```

Untuk melihat output HTML dari file ini pada tab pratinjau aplikasi, jalankan `server.js` dengan Node.js atau `server.py` file dengan Python. Kemudian, ikuti langkah-langkah di bagian selanjutnya untuk melihat pratinjaunya. Pada tab pratinjau aplikasi, tambahkan `/index.html` ke akhir URL, lalu tekan `Enter`.

## Mempratinjau aplikasi yang sedang berjalan

Sebelum Anda melihat pratinjau aplikasi Anda, konfirmasikan hal berikut:

- Aplikasi Anda berjalan menggunakan protokol HTTP melalui port`8080`,`8081`, atau`8082`.
- Alamat IP aplikasi Anda di lingkungan adalah`127.0.0.1`,`localhost`, atau`0.0.0.0`.
- File kode aplikasi Anda terbuka dan aktif di AWS Cloud9 IDE.

Setelah Anda mengonfirmasi semua detail ini, pilih salah satu opsi berikut dari bilah menu:

- Pratinjau, Pratinjau Aplikasi yang Berjalan

- Alat, Pratinjau, Pratinjau Aplikasi yang Menjalankan

Salah satu opsi ini membuka tab pratinjau aplikasi dalam lingkungan, dan kemudian menampilkan output aplikasi pada tab.

> **Note**
>
> Jika tab pratinjau aplikasi menampilkan kesalahan atau kosong, ikuti langkah-langkah pemecahan masalah. Tab pratinjau aplikasi menampilkan kesalahan atau kosong Jika ketika Anda mencoba untuk melihat pratinjau aplikasi atau file, Anda mendapatkan pemberitahuan berikut “Fungsionalitas pratinjau dinonaktifkan karena browser Anda menonaktifkan cookie pihak ketiga”, ikuti langkah-langkah pemecahan masalah di. Pemberitahuan pratinjau aplikasi atau pratinjau file: "Cookie pihak ke tiga dinonaktifkan"

> **Note**
>
> Jika aplikasi belum berjalan, kesalahan muncul di tab pratinjau aplikasi. Untuk mengatasi masalah ini, jalankan atau restart aplikasi, lalu pilih perintah menu bar lagi.
> Misalkan, misalnya, aplikasi Anda tidak dapat berjalan di salah satu port atau IPs yang disebutkan. Atau, aplikasi Anda harus berjalan di lebih dari satu port ini secara bersamaan. Misalnya, aplikasi Anda harus berjalan pada port `8080` dan `3000` pada saat yang sama. Jika itu masalahnya, maka tab pratinjau aplikasi mungkin menampilkan kesalahan atau mungkin kosong. Ini karena tab pratinjau aplikasi dalam lingkungan hanya berfungsi dengan port sebelumnya dan. IPs Selain itu, aplikasi ini bekerja hanya dengan satu port pada satu waktu. Kami tidak menyarankan untuk berbagi URL di tab pratinjau aplikasi dengan orang lain. (URL dalam format berikut:`https://12a34567b8cd9012345ef67abcd890e1.vfs.cloud9.us-east-2.amazonaws.com/`. Dalam format ini, `12a34567b8cd9012345ef67abcd890e1` adalah ID yang AWS Cloud9 diberikan ke lingkungan. `us-east-2`adalah ID Wilayah AWS untuk lingkungan.) URL ini bekerja hanya ketika IDE untuk lingkungan terbuka dan aplikasi berjalan di peramban web yang sama.
> Jika Anda mencoba mengunjungi IP`127.0.0.1`,`localhost`, atau `0.0.0.0` dengan menggunakan tab pratinjau aplikasi di IDE atau di tab browser web terpisah di luar IDE, AWS Cloud9 IDE secara default mencoba masuk ke komputer lokal Anda, bukan instance atau server Anda sendiri yang terhubung ke lingkungan.

Untuk petunjuk tentang cara memberi orang lain pratinjau aplikasi yang sedang berjalan di luar IDE, lihatMembagikan aplikasi yang sedang berjalan melalui internet.

## Memuat ulang pratinjau aplikasi

Untuk memuat ulang pratinjau aplikasi, pilih opsi berikut:

Pada tab pratinjau aplikasi, pilih tombol Refresh (panah melingkar).

> **Note**
>
> Perintah ini tidak me-restart server. Ini hanya menyegarkan isi tab pratinjau aplikasi.

## Mengubah jenis pratinjau aplikasi

Untuk mengubah jenis pratinjau aplikasi, pilih opsi berikut:

Pada tab pratinjau aplikasi, pilih salah satu dari berikut ini dari daftar jenis pratinjau:

- Peramban: Pratinjau output dalam format peramban web.
- Konten Mentah (UTF-8): Upaya untuk melihat pratinjau output dalam format Unicode Transformation Format 8-bit (UTF-8), jika berlaku.
- Markdown: Mencoba untuk melihat pratinjau output dalam format Markdown, jika berlaku.

## Membuka pratinjau aplikasi di tab peramban web terpisah

Untuk membuka pratinjau aplikasi di tab browser web terpisah, pilih opsi berikut:

Pada tab pratinjau aplikasi, pilih Pop Out Ke Jendela Baru.

> **Note**
>
> AWS Cloud9 IDE juga harus berjalan di setidaknya satu tab lain di browser web yang sama. Jika tidak, pratinjau aplikasi tidak ditampilkan di tab browser web terpisah.
> AWS Cloud9 IDE juga harus berjalan di setidaknya satu tab lain di browser web yang sama. Jika tidak, pratinjau aplikasi tidak ditampilkan di tab browser web terpisah. Jika tab pratinjau aplikasi menampilkan kesalahan atau kosong, ikuti langkah-langkah pemecahan masalah. Pemberitahuan pratinjau aplikasi atau pratinjau file: "Cookie pihak ke tiga dinonaktifkan"

## Beralih ke URL pratinjau yang berbeda

Untuk beralih ke URL pratinjau yang berbeda, pilih opsi berikut:

Pada tab pratinjau aplikasi, masukkan jalur ke URL yang berbeda di bilah alamat. Bilah alamat terletak di antara tombol Refresh dan daftar jenis pratinjau.

## Membagikan aplikasi yang sedang berjalan melalui internet

Setelah Anda melihat pratinjau aplikasi yang berjalan, Anda dapat membuatnya tersedia untuk orang lain melalui internet.

Jika EC2 instans Amazon terhubung ke lingkungan Anda, ikuti langkah-langkah ini. Jika tidak, konsultasikan dokumentasi server Anda.

Topik


- Langkah 1: Dapatkan ID dan alamat IP instans
- Langkah 2: Siapkan grup keamanan untuk instans
- Langkah 3: Siapkan subnet untuk instans
- Langkah 4: Bagikan URL aplikasi yang sedang berjalan


### Langkah 1: Dapatkan ID dan alamat IP instans

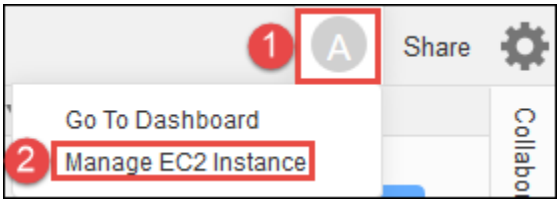
Pada langkah ini, Anda mencatat ID instans dan alamat IP publik untuk EC2 instans Amazon yang terhubung ke lingkungan. Anda memerlukan ID instans di langkah selanjutnya untuk mengizinkan permintaan aplikasi masuk. Kemudian, bagikan alamat IP publik kepada orang lain sehingga mereka dapat mengakses aplikasi yang sedang berjalan.

1. Dapatkan ID EC2 instans Amazon. Untuk mendapatkannya, lakukan salah satu hal berikut:

   - Dalam sesi terminal di AWS Cloud9 IDE untuk lingkungan, jalankan perintah berikut untuk mendapatkan ID EC2 instans Amazon.

     ```
     curl http://169.254.169.254/latest/meta-data/instance-id
     ```

     ID instance dalam format berikut:`i-12a3b456c789d0123`. Catat ID instans ini.
   - Di IDE untuk lingkungan, pada bilah menu, pilih ikon pengguna Anda, lalu pilih Kelola EC2 Instance.




Di EC2 konsol Amazon yang ditampilkan, catat ID instance yang ditampilkan di kolom ID Instance. ID instance dalam format ini:`i-12a3b456c789d0123`.

2. Dapatkan alamat IP publik EC2 instans Amazon. Untuk mendapatkannya, lakukan salah satu hal berikut:

   - Di IDE untuk lingkungan, pada bilah menu, pilih Bagikan. Di kotak dialog Bagikan lingkungan ini, catat alamat IP publik di kotak Aplikasi. Alamat IP publik dalam format ini:`192.0.2.0`.
   - Dalam sesi terminal di IDE untuk lingkungan, jalankan perintah berikut untuk mendapatkan alamat IP publik EC2 instans Amazon.

     ```
     curl http://169.254.169.254/latest/meta-data/public-ipv4
     ```

     Alamat IP publik dalam format ini:`192.0.2.0`. Catat alamat IP publik ini.
   - Di IDE untuk lingkungan, pada bilah menu, pilih ikon pengguna Anda, lalu pilih Kelola EC2 Instance. Di EC2 konsol Amazon yang menampilkan, pada tab Deskripsi, catat alamat IP publik untuk bidang IP IPv4 Publik. Alamat IP publik dalam format ini:`192.0.2.0`.

> **Note**
>
> Alamat IP publik aplikasi Anda dapat berubah kapan saja instans untuk aplikasi Anda dimulai ulang. Untuk mencegah alamat IP Anda berubah, alokasikan alamat IP Elastis. Kemudian, tetapkan alamat itu ke instance yang sedang berjalan. Untuk petunjuk, lihat Mengalokasikan Alamat IP Elastis dan Mengaitkan Alamat IP Elastis dengan Instans Berjalan di Panduan Pengguna Amazon EC2 . Mengalokasikan alamat IP Elastis dapat Akun AWS menyebabkan Anda dikenakan biaya. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

## Langkah 2: Siapkan grup keamanan untuk instans

Pada langkah ini, di EC2 konsol Amazon, siapkan grup EC2 keamanan Amazon untuk instance yang terhubung ke lingkungan. Atur untuk memungkinkan permintaan HTTP masuk melalui port 8080, 8081, atau 8082.

> **Note**
>
> Anda tidak diharuskan menjalankan menggunakan HTTP melalui port`8080`,`8081`, atau`8082`. Jika Anda tidak melakukan ini, Anda tidak dapat melihat pratinjau aplikasi yang sedang berjalan dari dalam IDE. Untuk informasi selengkapnya, lihat Mempratinjau aplikasi yang sedang berjalan. Jika tidak, jika Anda menjalankan protokol atau port yang berbeda, gantilah di langkah ini.
> Untuk lapisan keamanan tambahan, siapkan daftar kontrol akses jaringan (ACL) untuk subnet di VPC yang dapat digunakan instance. Untuk informasi selengkapnya tentang grup keamanan dan jaringan ACLs, lihat berikut ini:
>
> - Langkah 3: Siapkan subnet untuk instans
> - Keamanan di Panduan Pengguna Amazon VPC
> - Grup Keamanan untuk VPC Anda di Panduan Pengguna Amazon VPC
> - Jaringan ACLs di Panduan Pengguna VPC Amazon

1. Di IDE untuk lingkungan, pada bilah menu, pilih ikon pengguna Anda, lalu pilih Kelola EC2 Instance. Kemudian lanjutkan ke langkah 3 dalam prosedur ini.
2. Jika memilih Kelola EC2 Instans atau langkah lain dalam prosedur ini menampilkan kesalahan, masuk ke EC2 konsol Amazon menggunakan kredensil untuk administrator di konsol Anda. Akun AWS Kemudian, lengkapi instruksi berikut. Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

   a. Masuk ke AWS Management Console at https://console.aws.amazon.com/jika Anda belum masuk.
   b. Buka EC2 konsol Amazon. Untuk melakukan ini, di bilah navigasi, pilih Layanan. Kemudian, pilih EC2.
   c. Di bilah navigasi, pilih Wilayah AWS tempat lingkungan Anda berada.
   d. Jika EC2 Dasbor ditampilkan, pilih Menjalankan Instans. Jika tidak, di panel navigasi layanan, perluas Instans jika belum diperluas dan pilih Instans.

e. Dalam daftar instance, pilih instance dengan ID Instance yang cocok dengan ID instans yang Anda catat sebelumnya.

3. Di tab Deskripsi untuk contoh, pilih tautan grup keamanan yang berada di sebelah Grup keamanan.
4. Dengan grup keamanan yang ditampilkan, lihat tab Masuk. Jika ada aturan dengan Jenis diatur ke Aturan TCP Kustom dan Rentang Port diatur ke 8080, 8081, atau 8082, pilih Batal, dan lewati ke depan. Langkah 3: Siapkan subnet untuk instans Jika tidak, pilih Edit.
5. Di kotak dialog Edit aturan masuk, pilih Tambahkan Aturan.
6. Untuk Jenis, pilih Aturan TCP Kustom.
7. Untuk Port Range, masukkan`8080`,`8081`, atau`8082`.
8. Untuk Sumber, pilih Di mana saja.

> **Note**
>
> Dengan memilih Anywhere for Source, Anda mengizinkan permintaan masuk dari alamat IP apa pun. Untuk membatasi ini ke alamat IP tertentu, pilih Custom dan kemudian masukkan rentang alamat IP. Atau, pilih IP Saya untuk membatasi permintaan hanya dari alamat IP Anda.

9. Pilih Simpan.

## Langkah 3: Siapkan subnet untuk instans

Gunakan konsol Amazon EC2 dan Amazon VPC untuk menyiapkan subnet untuk EC2 instans Amazon yang terhubung ke lingkungan. Kemudian, izinkan permintaan HTTP masuk melalui port 8080, 8081, atau 8082.

> **Note**
>
> Anda tidak diharuskan untuk menjalankan menggunakan HTTP melalui port`8080`,`8081`, atau`8082`. Namun, jika tidak, Anda tidak dapat melihat pratinjau aplikasi yang sedang berjalan dari dalam IDE. Untuk informasi selengkapnya, lihat Mempratinjau aplikasi yang sedang berjalan. Jika tidak, jika Anda menjalankan protokol atau port yang berbeda, gantilah di langkah ini.
>
> Langkah ini menjelaskan cara menyiapkan ACL jaringan untuk subnet di Amazon VPC yang dapat digunakan instans. Ini tidak diperlukan tetapi direkomendasikan. Menyiapkan jaringan

ACL menambahkan lapisan keamanan tambahan. Untuk informasi selengkapnya tentang jaringan ACLs, lihat berikut ini:

- Keamanan di Panduan Pengguna Amazon VPC
- Jaringan ACLs di Panduan Pengguna VPC Amazon

1. Di EC2 konsol Amazon, di panel navigasi layanan, perluas Instans jika belum diperluas, dan pilih Instans.
2. Dalam daftar instance, pilih instance dengan ID Instance yang cocok dengan ID instans yang Anda catat sebelumnya.
3. Di tab Deskripsi untuk instans, perhatikan nilai ID Subnet. ID subnet dalam format berikut:`subnet-1fab8aEX`.
4. Buka konsol Amazon VPC. Untuk melakukan ini, di bilah AWS navigasi, pilih Layanan dan kemudian pilih VPC.

   Untuk langkah ini, kami sarankan Anda masuk ke konsol VPC Amazon menggunakan kredensi administrator di konsol Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.
5. Jika Dasbor VPC ditampilkan, pilih Subnet. Jika tidak, di panel navigasi layanan, pilih Subnet.
6. Dalam daftar subnet, pilih subnet dengan nilai Subnet ID yang cocok dengan yang Anda catat sebelumnya.
7. Pada tab Ringkasan, pilih tautan ACL jaringan yang berada di sebelah Network ACL.
8. Dalam daftar jaringan ACLs, pilih jaringan ACL. (Hanya ada satu ACL jaringan.)
9. Lihat pada tab Aturan Masuk untuk ACL jaringan. Jika aturan sudah ada di mana Jenis diatur ke HTTP* (8080), HTTP* (8081), atau HTTP* (8082), lanjutkan ke Langkah 4: Bagikan URL aplikasi yang sedang berjalan. Jika tidak, pilih Edit.
10. Pilih Tambahkan aturan lain.
11. Untuk Aturan #, masukkan nomor untuk aturan (misalnya,`200`).
12. Untuk Jenis, pilih Aturan TCP Kustom.
13. Untuk Rentang Port, ketikkan `8080`, `8081`, atau `8082`.
14. Untuk Sumber, ketikkan rentang alamat IP untuk mengizinkan permintaan masuk. Misalnya, untuk mengizinkan permintaan masuk dari alamat IP apa pun, masukkan`0.0.0.0/0`.
15. Dengan Izinkan / Tolak diatur ke IZINKAN, pilih Simpan.

## Langkah 4: Bagikan URL aplikasi yang sedang berjalan

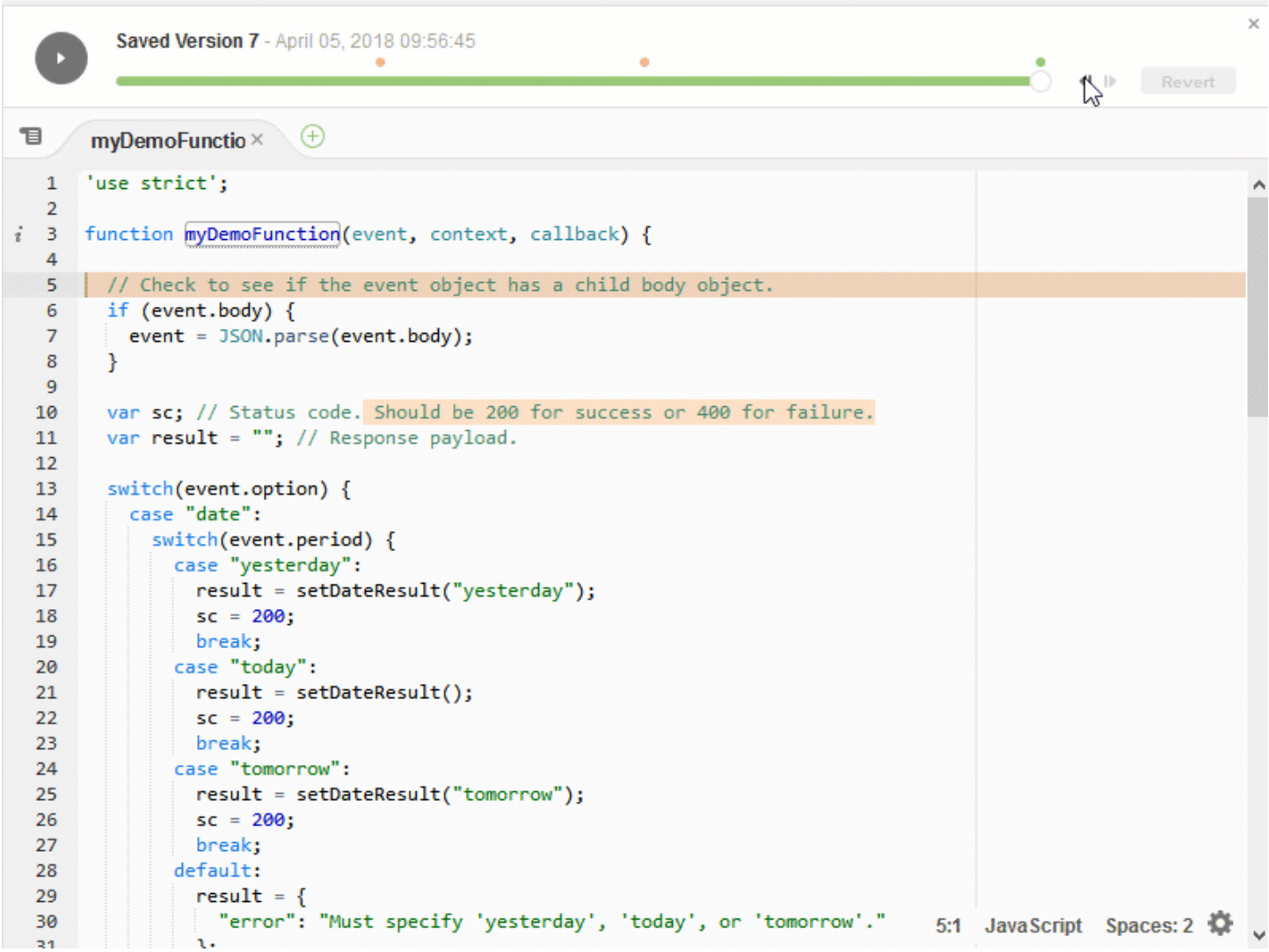
Setelah aplikasi Anda berjalan, Anda dapat berbagi aplikasi Anda dengan orang lain dengan memberikan URL aplikasi Anda. Untuk ini, Anda memerlukan alamat IP publik yang Anda catat sebelumnya. Untuk menulis URL lengkap aplikasi Anda, pastikan untuk memulai alamat IP publik aplikasi Anda dengan protokol yang benar. Selanjutnya, jika port aplikasi Anda bukan port default untuk protokol yang digunakannya, tambahkan informasi nomor port. Berikut ini adalah contoh URL aplikasi: `http://192.0.2.0:8080/index.html` menggunakan HTTP melalui port 8080.

Jika tab browser web yang dihasilkan menampilkan kesalahan, atau tab kosong, ikuti langkah-langkah pemecahan masalah. Tidak dapat menampilkan aplikasi yang sedang berjalan di luar IDE

> **Note**
>
> Alamat IP publik aplikasi Anda dapat berubah kapan saja instans untuk aplikasi Anda dimulai ulang. Untuk mencegah alamat IP Anda berubah, alokasikan alamat IP Elastis, lalu tetapkan alamat itu ke instance yang sedang berjalan. Untuk petunjuk, lihat Mengalokasikan Alamat IP Elastis dan Mengaitkan Alamat IP Elastis dengan Instans Berjalan di Panduan Pengguna Amazon EC2 . Mengalokasikan alamat IP Elastis dapat Akun AWS menyebabkan Anda dikenakan biaya. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.
> Anda tidak diharuskan untuk menjalankan aplikasi Anda menggunakan HTTP melalui port`8080`,`8081`, atau`8082`. Namun, jika tidak, Anda tidak dapat melihat pratinjau aplikasi yang sedang berjalan dari dalam IDE. Untuk informasi selengkapnya, lihat Mempratinjau aplikasi yang sedang berjalan.
> Misalkan, misalnya, permintaan yang berasal dari VPN yang memblokir lalu lintas melalui protokol atau port yang diminta. Kemudian, permintaan untuk mengakses URL aplikasi Anda mungkin gagal. Permintaan harus dibuat dari jaringan yang berbeda yang memungkinkan lalu lintas melalui protokol dan port yang diminta. Untuk informasi selengkapnya, hubungi administrator jaringan Anda.
> Kami tidak menyarankan berbagi URL di tab pratinjau aplikasi Anda di IDE dengan orang lain. (URL ini dalam format berikut:`https://12a34567b8cd9012345ef67abcd890e1.vfs.cloud9.us-east-2.amazonaws.com/`. Dalam format ini, `12a34567b8cd9012345ef67abcd890e1` adalah ID yang AWS Cloud9 diberikan ke lingkungan. `us-east-2`adalah ID Wilayah AWS untuk lingkungan.) URL ini bekerja hanya ketika IDE untuk lingkungan terbuka dan aplikasi berjalan di peramban web yang sama.

# Bekerja dengan Revisi File di IDE AWS Cloud9

Anda dapat menggunakan panel Riwayat Revisi File di Lingkungan Pengembangan AWS Cloud9 Terpadu (IDE) untuk melihat dan mengelola perubahan pada file di lingkungan AWS Cloud9 EC2 pengembangan. Panel Riwayat Revisi File tidak tersedia untuk file di lingkungan pengembangan AWS Cloud9 SSH.

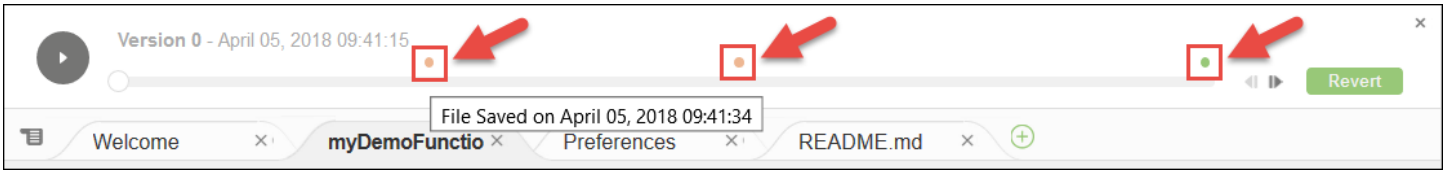



Untuk menampilkan panel Riwayat Revisi File untuk file, buka file di editor. Kemudian, pada bilah menu, pilih File, Tampilkan Riwayat Revisi File.

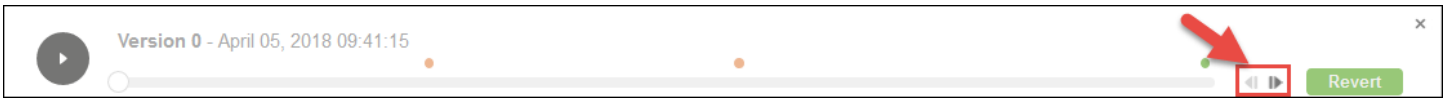
Panel Riwayat Revisi File mulai melacak riwayat revisi file di IDE setelah Anda pertama kali membuka file di editor di lingkungan, dan hanya untuk lingkungan tersebut. Panel Riwayat Revisi File melacak revisi file hanya dari editor itu sendiri. Ini tidak melacak revisi file yang dibuat dengan cara lain (misalnya oleh terminal, Git, atau alat revisi file lainnya).

Anda tidak dapat mengedit file saat panel Riwayat Revisi File ditampilkan. Untuk menyembunyikan panel, pilih File, Tampilkan Riwayat Revisi lagi, atau pilih X (Tutup penggeser waktu) di sudut panel.

Untuk melompat ke versi file yang terkait dengan tindakan penyimpanan file, pilih titik File Disimpan pada di atas penggeser revisi.




Untuk maju atau mundur satu versi dari versi file yang saat ini dipilih pada penggeser revisi, pilih salah satu panah langkah (Langkah revisi maju atau Langkah revisi mundur).




Untuk maju secara otomatis satu versi file pada satu waktu dari awal hingga akhir riwayat revisi, pilih tombol putar (Putar ulang riwayat file).

Untuk membuat versi file yang saat ini dipilih menjadi versi terbaru dalam riwayat revisi, pilih Kembalikan.

## Bekerja dengan File Gambar di AWS Cloud9 IDE

Anda dapat menggunakan AWS Cloud9 Integrated Development Environment (IDE) untuk melihat dan mengedit file gambar.

- Melihat atau Mengedit Gambar
- Mengubah ukuran Gambar
- Pangkas Gambar
- Memutar Gambar
- Balikkan Gambar
- Memperbesar Gambar
- Menghaluskan Gambar

## Melihat atau Mengedit Gambar

Di AWS Cloud9 IDE, buka file untuk gambar yang ingin Anda lihat atau edit. Jenis file gambar yang didukung meliputi:

- `.bmp`
- `.gif` (hanya lihat)
- `.ico` (hanya lihat)
- `.jpeg`
- `.jpg`
- `.png`
- `.tiff`

## Mengubah Ukuran Gambar

1. Buka file gambar di IDE.
2. Pada bilah pengeditan gambar, pilih Ubah Ukuran.
3. Untuk mengubah lebar gambar, ketikkan Lebar baru dalam piksel. Atau pilih "-" atau "+" di sebelah Lebar untuk mengubah lebar saat ini satu piksel pada satu waktu.
4. Untuk mengubah tinggi gambar, ketikkan Tinggi baru dalam piksel. Atau pilih "-" atau "+" di sebelah Tinggi untuk mengubah tinggi saat ini satu piksel pada satu waktu.
5. Untuk mempertahankan rasio lebar dan tinggi gambar, biarkan Pertahankan Rasio Aspek tercentang.
6. Untuk mengonfirmasi ukuran baru gambar, pada bilah pengeditan gambar, lihat pengukuran lebar (W) dan tinggi (H) dalam piksel.
7. Pilih Ubah Ukuran.
8. Untuk membuang pengubahan ukuran, pada bilah menu, pilih Edit, Urungkan. Untuk mempertahankan ukuran baru, pilih File, Simpan.

## Memangkas Gambar

1. Buka file gambar di IDE.
2. Seret penunjuk ke bagian gambar yang ingin Anda simpan.

3. Untuk mengonfirmasi dimensi seleksi, pada bilah pengeditan gambar, lihat dimensi Seleksi, sebagai berikut:
   - Jarak dalam piksel dari tepi kiri gambar asli ke tepi kiri seleksi (L)
   - Jarak dalam piksel dari tepi atas gambar asli ke tepi atas seleksi (T)
   - Lebar seleksi dalam piksel (W)
   - Tinggi seleksi dalam piksel (H)
4. Pada bilah pengeditan gambar, pilih Pangkas.
5. Untuk membuang pemangkasan, pada bilah menu, pilih Edit, Urungkan. Untuk mempertahankan gambar yang baru dipangkas, pilih File, Simpan.

## Memutar Gambar

1. Buka file gambar di IDE.
2. Untuk memutar gambar berlawanan arah jarum jam, pada bilah pengeditan gambar, pilih Putar 90 Derajat ke Kiri.
3. Untuk memutar gambar searah jarum jam, pada bilah pengeditan gambar, pilih Putar 90 Derajat ke Kanan.
4. Untuk membuang pemutaran, pada bilah menu, pilih Edit, Urungkan. Untuk mempertahankan gambar yang baru diputar, pilih File, Simpan.

## Membalik Gambar

1. Buka file gambar di IDE.
2. Untuk membalik gambar secara horizontal, pada bilah pengeditan gambar, pilih FlipH.
3. Untuk membalik gambar secara vertikal, pada bilah pengeditan gambar, pilih FlipV.
4. Untuk membuang pembalikan, pada bilah menu, pilih Edit, Urungkan. Untuk mempertahankan gambar yang baru dibalik, pilih File, Simpan.

## Memperbesar Gambar

1. Buka file gambar di IDE.
2. Pada bilah pengeditan gambar, pilih salah satu faktor zoom yang tersedia (misalnya, 75%, 100%, atau 200%).

## Menghaluskan Gambar

1. Buka file gambar di IDE.
2. Pada bilah pengeditan gambar, pilih Haluskan untuk mengurangi jumlah pikselasi pada gambar. Untuk membuang penghalusan, batalkan pilihan Haluskan.
3. Pada bilah menu, pilih File, Simpan.

# Bekerja dengan Pembangun, Pelari, dan Debugger di IDE AWS Cloud9

Seorang builder menginstruksikan AWS Cloud9 Integrated Development Environment (IDE) bagaimana membangun file proyek. Seorang pelari menginstruksikan AWS Cloud9 IDE cara menjalankan file dari jenis tertentu. Sebuah runner dapat menggunakan debugger untuk membantu menemukan masalah apa pun dalam kode sumber file.

Anda dapat menggunakan AWS Cloud9 IDE untuk membangun, menjalankan, dan men-debug kode Anda dengan cara berikut:

- Menggunakan builder untuk membangun file proyek Anda. Untuk informasi selengkapnya, lihat Membangun File Proyek Anda.
- Gunakan runner untuk menjalankan (serta opsional, untuk men-debug) kode Anda. Untuk informasi selengkapnya, lihat Build, Run, dan Debug Support bawaan dan Jalankan Kode Anda.
- Ubah runner bawaan untuk menjalankan (serta opsional, untuk men-debug) kode Anda dengan cara yang berbeda dari bagaimana itu awalnya didefinisikan. Untuk informasi selengkapnya, lihat Mengubah Pelari Bawaan.
- Gunakan runner untuk menjalankan (serta opsional, untuk men-debug) kode Anda dengan kombinasi kustom dari nama file, opsi baris perintah, mode debug, direktori kerja saat ini, dan variabel lingkungan. Untuk informasi selengkapnya, lihat Membuat Konfigurasi Jalankan.
- Buat builder atau runner Anda sendiri. Untuk informasi selengkapnya, lihat Membuat Builder atau Runner.

## Dukungan Bangun, Jalankan, dan Debug Bawaan

AWS Cloud9 IDE menyediakan dukungan bawaan untuk membangun, menjalankan, dan men-debug kode untuk beberapa bahasa. Untuk daftar lengkap, lihat Dukungan Bahasa.

Dukungan bangun bawaan tersedia pada bilah menu dengan perintah menu Jalankan, Bangun Sistem dan Jalankan, Bangun. Untuk menambahkan dukungan untuk bahasa atau alat pemrograman yang tidak tercantum, lihat Buat Builder atau Runner.

Dukungan eksekusi bawaan tersedia dengan tombol Jalankan, dan pada bilah menu dengan perintah menu Jalankan, Jalankan Dengan dan Jalankan, Konfigurasi Penjalanan. Untuk menambahkan dukungan untuk bahasa atau alat pemrograman yang tidak tercantum, lihat Buat Builder atau Runner dan Buat Konfigurasi Penjalanan.

Dukungan debug bawaan tersedia melalui jendela Debugger. Untuk menampilkan jendela Debugger, pilih tombol Debugger. Jika tombol Debugger tidak terlihat, pilih Jendela, Debugger pada bilah menu.

## Bangun File Proyek Anda

1. Buka file yang sesuai dengan kode yang ingin Anda bangun.
2. Pada bilah menu, pilih Jalankan, Bangun Sistem, lalu pilih nama builder yang akan digunakan, jika belum dipilih. Jika builder yang ingin Anda gunakan tidak terdaftar, hentikan prosedur ini, selesaikan langkah-langkah di Buat Builder atau Runner, dan kemudian kembali ke prosedur ini.
3. Pilih Jalankan, Bangun.

## Jalankan Kode Anda

1. Buka file yang sesuai dengan kode yang ingin Anda jalankan, jika file tersebut belum dibuka dan dipilih.
2. Pada bilah menu, pilih salah satu dari berikut ini:
   - Untuk menjalankan kode dengan runner bawaan yang paling cocok, pilih Jalankan, Jalankan. Jika AWS Cloud9 tidak dapat menemukannya, perintah ini dinonaktifkan.
   - Untuk menjalankan kode dengan konfigurasi run yang AWS Cloud9 terakhir digunakan, pilih Run, Run Last.
   - Untuk menjalankan kode dengan runner tertentu, pilih Jalankan, Jalankan Dengan, dan kemudian pilih nama runner. Jika runner yang ingin Anda gunakan tidak terdaftar, hentikan prosedur ini, selesaikan langkah-langkah di Buat Builder atau Runner, dan kemudian kembali ke prosedur ini.
   - Untuk menjalankan kode dengan runner tertentu dengan kombinasi nama file kustom, opsi baris perintah, mode debug, direktori kerja saat ini, dan variabel lingkungan, pilih Jalankan, Konfigurasi Penjalanan, dan kemudian pilih nama konfigurasi penjalanan. Pada tab konfigurasi

penjalanan yang ditampilkan, pilih Runner: Otomatis, pilih runner yang ingin Anda gunakan, lalu pilih Jalankan. Jika runner yang ingin Anda gunakan tidak terdaftar, hentikan prosedur ini, selesaikan langkah-langkah di Buat Builder atau Runner, dan kemudian kembali ke prosedur ini.

## Debug Kode Anda

1. Pada tab konfigurasi penjalanan untuk kode Anda, pilih Jalankan dalam Mode Debug. Ikon bug berubah menjadi hijau dengan latar belakang putih. Untuk informasi selengkapnya, lihat Jalankan Kode Anda dan Buat Konfigurasi Penjalanan.
2. Atur breakpoint dalam kode Anda yang ingin Anda jeda selama penjalanan, sebagai berikut:
   a. Buka setiap file yang ingin Anda atur breakpoint-nya.
   b. Pada setiap titik dalam file di mana Anda ingin mengatur breakpoint, pilih area kosong di saluran di sebelah kiri nomor baris. Sebuah lingkaran merah muncul.

      Untuk menghapus breakpoint, pilih breakpoint yang ada di saluran.

      Untuk menonaktifkan breakpoint alih-alih menghapusnya, di jendela Debugger, di Breakpoints, kosongkan kotak yang sesuai dengan breakpoint yang ingin Anda nonaktifkan. Untuk mengaktifkan breakpoint lagi, pilih kotak yang Anda kosongkan.

      Untuk menonaktifkan semua breakpoint sekaligus, di jendela Debugger, pilih Nonaktifkan Semua Breakpoint. Untuk mengaktifkan semua breakpoint lagi, pilih Aktifkan Semua Breakpoint.

      Jika jendela Debugger tidak terlihat, pilih tombol Debugger. Jika tombol Debugger tidak terlihat, pada bilah menu pilih Jendela, Debugger.
3. Atur ekspresi tontonan apa pun yang ingin Anda dapatkan nilainya pada titik di mana penjalanan dijeda, sebagai berikut:
   a. Di jendela Debugger, di Lihat Ekspresi, pilih Ketik ekspresi di sini.
   b. Ketik ekspresi yang ingin Anda tonton, lalu tekan `Enter`.

      Untuk mengubah ekspresi tontonan yang ada, klik kanan ekspresi tontonan, lalu pilih Edit Ekspresi Tonton. Ketikkan perubahan, lalu tekan `Enter`.

      Untuk menghapus ekspresi tontonan yang ada, klik kanan ekspresi tontonan, lalu pilih Hapus Ekspresi Tonton.
4. Eksekusi kode Anda seperti yang dijelaskan di Jalankan Kode Anda.

Setiap kali penjalanan dijeda, Anda juga dapat menjeda penunjuk pada bagian kode apa pun yang ditampilkan (misalnya, variabel) untuk menampilkan informasi apa pun yang tersedia tentangnya di tooltip.

## Ubah Runner Bawaan

Langkah ini menunjukkan bagaimana Anda dapat mengubah Built-in Runner untuk menjalankan (dan secara opsional, untuk men-debug) kode Anda dengan cara yang berbeda dari cara awalnya didefinisikan.

1. Pada bilah menu, pilih Jalankan, Jalankan Dengan, lalu pilih runner bawaan yang ingin Anda ubah.
2. Hentikan runner dari mencoba menjalankan kode Anda dengan memilih, Hentikan pada tab konfigurasi penjalanan yang ditampilkan.
3. Pilih Runner: Runner saya, di mana Runner Saya adalah nama runner yang ingin Anda ubah, lalu pilih Edit Runner.
4. Pada tab Runner Saya.run yang ditampilkan, ubah ketentuan runner saat ini. Lihat Tentukan Builder atau Runner.
5. Pilih File, Simpan Sebagai. Simpan file dengan nama yang sama (My Runner.run) di `my-environment/.c9/runners` direktori, di `my-environment` mana nama lingkungan pengembangan Anda AWS Cloud9 .

> **Note**
>
> Perubahan apa pun yang Anda buat pada runner internal hanya berlaku untuk lingkungan tempat Anda membuat perubahan tersebut. Untuk menerapkan perubahan ke lingkungan terpisah, buka lingkungan lain, lalu ikuti langkah sebelumnya untuk membuka, mengedit, dan menyimpan perubahan yang sama tersebut pada runner bawaan itu.

## Buat Konfigurasi Penjalanan

Langkah ini menunjukkan bagaimana Anda dapat menjalankan runner (dan secara opsional, untuk men-debug) kode Anda dengan kombinasi kustom nama file, opsi baris perintah, mode debug, direktori kerja saat ini, dan variabel lingkungan.

Pada bilah menu, pilih Jalankan, Konfigurasi Penjalanan, Konfigurasi Penjalanan Baru. Pada tab konfigurasi penjalanan yang ditampilkan, lakukan hal berikut:

1. Dalam kotak di sebelah Jalankan dan Mulai Ulang, ketikkan nama yang akan ditampilkan di menu Jalankan, Konfigurasi Penjalanan untuk konfigurasi penjalanan ini.
2. Di kotak Perintah, ketikkan opsi baris perintah kustom yang ingin Anda gunakan.
3. Agar konfigurasi penjalanan ini menggunakan pengaturan debug runner yang telah ditentukan, pilih Jalankan dalam Mode Debug. Ikon bug akan berubah menjadi hijau dengan latar belakang putih.
4. Agar konfigurasi penjalanan ini menggunakan direktori kerja tertentu, pilih CWD, pilih direktori yang akan digunakan, lalu pilih Pilih.
5. Agar konfigurasi penjalanan ini menggunakan variabel lingkungan tertentu, pilih ENV, lalu ketikkan nama dan nilai setiap variabel lingkungan.

Untuk menggunakan konfigurasi penjalanan ini, buka file yang sesuai dengan kode yang ingin Anda jalankan. Pilih Jalankan, Konfigurasi Penjalanan pada bilah menu, dan kemudian pilih nama konfigurasi penjalanan ini. Pada tab konfigurasi penjalanan yang ditampilkan, pilih Runner: Otomatis, pilih runner yang ingin Anda gunakan, lalu pilih Jalankan.

> **Note**
>
> Konfigurasi penjalanan apa pun yang Anda buat hanya berlaku untuk lingkungan tempat Anda membuat konfigurasi penjalanan. Untuk menambahkan konfigurasi penjalanan tersebut ke lingkungan terpisah, buka lingkungan lain, lalu ikuti langkah sebelumnya untuk membuat konfigurasi penjalanan yang sama di lingkungan tersebut.

## Buat Builder atau Runner

Langkah ini menunjukkan bagaimana Anda dapat membuat pembangun atau pelari Anda sendiri.

1. Untuk membuat builder, pada bilah menu, pilih Jalankan, Bangun Sistem, Bangun Sistem Baru. Untuk membuat runner, pada bilah menu, pilih Jalankan, Jalankan Dengan, Runner Baru.
2. Pada tab builder (berlabel Builder Saya.build) atau tab runner (berlabel Runner Saya.run) yang ditampilkan, tentukan builder atau runner. Lihat Tentukan Builder atau Runner.
3. Setelah Anda menentukan builder atau runner, pilih File, Simpan Sebagai. Untuk builder, simpan file dengan ekstensi `.build` dalam direktori `my-environment/.c9/builders`, di mana `my-environment` adalah nama lingkungan Anda. Untuk runner, simpan file dengan ekstensi file `.run` dalam direktori `my-environment/.c9/runners`, di mana `my-environment` adalah

nama lingkungan Anda. Nama file yang Anda tentukan akan menjadi nama yang ditampilkan pada menu Jalankan, Bangun Sistem (untuk builder) atau menu Jalankan, Jalankan Dengan (untuk runner). Oleh karena itu, kecuali jika Anda menentukan nama file yang berbeda, secara default nama tampilan akan berupa Builder Saya (untuk builder) atau Runner Saya(untuk runner).

Untuk menggunakan builder atau runner ini, lihat Bangun File Proyek Anda atau Jalankan Kode Anda.

> **Note**
>
> Setiap builder atau runner yang Anda buat hanya berlaku untuk lingkungan tempat Anda membuat builder atau runner tersebut. Untuk menambahkan builder atau runner penjalanan tersebut ke lingkungan terpisah, buka lingkungan lain, lalu ikuti langkah sebelumnya untuk membuat builder atau runner yang sama di lingkungan tersebut.

## Tentukan Builder atau Runner

Topik ini menunjukkan bagaimana Anda dapat mendefinisikan pembangun atau pelari. Sebelum Anda menentukan pembangun atau pelari, pastikan Anda telah membuat pembangun atau pelari.

Pada tab builder atau runner yang ditampilkan, gunakan JSON untuk menentukan runner atau builder. Mulailah dengan kode berikut sebagai templat.

Untuk builder, mulailah dengan kode ini.

```
{
  "cmd": [],
  "info": "",
  "env": {},
  "selector": ""
}
```

Untuk runner, mulailah dengan kode ini.

```
{
  "cmd": [],
  "script": "",
```

```
  "working_dir": "",
  "info": "",
  "env": {},
  "selector": "",
  "debugger": "",
  "debugport": ""
}
```

Dalam kode sebelumnya:

- `cmd`: Merupakan daftar string yang dipisahkan koma AWS Cloud9 untuk dijalankan sebagai satu perintah.

  Saat AWS Cloud9 menjalankan perintah ini, setiap string dalam daftar akan dipisahkan oleh satu spasi. Misalnya, AWS Cloud9 akan berjalan `"cmd": [ "ls", "$file", "$args"]` sebagai`ls $file $args`, di mana AWS Cloud9 akan mengganti `$file` dengan path lengkap ke file saat ini dan `$args` dengan argumen apa pun yang dimasukkan setelah nama file. Untuk informasi selengkapnya, lihat daftar variabel yang didukung nanti di bagian ini.
- `script`: Mewakili skrip bash (yang juga dapat ditentukan sebagai larik baris yang diperlukan agar mudah dibaca) yang dijalankan runner di terminal.
- `working_dir`: Mewakili direktori tempat runner akan dijalankan.
- `info`: Mewakili string teks apa pun yang ingin Anda tampilkan kepada pengguna di awal menjalankan. String ini dapat berisi variabel, misalnya`Running $project_path $file_name...`, di mana AWS Cloud9 akan menggantikan `$project_path` dengan path direktori dari file saat ini dan `$file_name` dengan bagian nama dari file saat ini. Lihat daftar variabel yang didukung nanti di bagian ini.
- `env`: Merupakan array argumen baris perintah AWS Cloud9 untuk digunakan, misalnya:

  ```
  "env": {
    "LANG": "en_US.UTF-8",
    "SHLVL": "1"
  }
  ```
- `selector`: Merupakan ekspresi reguler apa pun yang AWS Cloud9 ingin Anda gunakan untuk mengidentifikasi nama file yang berlaku untuk pelari ini. Misalnya, Anda dapat menentukan `source.py` untuk file Python.
- `debugger`: Merupakan nama debugger yang tersedia yang AWS Cloud9 ingin Anda gunakan yang kompatibel dengan runner ini. Misalnya, Anda dapat menentukan `v8` untuk debugger V8.

- `debugport`: Merupakan nomor port yang AWS Cloud9 ingin Anda gunakan selama debugging. Misalnya, Anda dapat menentukan `15454` untuk nomor port yang akan digunakan.

Tabel berikut menunjukkan variabel yang dapat Anda gunakan.

| Variabel | Deskripsi |
| --- | --- |
| `$file_path` | Direktori dari file saat ini, misalnya, `/home/ec2-user/environment` atau `/home/ubuntu/environment` . |
| `$file` | Jalur lengkap ke file saat ini, misalnya, `/home/ec2-user/environment/hello.py` atau `/home/ubuntu/environment/hello.py` . |
| `$args` | Argumen apa pun yang dimasukkan setelah nama file, misalnya, `"5" "9"`. |
| `$file_name` | Bagian nama file saat ini, misalnya, `hello.py`. |
| `$file_extension` | Ekstensi file saat ini, misalnya, `py`. |
| `$file_base_name` | Nama file saat ini tanpa ekstensi file, misalnya, `hello`. |
| `$packages` | Jalur lengkap ke folder paket. |
| `$project` | Jalur lengkap ke folder proyek saat ini. |
| `$project_path` | Direktori file proyek saat ini, misalnya, `/home/ec2-user/environment/` atau `/home/ubuntu/environment/` . |
| `$project_name` | Nama file proyek saat ini tanpa ekstensi file, misalnya, `my-demo-environment` . |
| `$project_extension` | Ekstensi file proyek saat ini. |

| Variabel | Deskripsi |
| --- | --- |
| `$project_base_name` | Nama file proyek saat ini tanpa ekstensi. |
| `$hostname` | Nama host lingkungan, misalnya, `192.0.2.0` . |
| `$hostname_path` | Nama host lingkungan dengan jalur relatif ke file proyek, misalnya, `https://192.0.2.0/hello.js` . |
| `$url` | URL lengkap untuk mengakses lingkungan, misalnya, `https://192.0.2.0.` . |
| `$port` | Port yang ditetapkan ke lingkungan, misalnya, `8080`. |
| `$ip` | Alamat IP untuk menjalankan proses terhadap lingkungan, misalnya, `0.0.0.0`. |

Sebagai contoh, file builder berikut bernama `G++.build` mendefinisikan builder untuk GCC yang menjalankan perintah **`g++`** dengan opsi `-o` untuk mengompilasi file saat ini (misalnya, `hello.cpp`) ke dalam modul objek. Kemudian menautkan modul objek ke dalam program dengan nama yang sama dengan file saat ini (misalnya, `hello`). Berikut perintah setara adalah `g++ -o hello hello.cpp`.

```
{
  "cmd": [ "g++", "-o", "$file_base_name", "$file_name" ],
  "info": "Compiling $file_name and linking to $file_base_name...",
  "selector": "source.cpp"
}
```

Sebagai contoh lain, file runner berikut bernama `Python.run` mendefinisikan runner yang menggunakan Python untuk menjalankan file saat ini dengan argumen apa pun yang disediakan. Misalnya, jika file saat ini diberi nama `hello.py` serta argumen `5` dan `9` telah disediakan, perintah yang setara adalah `python hello.py 5 9`.

```
{
  "cmd": [ "python", "$file_name", "$args" ],
```

```
  "info": "Running $file_name...",
  "selector": "source.py"
}
```

Akhirnya, file runner berikut bernama `Print Run Variables.run` mendefinisikan runner yang hanya mengeluarkan nilai dari setiap variabel yang tersedia dan kemudian berhenti.

```
{
  "info": "file_path = $file_path, file = $file, args = $args, file_name = $file_name,
 file_extension = $file_extension, file_base_name = $file_base_name, packages
 = $packages, project = $project, project_path = $project_path, project_name
 = $project_name, project_extension = $project_extension, project_base_name =
 $project_base_name, hostname = $hostname, hostname_path = $hostname_path, url = $url,
 port = $port, ip = $ip"
}
```

# Bekerja dengan Variabel Lingkungan Kustom di AWS Cloud9 IDE

AWS Cloud9 Integrated Development Environment (IDE) mendukung pengaturan variabel lingkungan kustom. Anda dapat mengatur variabel lingkungan kustom di AWS Cloud9 IDE dengan cara berikut.

- Tetapkan Variabel Lingkungan Kustom Tingkat Perintah
- Tetapkan Variabel Lingkungan Pengguna Kustom di ~/.bash_profile
- Tetapkan Variabel Lingkungan Kustom Lokal
- Tetapkan Variabel Lingkungan Pengguna Kustom di ~/.bashrc
- Mengatur Variabel Lingkungan Kustom dalam Daftar ENV

## Atur Variabel Lingkungan Kustom Tingkat Perintah

Anda dapat mengatur variabel lingkungan kustom tingkat perintah saat Anda menjalankan perintah di lingkungan AWS Cloud9 pengembangan Anda. Untuk menguji perilaku ini, buat file bernama `script.sh` dengan kode berikut:

```
#!/bin/bash

echo $MY_ENV_VAR
```

Jika Anda menjalankan perintah berikut, terminal menampilkan `Terminal session`:

```
MY_ENV_VAR='Terminal session' sh ./script.sh
```

Jika Anda mengatur variabel lingkungan kustom dengan menggunakan beberapa pendekatan yang dijelaskan dalam topik ini, maka ketika Anda mencoba untuk mendapatkan nilai variabel lingkungan kustom, pengaturan ini akan diprioritaskan di atas yang lainnya.

## Atur Variabel Lingkungan Pengguna Kustom di ~/.bash_profile

Anda dapat mengatur variabel lingkungan pengguna khusus dalam file `~/.bash_profile` di lingkungan Anda. Untuk menguji perilaku ini, tambahkan kode berikut ke file `~/.bash_profile` di lingkungan Anda:

```
export MY_ENV_VAR='.bash_profile file'
```

Jika Anda kemudian menjalankan `sh ./script.sh` dari baris perintah, terminal menampilkan `.bash_profile file`. (Ini mengasumsikan Anda membuat file `script.sh` seperti yang dijelaskan sebelumnya.)

## Atur Variabel Lingkungan Kustom Lokal

Anda dapat mengatur variabel lingkungan kustom lokal dalam sesi terminal dengan menjalankan perintah **`export`** . Untuk menguji perilaku ini, jalankan perintah berikut di sesi terminal:

```
export MY_ENV_VAR='Command line export'
```

Jika Anda kemudian menjalankan `sh ./script.sh` dari baris perintah, terminal menampilkan `Command line export`. (Ini mengasumsikan Anda membuat file `script.sh` seperti yang dijelaskan sebelumnya.)

Jika Anda mengatur variabel lingkungan kustom yang sama dengan perintah **`export`** dan dalam file `~/.bash_profile` Anda, maka ketika Anda mencoba untuk mendapatkan nilai variabel lingkungan kustom, pengaturan perintah **`export`** akan diprioritaskan.

## Atur Variabel Lingkungan Pengguna Kustom di ~/.bashrc

Anda dapat mengatur variabel lingkungan pengguna khusus dalam file `~/.bashrc` di lingkungan Anda. Untuk menguji perilaku ini, tambahkan kode berikut ke file `~/.bashrc` di lingkungan Anda:

```
export MY_ENV_VAR='.bashrc file'
```

Jika Anda kemudian menjalankan `sh ./script.sh` dari baris perintah, terminal menampilkan `.bashrc file`. (Ini mengasumsikan Anda membuat file `script.sh` seperti yang dijelaskan sebelumnya.)

Jika Anda mengatur variabel lingkungan kustom yang sama dengan perintah **`export`** dan dalam file `~/.bashrc` Anda, maka ketika Anda mencoba untuk mendapatkan nilai variabel lingkungan kustom, pengaturan perintah **`export`** akan diprioritaskan.

### Atur Variabel Lingkungan Kustom dalam Daftar ENV

Anda dapat mengatur variabel lingkungan kustom dalam daftar ENV pada tab Jalankan.

Untuk menguji perilaku ini, lakukan hal berikut:

1. Pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.
2. Di tab [Baru] - Diam, Pilih Runner: Otomatis, lalu pilih Shell script.
3. Pilih ENV, dan kemudian ketikkan `MY_ENV_VAR` untuk Nama dan `ENV list` untuk Nilai.
4. Untuk Perintah, ketikkan `./script.sh`.
5. Pilih tombol Jalankan. Tab runner menampilkan `ENV list`. (Ini mengasumsikan Anda membuat file `script.sh` seperti yang dijelaskan sebelumnya.)

Jika Anda mengatur variabel lingkungan kustom yang sama dalam file `~/.bash_profile` Anda, dengan perintah **`export`** , dalam file `~/.bashrc` Anda, dan dalam daftar ENV, maka ketika Anda mencoba untuk mendapatkan nilai variabel lingkungan kustom, pengaturan file `~/.bash_profile` mengambil prioritas pertama, diikuti dengan pengaturan perintah **`export`** , pengaturan file `~/.bashrc`, dan pengaturan daftar ENV.

> **Note**
>
> Daftar ENV adalah satu-satunya pendekatan untuk mendapatkan dan mengatur variabel lingkungan kustom dengan menggunakan kode, terpisah dari shell script.

## Bekerja dengan pengaturan proyek di AWS Cloud9 IDE

Pengaturan proyek, yang hanya berlaku untuk lingkungan AWS Cloud9 pengembangan saat ini, mencakup jenis pengaturan berikut:

- Pengaturan editor kode, seperti apakah akan menggunakan tab lunak dan akhir baris file baru
- Jenis file yang harus diabaikan
- Jenis petunjuk dan peringatan untuk menampilkan atau menekan
- Pengaturan kode dan pemformatan untuk bahasa pemrograman seperti JavaScript, PHP, Python, dan Go
- Jenis konfigurasi yang digunakan saat menjalankan dan membangun kode

Meskipun pengaturan proyek berlaku hanya untuk satu lingkungan, Anda dapat menerapkan pengaturan proyek untuk satu lingkungan pada lingkungan lain.

- Melihat atau mengubah pengaturan proyek
- Menerapkan pengaturan proyek saat ini untuk lingkungan ke lingkungan lain
- Sesuaikan setelan proyek Anda

## Melihat atau mengubah pengaturan proyek

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Agar dapat melihat pengaturan proyek untuk lingkungan saat ini, pada tab Preferensi, di panel navigasi samping, pilih Pengaturan proyek.
3. Untuk mengubah pengaturan proyek saat ini untuk lingkungan, ubah pengaturan yang Anda inginkan di panel Pengaturan Proyek.

Untuk informasi selengkapnya tentang cara membuat perubahan dalam setelan project, lihat Menyesuaikan setelan project.

## Menerapkan pengaturan proyek saat ini untuk lingkungan ke lingkungan lain

1. Di lingkungan sumber dan target, pada bilah menu AWS Cloud9 IDE, pilih AWS Cloud9, Buka Pengaturan Proyek Anda.
2. Di lingkungan sumber, salin konten tab project.settings yang ditampilkan.
3. Di lingkungan target, timpa konten tab pengaturan.proyek dengan konten yang disalin dari lingkungan sumber.
4. Dalam lingkungan target, simpan tab pengaturan.proyek.

# Sesuaikan setelan proyek Anda

Bagian ini menjelaskan jenis pengaturan proyek yang dapat Anda ubah di panel Pengaturan Proyek tab Preferensi.

- EC2 contoh
- Editor kode (Ace)
- Temukan dalam file
- Petunjuk dan peringatan
- JavaScript dukungan
- Membangun
- Jalankan dan debug
- Jalankan konfigurasi
- Pemformat kode
- TypeScript dukungan
- PHP dukungan
- Python dukungan
- Go dukungan

## EC2 contoh

Hentikan lingkungan saya

Pilih kapan akan menghentikan EC2 instans Amazon lingkungan Anda secara otomatis (jika digunakan) setelah Anda menutup semua instance browser web yang terhubung ke IDE untuk lingkungan tersebut. Anda dapat memilih rentang periode waktu dari seminggu hingga 30 menit. Anda juga dapat memilih untuk tidak menghentikan EC2 instans Amazon secara otomatis setelah keluar dari AWS Cloud9 IDE.

Jika Anda ingin menghentikan instans bahkan lebih cepat dari 30 menit setelah selesai dengan IDE, Anda dapat menghentikannya secara manual menggunakan antarmuka konsol.

## Editor kode (Ace)

Tab lembut

Jika dipilih, masukkan jumlah spasi yang ditentukan, bukan karakter tab setiap kali Anda menekan `Tab`.

Deteksi otomatis ukuran tab saat dimuat

Jika dipilih, AWS Cloud9 coba tebak ukuran tab.

Akhiran baris file baru

Jenis akhiran baris yang digunakan untuk file baru.

Opsi yang valid meliputi yang berikut:

- Windows (CRLF) untuk mengakhiri baris dengan carriage return lalu umpan baris.
- Unix (LF) untuk mengakhiri baris hanya dengan umpan baris.

Saat menyimpan, strip spasi

Jika dipilih, AWS Cloud9 cobalah untuk menghapus apa yang dianggap sebagai spasi dan tab yang tidak perlu dari file setiap kali file itu disimpan.

## Temukan dalam file

Abaikan File-file ini

Saat menemukan dalam file, jenis file yang AWS Cloud9 diabaikan.

Jumlah maksimum file yang akan dicari (dalam 1000)

Saat menemukan dalam file, jumlah maksimum file, dalam kelipatan 1.000, yang AWS Cloud9 ditemukan dalam lingkup saat ini.

## Petunjuk dan peringatan

Tingkat Peringatan

Tingkat minimum pesan yang akan diaktifkan.

Nilai-nilai yang valid meliputi:

- Info untuk mengaktifkan informasi, peringatan, dan pesan kesalahan.
- Peringatan untuk mengaktifkan hanya pesan peringatan dan kesalahan.
- Kesalahan untuk mengaktifkan pesan kesalahan saja.

Tandai Titik Koma Opsional yang Hilang

Jika diaktifkan, AWS Cloud9 tandai dalam file setiap kali melihat titik koma yang dapat digunakan dalam kode, tetapi itu tidak digunakan.

Tandai Variabel yang Tidak Dideklarasikan

Jika diaktifkan, AWS Cloud9 tandai dalam file setiap kali melihat variabel yang tidak dideklarasikan dalam kode.

Tandai Argumen Fungsi yang Tidak Digunakan

Jika diaktifkan, AWS Cloud9 tandai dalam file setiap kali melihat argumen yang tidak digunakan dalam suatu fungsi.

Abaikan Pesan yang Cocokkan Regex

AWS Cloud9 tidak akan menampilkan pesan apa pun yang cocok dengan ekspresi reguler yang ditentukan. Untuk informasi selengkapnya, lihat Menulis pola ekspresi reguler di JavaScript Topik Ekspresi Reguler di Jaringan Pengembang Mozilla.

## JavaScript dukungan

Kustomisasi JavaScript peringatan dengan .eslintrc

Jika diaktifkan, AWS Cloud9 gunakan `.eslintrc` file untuk menentukan JavaScript peringatan untuk mengaktifkan atau menonaktifkan. Untuk informasi selengkapnya, lihat Format File Konfigurasi di ESLint situs web.

JavaScript penyelesaian kode perpustakaan

Bagian JavaScript pustaka yang AWS Cloud9 digunakan untuk mencoba menyarankan atau melakukan penyelesaian kode otomatis.

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 mencoba untuk memformat kode dalam JavaScript berkas setiap kali file tersebut disimpan.

Gunakan bawaan JSBeautify sebagai pemformat kode

Jika diaktifkan, AWS Cloud9 gunakan implementasi internalnya JSBeautify untuk mencoba meningkatkan keterbacaan kode dalam file.

Pemformat kode kustom

Perintah AWS Cloud9 untuk mencoba menjalankan saat memformat kode dalam JavaScript berkas.

## Membangun

Jalur pembangun di lingkungan

Jalur untuk setiap konfigurasi bangunan kustom.

## Jalankan dan debug

Jalur pelari di lingkungan

Jalur untuk setiap konfigurasi pelaksanaan kustom.

URL pratinjau

URL yang digunakan agar dapat melihat pratinjau aplikasi untuk lingkungan.

## Jalankan konfigurasi

Konfigurasi pelaksanaan kustom untuk lingkungan ini.

Hapus konfigurasi yang dipilih

Menghapus konfigurasi pelaksanaan yang dipilih.

Tambahkan konfigurasi baru

Menciptakan konfigurasi pelaksanaan baru.

Tetapkan sebagai default

Atur konfigurasi pelaksanaan yang dipilih sebagai konfigurasi pelaksanaan default.

## Pemformat kode

### JSBeautify pengaturan

Pengaturan untuk meningkatkan keterbacaan kode dalam file.

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 upaya untuk menerapkan JSBeautify pengaturan setiap kali file kode disimpan.

Gunakan JSBeautify untuk JavaScript

Jika diaktifkan, AWS Cloud9 upaya untuk menerapkan JSBeautify pengaturan kapan saja JavaScript file disimpan.

Pertahankan baris kosong

Jika diaktifkan, AWS Cloud9 tidak menghapus baris kosong dalam file kode.

Pertahankan lekukan array

Jika diaktifkan, AWS Cloud9 mempertahankan lekukan deklarasi elemen dalam array dalam file kode.

JSLint spasi putih yang ketat

Jika diaktifkan, AWS Cloud9 mencoba menerapkan aturan JSLint spasi putih dalam file kode. Untuk informasi selengkapnya, lihat “Spasi Putih” di JSLint Bantuan.

Kawat gigi

Menentukan penyelarasan tanda kurung besar dalam kode.

Nilai-nilai yang valid meliputi:

- Tanda kurung dengan pernyataan kontrol untuk memindahkan setiap tanda kurung awal dan akhir agar selaras dengan pernyataan kontrol terkait, sesuai kebutuhan.

  Misalnya, kode ini diformat seperti:

  ```
  for (var i = 0; i < 10; i++) { if (i == 5) { console.log("Halfway done.") }}
  ```

  Berubah menjadi kode ini saat file disimpan:

  ```
  for (var i = 0; i < 10; i++) {
  ```

```
   if (i == 5) {
      console.log("Halfway done.")
   }
}
```

- Tanda kurung pada garis sendiri untuk memindahkan setiap tanda kurung ke garisnya sendiri, sesuai kebutuhan.

  Misalnya, kode ini diformat seperti:

  ```
  for (var i = 0; i < 10; i++) { if (i == 5) { console.log("Halfway done.") }}
  ```

  Berubah menjadi kode ini saat file disimpan:

  ```
  for (var i = 0; i < 10; i++) {if (i == 5)
    {
       console.log("Halfway done.")
    }
    }
  ```

- Tanda kurung akhir pada garis sendiri untuk memindahkan setiap tanda kurung akhir ke garisnya sendiri, sesuai kebutuhan.

  Misalnya, kode ini diformat seperti:

  ```
  for (var i = 0; i < 10; i++) {
    if (i == 5) { console.log("Halfway done.") }
  }
  ```

  Berubah menjadi kode ini saat file disimpan:

  ```
  for (var i = 0; i < 10; i++) {
     if (i == 5) {
        console.log("Halfway done.")
     }
  }
  ```

Pertahankan blok sebaris

Jika diaktifkan, AWS Cloud9 tidak mencoba memindahkan tanda kurung awal dan akhir untuk blok sebaris ke garis terpisah, jika kurung kurung tersebut berada pada baris yang sama.

Ruang sebelum kondisional

Jika diaktifkan, AWS Cloud9 tambahkan spasi sebelum setiap deklarasi bersyarat, sesuai kebutuhan.

String yang tidak bisa melarikan diri

Jika diaktifkan, AWS Cloud9 konversi string yang lolos ke padanannya yang tidak terlolos. Misalnya, mengonversi `\n` ke karakter baris baru dan mengonversi `\r` ke karakter carriage return.

Inden HTML batin

Jika diaktifkan, AWS Cloud9 indentasi `<head>` dan `<body>` bagian dalam kode HTML.

## TypeScript dukungan

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 coba format TypeScript kode kapan saja TypeScript file disimpan.

Pemformat kode kustom

Jalur ke konfigurasi pemformatan kode khusus untuk TypeScript kode.

## PHP dukungan

Aktifkan PHP penyelesaian kode

Jika diaktifkan, AWS Cloud9 upaya untuk menyelesaikan PHP kode.

PHP penyelesaian termasuk jalur

Lokasi yang AWS Cloud9 digunakan untuk mencoba membantu menyelesaikan PHP kode. Misalnya, jika Anda memiliki custom PHP file yang ingin Anda gunakan AWS Cloud9 untuk penyelesaian, dan file-file itu ada di suatu tempat di `~/environment` direktori, tambahkan `~/environment` ke jalur ini.

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 coba format PHP kode kapan saja PHP file disimpan.

Pemformat kode kustom

Jalur ke konfigurasi pemformatan kode khusus untuk PHP kode.

## Python dukungan

Aktifkan Python penyelesaian kode

Jika diaktifkan, AWS Cloud9 upaya untuk menyelesaikan Python kode. Untuk mengatur jalur yang AWS Cloud9 akan digunakan untuk menyelesaikan Python kode, gunakan pengaturan PYTHONPATH.

Python versi

Menentukan versi Python untuk digunakan.

Pylint opsi baris perintah

Pilihan AWS Cloud9 untuk digunakan untuk Pylint dengan Python kode. Untuk informasi lebih lanjut, lihat Pylint Panduan Pengguna pada Pylint situs web.

PYTHONPATH

Jalan menuju Python perpustakaan dan paket AWS Cloud9 untuk digunakan. Misalnya, jika Anda memiliki custom Python perpustakaan dan paket di `~/environment` direktori, tambahkan `~/environment` ke jalur ini.

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 coba format Python kode kapan saja Python file disimpan.

Pemformat kode kustom

Jalur ke konfigurasi pemformatan kode khusus untuk Python kode.

## Go dukungan

Aktifkan Go penyelesaian kode

Jika diaktifkan, AWS Cloud9 upaya untuk menyelesaikan Go kode.

Format Kode di Simpan

Jika diaktifkan, AWS Cloud9 coba format Go kode kapan saja Go file disimpan.

Pemformat kode kustom

Jalur ke konfigurasi pemformatan kode khusus untuk Go kode.

## Menghentikan EC2 instans lingkungan Anda secara manual

Pengaturan EC2 Instans memungkinkan Anda untuk secara otomatis menghentikan EC2 instans Amazon lingkungan Anda secepat 30 menit setelah Anda menutup semua instance browser web yang terhubung ke IDE.

Anda juga dapat menghentikan instance secara manual segera menggunakan konsol.

Untuk menghentikan EC2 instance lingkungan secara manual, pilih langkah-langkah berikut:

1. Setelah Anda menutup semua instance browser web yang terhubung ke IDE, pilih Lingkungan Anda di AWS Cloud9 konsol.
2. Pilih tombol di kanan atas panel yang memperlihatkan detail lingkungan yang Anda gunakan, dan pilih Lihat detail.
3. Dalam detail Lingkungan, di bawah EC2Instance, pilih Go To Instance.
4. Di EC2 konsol Amazon, di bawah Status instans, pilih kotak centang untuk memilih instance lingkungan Anda. Status Instance mungkin menunjukkan bahwa instance masih berjalan.
5. Pilih Status instans dan pilih Hentikan instans.
6. Ketika diminta konfirmasi, pilih Berhenti. Ini dapat memerlukan waktu beberapa menit sampai instans berhenti.

# Bekerja dengan pengaturan pengguna di AWS Cloud9 IDE

Pengaturan pengguna adalah pengaturan yang berlaku di setiap lingkungan AWS Cloud9 pengembangan yang terkait dengan AWS Identity and Access Management (pengguna IAM) Anda. Mereka termasuk pengaturan berikut:

- Pengaturan antarmuka pengguna umum seperti mengaktifkan animasi dan menandai tab yang diubah
- Pengaturan navigasi sistem file
- Pengaturan pencarian dan pencarian file
- Skema warna untuk sesi terminal dan output
- Pengaturan editor kode tambahan, seperti ukuran font, pelipatan kode, pemilihan baris penuh, animasi gulir, dan ukuran font

Saat Anda mengubah pengaturan pengguna, AWS Cloud9 mendorong perubahan tersebut ke cloud dan mengaitkannya dengan pengguna IAM Anda. AWS Cloud9 juga terus memindai cloud untuk perubahan pengaturan pengguna yang terkait dengan pengguna IAM Anda, dan menerapkan pengaturan tersebut ke lingkungan Anda saat ini. Anda dapat menggunakan ini untuk mengalami tampilan dan nuansa yang sama, apa pun AWS Cloud9 lingkungan tempat Anda bekerja.

> **Note**
>
> Untuk menyimpan dan mengambil pengaturan IDE Anda, AWS Cloud9 gunakan internal APIs `GetUserSettings` dan`UpdateUserSettings`.

Anda dapat berbagi pengaturan pengguna dengan pengguna lain, sebagai berikut:

- Melihat atau mengubah setelan pengguna Anda
- Bagikan pengaturan pengguna Anda dengan pengguna lain
- Sesuaikan setelan pengguna Anda

## Melihat atau mengubah setelan pengguna

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Untuk melihat pengaturan pengguna Anda di setiap lingkungan Anda, pada tab Preferensi, di panel navigasi samping, pilih Pengaturan Pengguna.
3. Di panel Pengaturan Pengguna, ubah pengaturan pengguna Anda di setiap lingkungan Anda.
4. Untuk menerapkan perubahan Anda ke lingkungan lain, cukup buka lingkungan tersebut. Jika lingkungan tersebut sudah terbuka, segarkan tab peramban web untuk lingkungan tersebut.

Untuk informasi selengkapnya tentang cara membuat perubahan di setelan pengguna, lihat Menyesuaikan setelan pengguna.

## Bagikan pengaturan pengguna Anda dengan pengguna lain

1. Di lingkungan sumber dan target, pada bilah menu AWS Cloud9 IDE, pilih AWS Cloud9, Buka Pengaturan Pengguna Anda.
2. Di lingkungan sumber, salin konten tab user.settings yang ditampilkan.

3. Di lingkungan target, timpa konten tab user.settings dengan konten yang disalin dari lingkungan sumber.
4. Dalam lingkungan target, simpan tab pengaturan.pengguna.

## Sesuaikan setelan pengguna Anda

Bagian ini menjelaskan jenis pengaturan pengguna yang dapat Anda ubah di panel Pengaturan Pengguna pada tab Preferensi:

- Umum
- Antarmuka pengguna
- Kolaborasi
- Tree and Go Panel
- Temukan dalam file
- Data meta
- Pengamat
- Terminal
- Keluaran
- Editor kode (Ace)
- Masukan
- Petunjuk dan peringatan
- Jalankan dan debug
- Pratinjau
- Membangun

### Umum

Setel ulang ke Pengaturan Pabrik

> Jika Anda memilih tombol Reset ke Default, AWS Cloud9 atur ulang semua pengaturan pengguna Anda ke pengaturan pengguna AWS Cloud9 default. Untuk mengonfirmasi, pilih Atur ulang pengaturan.

> **Warning**
>
> Anda tidak dapat membatalkan tindakan ini.

Peringatkan Sebelum Keluar

Setiap kali Anda mencoba untuk menutup IDE, AWS Cloud9 meminta Anda untuk mengkonfirmasi bahwa Anda ingin keluar.

## Antarmuka pengguna

Aktifkan Animasi UI

AWS Cloud9 menggunakan animasi dalam IDE.

Gunakan Tanda Bintang (*) untuk Menandai Tab yang Diubah

AWS Cloud9 menambahkan tanda bintang (*) ke tab yang memiliki perubahan tetapi kontennya belum disimpan.

Tampilkan Judul Tab Aktif sebagai Judul Browser

AWS Cloud9 mengubah judul tab browser web terkait ke judul tab aktif (misalnya, Untitled1, hello.js, Terminal, Preferences).

Secara Otomatis Menutup Panel Kosong

Setiap kali Anda memuat ulang lingkungan, AWS Cloud9 secara otomatis menutup panel apa pun yang dianggap kosong.

Ikon File Lingkungan dan Gaya Seleksi

Ikon AWS Cloud9 digunakan untuk file lingkungan, dan perilaku pemilihan file AWS Cloud9 digunakan.

Nilai yang valid meliputi:

- Default - AWS Cloud9 menggunakan ikon default dan perilaku pemilihan file default.
- Alternatif — AWS Cloud9 menggunakan ikon alternatif dan perilaku pemilihan file alternatif.

## Kolaborasi

Nonaktifkan peringatan keamanan kolaborasi

Ketika anggota baca/tulis ditambahkan ke lingkungan, AWS Cloud9 tidak menampilkan kotak dialog peringatan keamanan.

Tampilkan Info Kepengarangan

AWS Cloud9 menggarisbawahi teks yang dimasukkan oleh anggota lingkungan lain dengan sorotan terkait di selokan.

## Tree and Go panel

Lingkup Pergi ke Apa Pun ke Favorit

Buka File dalam jendela Go menampilkan hasil cakupan hanya untuk Favorit dalam jendela Lingkungan.

Aktifkan Pratinjau pada Pemilihan Pohon

AWS Cloud9 menampilkan file yang dipilih dengan satu klik, bukan klik dua kali.

Pola File Tersembunyi

Jenis file AWS Cloud9 untuk diperlakukan sebagai tersembunyi.

Mengungkapkan File Aktif di Pohon Proyek

AWS Cloud9 menyoroti file aktif di jendela Lingkungan.

Unduh File Sebagai

Perilaku AWS Cloud9 untuk digunakan saat men-download file.

Nilai-nilai yang valid meliputi:

- auto — AWS Cloud9 download file tanpa modifikasi.
- tar.gz - AWS Cloud9 mengunduh file sebagai terkompresi TAR berkas.
- zip - AWS Cloud9 mengunduh file sebagai .zip berkas.

## Temukan di File

Cari Di Jalur Ini Saat 'Proyek' Dipilih

Saat mencari di bar file, ketika Proyek dipilih untuk lingkup pencarian, jalur untuk pencarian.

Tampilkan Jalur Lengkap dalam Hasil

Menampilkan jalur lengkap ke setiap file yang cocok di tab Hasil Pencarian.

Hapus Hasil Sebelum Setiap Pencarian

Menghapus tab Hasil Pencarian dari hasil pencarian sebelumnya sebelum pencarian saat ini dimulai.

Gulir ke Bawah Saat Hasil Pencarian Masuk

Gulir tab Hasil Pencarian ke bagian bawah daftar hasil saat hasil pencarian diidentifikasi.

Buka File saat Menavigasi Hasil dengan (Atas dan Bawah)

Saat tombol panah atas dan bawah ditekan di tab Hasil Pencarian dalam daftar hasil, buka setiap file yang cocok.

## Meta Data

Maksimum Undo Stack Item di Meta Data

Jumlah maksimum item yang AWS Cloud9 disimpan dalam daftar tindakan yang dapat dibatalkan.

## Pengamat

Gabungkan File Otomatis Saat Terjadi Konflik

AWS Cloud9 mencoba menggabungkan file secara otomatis setiap kali terjadi konflik penggabungan.

## Terminal

Warna Teks

Warna teks dalam tab Terminal.

Warna Latar Belakang

Warna latar belakang di tab Terminal.

Warna Seleksi

Warna teks yang dipilih dalam tab Terminal.

Font Keluarga

Gaya font teks di tab Terminal.

Ukuran Font

Ukuran teks dalam tab Terminal.

Font Antialiased

AWS Cloud9 mencoba untuk menghaluskan tampilan teks di tab Terminal.

Kursor Berkedip

AWS Cloud9 terus mengedipkan kursor di tab Terminal.

Scrollback

Jumlah baris yang dapat Anda gulir ke atas atau balik melalui tab Terminal.

Gunakan AWS Cloud9 sebagai Editor Default

Digunakan AWS Cloud9 sebagai editor teks default.

## Output

Warna Teks

Warna teks dalam tab yang menampilkan output.

Warna Latar Belakang

Warna latar belakang teks di tab yang menampilkan output.

Warna Seleksi

Warna teks yang dipilih di tab yang menampilkan output.

Peringatkan Sebelum Menutup Konfigurasi Tanpa Nama

AWS Cloud9 meminta Anda untuk menyimpan tab konfigurasi yang belum disimpan sebelum ditutup.

Pertahankan log di antara proses

AWS Cloud9 menyimpan log dari semua percobaan yang dijalankan.

## Editor kode (Ace)

Tanda kurung pasangan otomatis, kutipan, dll.

AWS Cloud9 mencoba menambahkan karakter penutup yang cocok untuk setiap karakter awal terkait yang diketik di tab editor, seperti untuk tanda kurung, tanda kutip, dan tanda kurung.

Bungkus Seleksi dengan Tanda Kurung, Kutipan, dll.

AWS Cloud9 mencoba untuk menyisipkan karakter penutup yang cocok di akhir teks di tab editor setelah teks dipilih dan karakter awal terkait diketik, seperti untuk tanda kurung, tanda kutip, dan tanda kurung.

Kode Lipat

AWS Cloud9 mencoba untuk menampilkan, memperluas, menyembunyikan, atau menciutkan bagian kode di tab editor sesuai dengan aturan sintaks kode terkait.

Widget Lipat Pudar

AWS Cloud9 menampilkan kontrol pelipatan kode di selokan setiap kali Anda menjeda mouse di atas kontrol tersebut di tab editor.

Salin Dengan Pilihan Kosong

AWS Cloud9 memungkinkan Anda untuk Menyalin dan atau Memotong teks dan opsi ini menentukan apakah teks kosong akan disalin ke clipboard.

Seleksi Baris Penuh

AWS Cloud9 memilih seluruh baris yang diklik tiga kali di tab editor.

Sorot Garis Aktif

AWS Cloud9 menyoroti seluruh baris aktif di tab editor.

Sorot Garis Talang

AWS Cloud9 menyoroti lokasi di selokan di sebelah baris aktif di tab editor.

Tampilkan Karakter Tak Terlihat

AWS Cloud9 menampilkan apa yang dianggap sebagai karakter tak terlihat di tab editor, misalnya, carriage return dan line feed, spasi, dan tab.

Tampilkan Talang

AWS Cloud9 menampilkan selokan.

Tampilkan Nomor Baris

Perilaku untuk menampilkan nomor baris di saluran.

Nilai-nilai yang valid meliputi:

- Normal – Tampilkan nomor baris.
- Relatif – Tampilkan nomor baris relatif terhadap baris aktif.
- Tidak ada – Sembunyikan nomor baris.

Tampilkan Panduan Indent

AWS Cloud9 menampilkan panduan untuk lebih mudah memvisualisasikan teks indentasi di tab editor.

Sorot Kata yang Dipilih

AWS Cloud9 memilih seluruh kata yang diklik dua kali di tab editor.

Gulir Melewati Akhir Dokumen

Perilaku untuk mengizinkan pengguna menggulir melewati akhir file saat ini di tab editor.

Nilai-nilai yang valid meliputi:

- Mati – Jangan izinkan pengguliran melewati akhir file saat ini.
- Tinggi Setengah Editor – Memungkinkan pengguliran melewati akhir file saat ini hingga setengah tinggi layar editor.
- Tinggi Editor Maksimal – Memungkinkan pengguliran melewati akhir file saat ini hingga tinggi maksimal layar editor.

Animate Scrolling

AWS Cloud9 menerapkan perilaku animasi selama tindakan pengguliran di tab editor.

Font Keluarga

Gaya font yang digunakan dalam tab editor.

Ukuran Font

Ukuran font yang digunakan dalam tab editor.

Font Antialiased

AWS Cloud9 mencoba untuk menghaluskan tampilan teks di tab editor.

Tampilkan Margin Cetak

Menampilkan garis vertikal di tab editor setelah lokasi karakter yang ditentukan.

Kecepatan Gulir Mouse

Kecepatan relatif pengguliran mouse di tab editor. Nilai yang lebih besar menghasilkan pengguliran yang lebih cepat.

Gaya Kursor

Gaya dan perilaku pointer di tab editor.

Nilai yang valid meliputi:

- Ace — Menampilkan pointer sebagai bar vertikal yang relatif lebih lebar dari Slim.
- Slim — Tampilkan pointer sebagai bilah vertikal yang relatif ramping.
- Smooth — Tampilkan pointer sebagai bilah vertikal yang relatif lebih lebar dari Slim dan yang berkedip lebih mulus daripada Slim.
- Smooth and Slim — Tampilkan pointer sebagai bilah vertikal yang relatif ramping yang berkedip lebih mulus daripada Slim.
- Lebar - Menampilkan pointer sebagai bilah vertikal yang relatif lebar.

Gabungkan Undo Delta

- Selalu – Izinkan konflik gabungan untuk dikembalikan.
- Tidak pernah – Jangan izinkan konflik gabungan untuk dikembalikan.

- Timed — Izinkan konflik gabungan dikembalikan setelah periode tertentu.

Aktifkan Pembungkus Untuk Dokumen Baru

AWS Cloud9 membungkus kode dalam file baru.

## Input

Lengkap Saat Anda Mengetik

AWS Cloud9 mencoba untuk menampilkan kemungkinan penyelesaian teks saat Anda mengetik.

Selesaikan Pada Enter

AWS Cloud9 mencoba untuk menampilkan kemungkinan penyelesaian teks setelah Anda menekan Enter.

Sorot Variabel Di Bawah Kursor

AWS Cloud9 menyoroti semua referensi dalam kode ke variabel yang dipilih.

Gunakan Cmd-Click untuk Langsung ke Definisi

AWS Cloud9 pergi ke definisi asli untuk kode yang dipilih sambil menekan dan menahan Command untuk Mac atau Ctrl untuk Windows.

## Petunjuk dan peringatan

Aktifkan Petunjuk dan Peringatan

AWS Cloud9 menampilkan pesan petunjuk dan peringatan yang berlaku.

Tampilkan Perbaikan Cepat yang Tersedia di Klik

AWS Cloud9 menampilkan tip alat dengan saran refactoring ketika Anda mengklik kata kunci dalam kode Anda.

Abaikan Pesan yang Cocokkan Regex

AWS Cloud9 tidak menampilkan pesan apa pun yang cocok dengan ekspresi reguler yang ditentukan. Untuk informasi selengkapnya, lihat Menulis pola ekspresi reguler dalam topik Ekspresi JavaScript Reguler di Jaringan Pengembang Mozilla.

## Jalankan dan debug

### Simpan Semua Tab yang Belum Disimpan Sebelum Menjalankan

Sebelum menjalankan kode terkait, AWS Cloud9 mencoba menyimpan semua file yang belum disimpan dengan tab terbuka.

## Pratinjau

### Pratinjau Aplikasi Berjalan

AWS Cloud9 mencoba untuk menampilkan pratinjau output untuk kode di tab aktif setiap kali tombol Pratinjau dipilih.

### Previewer Default

Format ini AWS Cloud9 digunakan untuk melihat output kode.

Nilai yang valid meliputi:

- Raw – Mencoba menampilkan output kode dalam format biasa.
- Browser — Mencoba menampilkan output kode dalam format yang disukai untuk browser web.

### Saat Menyimpan Reload Previewer

Perilaku ini AWS Cloud9 digunakan untuk melihat pratinjau keluaran kode setiap kali file kode disimpan.

Nilai-nilai yang valid meliputi:

- Hanya pada Ctrl-Enter – Mencoba mempratinjau output kode kapan pun Ctrl+Enter ditekan untuk tab kode saat ini.
- Selalu – Mencoba untuk mempratinjau output kode setiap kali file kode disimpan.

## Membangun

### Secara Otomatis Membangun File yang Didukung

AWS Cloud9 mencoba untuk secara otomatis membangun kode saat ini jika tindakan build dimulai dan kode dalam format yang didukung.

# Bekerja dengan pengaturan AWS proyek dan pengguna di AWS Cloud9 IDE

Layanan AWS pengaturan, yang terletak di panel AWS Pengaturan pada tab Preferensi, termasuk pengaturan berikut:

- Yang digunakan Wilayah AWS untuk jendela AWS Sumber Daya
- Apakah akan menggunakan kredenal sementara AWS terkelola
- Apakah akan menampilkan AWS Serverless Application Model (AWS SAM) editor template dalam teks biasa atau mode visual

Untuk melihat atau mengubah pengaturan ini, pilih AWS Cloud9, Preferensi di bilah menu IDE untuk lingkungan.

Dalam daftar berikut, pengaturan tingkat proyek hanya berlaku untuk lingkungan pengembangan saat ini AWS Cloud9 . Sebaliknya, pengaturan tingkat pengguna berlaku di setiap lingkungan yang terkait dengan pengguna IAM Anda. Untuk informasi selengkapnya, lihat Menerapkan Pengaturan Proyek Saat Ini untuk Lingkungan ke Lingkungan Lain dan Berbagi Pengaturan Pengguna Anda dengan Pengguna Lain.

- Pengaturan Tingkat Proyek
- Pengaturan Tingkat Pengguna

## Pengaturan tingkat proyek

AWS Wilayah

Yang digunakan Wilayah AWS untuk bagian Lambda dari jendela AWS Sumber Daya.

AWS kredensi sementara yang dikelola

Jika diaktifkan, kredenal sementara AWS terkelola akan digunakan saat Anda memanggil Layanan AWS dari AWS CLI, kode AWS CloudShell, atau AWS SDK dari lingkungan. Untuk informasi selengkapnya, lihat Kredensial Sementara terkelola AWS.

## Pengaturan tingkat pengguna

Gunakan editor AWS SAM visual

Jika diaktifkan, editor templat AWS Serverless Application Model (AWS SAM) ditampilkan dalam mode visual saat Anda menggunakan bagian Lambda di jendela AWS Sumber Daya. Jika dimatikan, editor ditampilkan dalam mode teks.

# Bekerja dengan Keybindings di IDE AWS Cloud9

Keybindings menentukan kombinasi tombol pintas Anda. Keybindings berlaku di setiap lingkungan AWS Cloud9 pengembangan yang terkait dengan pengguna IAM Anda. Saat Anda membuat perubahan pada ikatan kunci, AWS Cloud9 dorong perubahan tersebut ke cloud, dan kaitkan dengan pengguna IAM Anda. AWS Cloud9 juga terus memindai cloud untuk perubahan keybinding yang terkait dengan pengguna IAM Anda, dan menerapkan perubahan tersebut ke lingkungan Anda saat ini.

Anda dapat membagikan keybinding Anda ke pengguna lain.

- Lihat atau Ubah Keybindings Anda
- Bagikan Keybinding Anda dengan Pengguna Lain
- Ubah Mode Keyboard Anda
- Ubah Keybindings Sistem Operasi Anda
- Ubah Keybindings Tertentu
- Hapus Semua Keybindings Kustom Anda

## Melihat atau mengubah Keybindings Anda

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Untuk melihat keybinding Anda di setiap lingkungan Anda, pada tab Preferensi, di panel navigasi samping, pilih Keybinding.
3. Untuk mengubah ikatan kunci di setiap lingkungan Anda, di panel Keybindings, ubah pengaturan yang Anda inginkan.
4. Untuk menerapkan perubahan Anda ke lingkungan apa pun, cukup buka lingkungan tersebut. Jika lingkungan tersebut sudah terbuka, segarkan tab peramban web untuk lingkungan tersebut.

Untuk informasi selengkapnya, lihat berikut ini:

- Referensi Keybindings Default macOS
- Referensi Keybindings macOS Vim
- Referensi Keybindings macOS Emacs
- Referensi Keybindings Sublime macOS
- Referensi Keybinding Default Windows/Linux
- Referensi Pengikatan Kunci Vim Windows/Linux
- Referensi Keybinding Windows /Linux Emacs
- Referensi Keybinding Sublime Windows/Linux

## Bagikan Keybindings Anda dengan pengguna lain

1. Di lingkungan sumber dan target, pada bilah menu AWS Cloud9 IDE, pilih AWS Cloud9, Buka Peta Kunci Anda.
2. Di lingkungan sumber, salin konten tab keybindings.settings yang ditampilkan.
3. Di lingkungan target, timpa konten tab pengaturan.keybinding dengan konten yang disalin dari lingkungan sumber.
4. Dalam lingkungan target, simpan tab pengaturan.keybinding.

## Ubah mode Keyboard Anda

Anda dapat mengubah mode keyboard yang digunakan AWS Cloud9 IDE untuk berinteraksi dengan teks di editor di setiap lingkungan yang terkait dengan pengguna IAM Anda.

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, di panel navigasi samping, pilih Keybinding.
3. Untuk Mode Keyboard, pilih salah satu mode keyboard berikut:
   - Default untuk menggunakan satu set keybindings default.
   - Vim untuk menggunakan mode Vim. Untuk informasi lebih lanjut, lihat situs web file bantuan Vim.
   - Emacs untuk menggunakan mode Emacs. Untuk informasi selengkapnya, lihat Editor Emacs di situs web Sistem Operasi GNU.

- Sublime untuk menggunakan mode Sublime. Untuk informasi lebih lanjut, lihat situs web Dokumentasi Teks Sublime.

## Ubah Keybindings sistem operasi Anda

Anda dapat mengubah kumpulan keybinding sistem operasi yang dikenali AWS Cloud9 IDE di setiap lingkungan yang terkait dengan pengguna IAM Anda.

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, di panel navigasi samping, pilih Keybinding.
3. Untuk Sistem Operasi, pilih salah satu sistem operasi ini:
   - Otomatis bagi AWS Cloud9 IDE untuk mencoba mendeteksi kumpulan keybinding sistem operasi mana yang akan digunakan.
   - macOS untuk AWS Cloud9 IDE menggunakan keybinding yang tercantum dalam format macOS.
   - Windows/Linux untuk AWS Cloud9 IDE untuk menggunakan keybinding yang tercantum dalam format Windows dan Linux.

## Ubah Keybindings tertentu

Anda dapat mengubah keybinding individual di setiap lingkungan yang terkait dengan pengguna IAM Anda.

Untuk mengubah satu pengikatan kunci secara bersamaan

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, di panel navigasi samping, pilih Keybinding.
3. Dalam daftar keybinding, buka (klik dua kali) keybinding di kolom Keystroke yang ingin Anda ubah.
4. Gunakan keyboard untuk menentukan kombinasi tombol pengganti, lalu tekan `Enter`.

> **Note**
>
> Untuk sepenuhnya menghapus kombinasi tombol saat ini, tekan `Backspace` untuk Windows atau Linux, atau `Delete` untuk macOS.

Untuk mengubah beberapa ikatan kunci secara bersamaan

1. Pada bilah menu, pilih AWS Cloud9, Buka Keymap Anda.
2. Dalam `keybindings.settings` file, tentukan setiap keybinding yang akan diubah. Berikut ini adalah contoh sintaks.

```
[
  {
    "command": "addfavorite",
    "keys": {
      "win": ["Ctrl-Alt-F"],
      "mac": ["Ctrl-Option-F"]
    }
  },
  {
    "command": "copyFilePath",
    "keys": {
      "win": ["Ctrl-Shift-F"],
      "mac": ["Alt-Shift-F"]
    }
  }
]
```

   Dalam contoh, `addFavorite` dan `copyFilePath` adalah nama ikatan tombol di kolom Keystroke di panel Keybinding pada tab Preferensi. Keybinding yang Anda inginkan adalah `win` dan `mac` untuk Windows atau Linux dan macOS, masing-masing.

   Untuk menerapkan perubahan, simpan file `keybindings.settings`. Perubahan Anda muncul di panel Keybindings setelah penundaan singkat.

## Hapus semua Keybindings kustom Anda

Anda dapat menghapus semua ikatan kunci khusus dan mengembalikan semua ikatan kunci ke nilai defaultnya, di setiap lingkungan yang terkait dengan pengguna IAM Anda.

> **Warning**
>
> Anda tidak dapat membatalkan tindakan ini.

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, di panel navigasi samping, pilih Keybinding.
3. Pilih Atur ulang ke Default.

# Bekerja dengan tema di AWS Cloud9 IDE

Tema mendefinisikan warna IDE Anda secara keseluruhan. Hal ini berlaku di setiap lingkungan pengembangan AWS Cloud9 yang terkait dengan pengguna IAM Anda. Saat Anda membuat perubahan pada tema Anda, AWS Cloud9 dorong perubahan tersebut ke cloud, dan kaitkan dengan pengguna IAM Anda. AWS Cloud9 juga terus memindai cloud untuk perubahan pada tema yang terkait dengan pengguna IAM Anda. AWS Cloud9 menerapkan perubahan tersebut ke lingkungan Anda saat ini.

- Lihat atau ubah tema Anda
- Pengaturan tema keseluruhan Anda dapat mengubah
- Penggantian tema

## Melihat atau mengubah tema Anda

1. Pada bar menu, pilih AWS Cloud9, Preferensi.
2. Untuk melihat keybinding Anda di setiap lingkungan Anda, pada tab Preferensi, di panel navigasi samping, pilih Tema.
3. Untuk mengubah tema Anda di setiap lingkungan Anda, di panel Tema, ubah pengaturan yang Anda inginkan. Untuk mengubah bagian tema Anda dengan menggunakan kode, pilih tautan lembar gaya Anda.
4. Untuk menerapkan perubahan Anda ke lingkungan Anda, buka lingkungan itu. Jika lingkungan tersebut sudah terbuka, segarkan tab peramban web untuk lingkungan tersebut.

## Pengaturan tema secara keseluruhan yang dapat Anda ubah

Anda dapat mengubah jenis pengaturan tema keseluruhan berikut pada tab Preferensi di panel Tema.

Tema datar

Menerapkan tema datar bawaan di seluruh AWS Cloud9 IDE.

Tema Klasik

Menerapkan tema klasik bawaan yang dipilih di seluruh AWS Cloud9 IDE.

Sintaks Tema

Menerapkan tema yang dipilih ke file kode di seluruh AWS Cloud9 IDE.

## Penggantian tema

**Important**

AWS Cloud9 tidak lagi mendukung fitur yang memungkinkan pengguna untuk mengganti tema IDE dengan memperbarui `styles.css` file. Pengguna dapat terus melihat, mengedit, dan menyimpan `styles.css` file menggunakan editor. Tapi, tidak ada penggantian tema yang diterapkan saat AWS Cloud9 IDE dimuat.
Jika AWS Cloud9 mendeteksi bahwa `styles.css` file telah dimodifikasi, pesan berikut ditampilkan di IDE:
Support untuk penggantian tema telah dihentikan. Isi file styles.css ini tidak akan lagi diterapkan saat memuat AWS Cloud9 IDE.
Jika Anda perlu menggunakan lembar gaya untuk menentukan tema IDE, silakan hubungi kami secara langsung.

# Mengelola skrip inisialisasi di IDE AWS Cloud9

**Important**

AWS Cloud9 tidak lagi mendukung fitur eksperimental yang memungkinkan pengguna untuk menyesuaikan skrip inisialisasi. Skrip ini secara otomatis dijalankan di IDE. Pengguna dapat terus melihat, mengedit, dan menyimpan `init.js` file menggunakan editor. Namun, skrip inisialisasi yang disesuaikan tidak lagi diizinkan untuk dijalankan dan tidak dapat mengubah perilaku IDE.
Jika AWS Cloud9 mendeteksi bahwa `init.js` file telah dimodifikasi, pesan berikut ditampilkan di IDE:

Support untuk tulisan inisialisasi telah dihentikan. Isi file init.js ini tidak akan lagi dieksekusi saat memuat AWS Cloud9 IDE.
Jika Anda perlu menjalankan skrip inisialisasi khusus untuk IDE, hubungi kami.

Tulisan inisialisasi mendefinisikan kode inisialisasi untuk dijalankan di IDE Anda setelah semua plugin dimuat. Ini berlaku di setiap lingkungan AWS Cloud9 pengembangan yang terkait dengan pengguna IAM Anda. AWS Cloud9 juga terus memindai perubahan pada skrip inisialisasi dan memperingatkan pengguna jika terjadi modifikasi.

### Membuka tulisan inisialisasi Anda

Untuk membuka tulisan inisialisasi Anda, pada bilah menu, pilih AWS Cloud9, Buka Tulisan Inisialisasi Anda.

**Important**

Anda dapat mengedit dan menyimpan `init.js` file menggunakan editor, tetapi skrip khusus Anda tidak diizinkan untuk berjalan di IDE.

## Referensi Keybindings Default macOS untuk IDE AWS Cloud9

Langkah ini menunjukkan cara memilih MacOS Default Keybindings Reference untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, pilih Keybinding.
3. Untuk Mode Keyboard pilih Default.
4. Untuk Sistem Operasi, pilih MacOS.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard default untuk sistem operasi macOS di AWS Cloud9 Integrated Development Environment (IDE):

- Umum
- Tab
- Panel

- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Command-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Control-Space` \| `Option-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Control-Shift-Space` \| `Option-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Command-C` | `copy` |
| Memotong seleksi ke clipboard | `Command-X` | `cut` |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Command-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Control-Option-G` | `findAll` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk | `Command-G` | `findnext` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| menemukan kueri yang Anda masukkan terakhir | | |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Command-Shift-G` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Command-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Command-L` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Command-Shift-U` | `lambdaUploadFunction` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuat file baru | `Control-N` | `newfile` |
| Menampilkan tab Preferensi | `Command-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Command-Option-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Command-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Command-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Command-Shift-Z \| Command-Y` | `redo` |
| Merefresh panel pratinjau | `Command-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Option-Command-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Option-Command-F` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Command-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Command-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Control-Shift-Q` | `reverttosaved` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Option-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Command-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Command-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Shift-Command-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Command-Option-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Command-Z` | `undo` |

## Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Option-Control-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Option-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Command-Control-W` | `closepane` |
| Menutup tab saat ini | `Option-W` | `closetab` |
| Pergi satu panel ke bawah | `Control-Command-Down` | `gotopanedown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi satu panel ke kiri | `Control-Command-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Control-Command-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Control-Command-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Command-[` | `gototableft` |
| Pergi satu tab ke kanan | `Command-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Command-Option-Shift-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Command-Option-Shift-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Command-Option-Shift-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Command-Option-Shift-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Option-Esc` | `nextpane` |
| Pergi ke tab berikutnya | `Option-Tab` | `nexttab` |
| Pergi ke panel sebelumnya | `Option-Shift-Esc` | `previouspane` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke tab sebelumnya | `Option-Shift-Tab` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Option-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Command-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Command-0` | `tab0` |
| Pergi ke tab pertama | `Command-1` | `tab1` |
| Pergi ke tab kedua | `Command-2` | `tab2` |
| Pergi ke tab ketiga | `Command-3` | `tab3` |
| Pergi ke tab keempat | `Command-4` | `tab4` |
| Pergi ke tab kelima | `Command-5` | `tab5` |
| Pergi ke tab keenam | `Command-6` | `tab6` |
| Pergi ke tab ketujuh | `Command-7` | `tab7` |
| Pergi ke tab kedelapan | `Command-8` | `tab8` |
| Pergi ke tab kesembilan | `Command` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Command-E\|Command-P` | `gotoanything` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Command-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Command-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Command-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Command-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `Control-Esc` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Command-U` | `toggletree` |

## Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Control-Option-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah | `Control-Option-Shi ft-Up` | `addCursorAboveSkip Current` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| ditambahkan, memindahk an kursor kedua ke atas satu baris | | |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Control-Option-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Control-Option-Shi ft-Down` | `addCursorBelowSkip Current` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Control-Option-A` | `alignCursors` |
| Backspace satu spasi | `Control-Backspace | Shift-Backspace | Backspace` | `backspace` |
| Meng-indent seleksi satu tab | `Control-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Control-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Command-Z | Command-S hift-Z | Command-Y` | `cancelBrowserUndoI nAce` |
| Memusatkan seleksi | `Control-L` | `centerselection` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Command-Option-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Command-Option-Up` | `copylinesup` |
| Menghapus satu spasi | `Delete \| Control-Delete \| Shift-Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Command-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Command-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Control-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih, atau jika unit terlipat yang dipilih, membuka lipatannya | `Command-Option-L \| Command-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat | `Control-Command-Option-0` | `foldall` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Command-Option-0` | `foldOther` |
| Turun satu baris | `Down \| Control-N` | `golinedown` |
| Naik satu baris | `Up \| Control-P` | `golineup` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke akhir file | `Command-End \| Command-Down` | `gotoend` |
| Pergi ke kiri satu spasi | `Left \| Control-B` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Command-Right \| End \| Control-E` | `gotolineend` |
| Pergi ke awal baris saat ini | `Command-Left \| Home \| Control-A` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `F4` | `goToNextError` |
| Turun satu halaman | `Page Down \| Control-V` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Shift-F4` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right \| Control-F` | `gotoright` |
| Pergi ke awal file | `Command-Home \| Command-Up` | `gotostart` |
| Pergi satu kata ke kiri | `Option-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Option-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Control-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Command-+ \| Command-=` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Down` | `modifyNumberDown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Option-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Option-Up` | `movelinesup` |
| Meng-outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau menonakti fkan jika aktif | `Insert` | `overwrite` |
| Turun satu halaman | `Option-Page Down` | `pagedown` |
| Naik satu halaman | `Option-Page Up` | `pageup` |
| Menghapus baris saat ini | `Command-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Control-K` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Command-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Option-Backspace \| Control-Option-Bac kspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Option-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Command-Shift-E` | `replaymacro` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memilih semua konten yang dapat dipilih | `Command-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down \| Control-Shift-N` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left \| Control-Shift-B` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Control-Option-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Control-Option-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Control-Option-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Control-Option-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Control-G` | `selectOrFindNext` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Control-Shift-G` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Command-Shift-End \| Command-Shift-Down` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Command-Shift-Right \| Shift-End \| Control-Shift-E` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Command-Shift-Left \| Control-Shift-A` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Control-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Command-Shift-Home \| Command-Shift-Up` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up \| Control-Shift-Up` | `selectup` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Option-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Option-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Command-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Command--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Command-Option-S` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Control-Option-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Control-O` | `splitline` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Command-Shift-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Command-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Option-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Command-Option-E` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Control-W` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Control-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Control-U` | `touppercase` |
| Menranspos seleksi | `Control-T` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Command-Option-Shift-L \| Command-Shift-F1` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Command-Option-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana | `Shift-Command-Y` | `emmet_evaluate_math_expression` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | | |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Control-Option-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Command-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Command-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Control-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Option-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Option-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Command-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8 \| Command-\` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Option-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11 \| Command-;` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11 \| Command-Shift-'` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10 \| Command-'` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Control-Shift-C` | `stopbuild` |

# Referensi Keybindings macOS Vim untuk IDE AWS Cloud9

Langkah ini menunjukkan cara memilih MacOS Vim Keybindings Reference untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, pilih Keybinding.
3. Untuk Mode Keyboard pilih Vim.

4. Untuk Sistem Operasi, pilih MacOS.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Vim untuk sistem operasi macOS di IDE: AWS Cloud9

- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Command-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Control-Space` \| `Option-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Control-Shift-Space` \| `Option-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Command-C` | `copy` |
| Memotong seleksi ke clipboard | `Command-X` | `cut` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Command-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Control-Option-G` | `findAll` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Command-G` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Command-Shift-G` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Command-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Command-L` | `gotoline` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Command-Shift-U` | `lambdaUploadFunction` |
| Membuat file baru | `Control-N` | `newfile` |
| Menampilkan tab Preferensi | `Command-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Command-Option-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Command-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Command-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Command-Shift-Z \| Command-Y` | `redo` |
| Merefresh panel pratinjau | `Command-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Option-Command-R` | `renameVar` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Option-Command-F` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Command-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Command-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Control-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Option-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Command-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Command-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Shift-Command-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Command-Option-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Command-Z` | `undo` |

# Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Option-Control-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Option-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Command-Control-W` | `closepane` |
| Menutup tab saat ini | `Option-W` | `closetab` |
| Pergi satu panel ke bawah | `Control-Command-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Control-Command-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Control-Command-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Control-Command-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Command-[` | `gototableft` |
| Pergi satu tab ke kanan | `Command-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Command-Option-Shift-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Command-Option-Shift-Left` | `movetableft` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Command-Option-Shift-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Command-Option-Shift-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Option-Esc` | `nextpane` |
| Pergi ke tab berikutnya | `Option-Tab` | `nexttab` |
| Pergi ke panel sebelumnya | `Option-Shift-Esc` | `previouspane` |
| Pergi ke tab sebelumnya | `Option-Shift-Tab` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Option-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Command-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Command-0` | `tab0` |
| Pergi ke tab pertama | `Command-1` | `tab1` |
| Pergi ke tab kedua | `Command-2` | `tab2` |
| Pergi ke tab ketiga | `Command-3` | `tab3` |
| Pergi ke tab keempat | `Command-4` | `tab4` |
| Pergi ke tab kelima | `Command-5` | `tab5` |
| Pergi ke tab keenam | `Command-6` | `tab6` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke tab ketujuh | `Command-7` | `tab7` |
| Pergi ke tab kedelapan | `Command-8` | `tab8` |
| Pergi ke tab kesembilan | `Command` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Command-E\|Command-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Command-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Command-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Command-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Command-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `Control-Esc` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Command-U` | `toggletree` |

# Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di atas kursor itu | `Control-Option-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke atas satu baris | `Control-Option-Shift-Up` | `addCursorAboveSkipCurrent` |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Control-Option-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Control-Option-Shift-Down` | `addCursorBelowSkipCurrent` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Control-Option-A` | `alignCursors` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Backspace satu spasi | `Control-Backspace \| Shift-Backspace \| Backspace` | `backspace` |
| Menyeleksi indent satu tab | `Control-]` | `blockindent` |
| Menyeleksi outdent satu tab | `Control-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Command-Z \| Command-Shift-Z \| Command-Y` | `cancelBrowserUndoInAce` |
| Memusatkan seleksi | `Control-L` | `centerselection` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Command-Option-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Command-Option-Up` | `copylinesup` |
| Menghapus satu spasi | `Delete \| Control-Delete \| Shift-Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Command-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Command-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Control-Shift-M` | `expandToMatching` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melipat kode yang dipilih, atau jika unit terlipat yang dipilih, membuka lipatannya | `Command-Option-L \| Command-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat | `Control-Command-Option-0` | `foldall` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Command-Option-0` | `foldOther` |
| Turun satu baris | `Down \| Control-N` | `golinedown` |
| Naik satu baris | `Up \| Control-P` | `golineup` |
| Pergi ke akhir file | `Command-End \| Command-Down` | `gotoend` |
| Pergi ke kiri satu spasi | `Left \| Control-B` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Command-Right \| End \| Control-E` | `gotolineend` |
| Pergi ke awal baris saat ini | `Command-Left \| Home \| Control-A` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `F4` | `goToNextError` |
| Turun satu halaman | `Page Down \| Control-V` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Shift-F4` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right \| Control-F` | `gotoright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke awal file | `Command-Home \| Command-Up` | `gotostart` |
| Pergi satu kata ke kiri | `Option-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Option-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Control-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Command-+ \| Command-=` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Option-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Option-Up` | `movelinesup` |
| Menyeleksi outdent satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau menonakti fkan jika aktif | `Insert` | `overwrite` |
| Turun satu halaman | `Option-Page Down` | `pagedown` |
| Naik satu halaman | `Option-Page Up` | `pageup` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menghapus baris saat ini | `Command-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Control-K` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Command-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Option-Backspace \| Control-Option-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Option-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Command-Shift-E` | `replaymacro` |
| Memilih semua konten yang dapat dipilih | `Command-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down \| Control-Shift-N` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left \| Control-Shift-B` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Control-Option-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Control-Option-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Control-Option-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Control-Option-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Control-G` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Control-Shift-G` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Command-Shift-End \| Command-Shift-Down` | `selecttoend` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Command-Shift-Right \| Shift-End \| Control-Shift-E` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Command-Shift-Left \| Control-Shift-A` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Control-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Command-Shift-Home \| Command-Shift-Up` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up \| Control-Shift-P` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Option-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Option-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Command-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Command--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Command-Option-S` | `sortlines` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor di akhir baris saat ini | `Control-Option-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Control-O` | `splitline` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Command-Shift-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Command-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Option-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Command-Option-E` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Control-W` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Control-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Control-U` | `touppercase` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menranspos seleksi | `Control-T` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Command-Option-Shift-L \| Command-Shift-F1` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Command-Option-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Command-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Control-Option-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Command-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Command-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan | `Shift-Control-A` | `emmet_wrap_with_abbreviation` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | | |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Option-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Option-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Command-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8 \| Command-\` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Option-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11 \| Command-;` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11 \| Command-Shift-'` | `stepout` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melangkahi ekspresi saat ini di tumpukan | `F10 \| Command-'` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Control-Shift-C` | `stopbuild` |

# Referensi Keybindings macOS Emacs untuk IDE AWS Cloud9

Langkah ini menunjukkan cara memilih MacOS Emacs Keybindings Reference untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, memilih Keybinding.
3. Untuk Mode Keyboard memilih Emacs.
4. Untuk Sistem Operasi, pilih MacOS.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Emacs untuk sistem operasi macOS di IDE: AWS Cloud9

- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

# Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Command-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Control-Space` \| `Option-Space` | `complete` |
| Kode lengkap, dan kemudian menimpa | `Control-Shift-Space` \| `Option-Shift-Space` | `completeoverwrite` |
| Menyalin pemilihan ke clipboard | `Command-C` | `copy` |
| Memotong seleksi ke clipboard | `Command-X` | `cut` |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Command-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Control-Option-G` | `findAll` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Command-G` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen | `Command-Shift-G` | `findprevious` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| saat ini untuk menemukan kueri yang Anda masukkan terakhir | | |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Command-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Command-L` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Command-Shift-U` | `lambdaUploadFunction` |
| Membuat file baru | `Control-N` | `newfile` |
| Menampilkan tab Preferensi | `Command-,` | `openpreferences` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Command-Option-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Command-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Command-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Command-Shift-Z \| Command-Y` | `redo` |
| Merefresh panel pratinjau | `Command-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Option-Command-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Option-Command-F` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Command-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Command-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Control-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Option-Shift-Q` | `reverttosavedall` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyimpan file saat ini ke disk | `Command-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Command-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Shift-Command-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Command-Option-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Command-Z` | `undo` |

## Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Option-Control-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Option-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Command-Control-W` | `closepane` |
| Menutup tab saat ini | `Option-W` | `closetab` |
| Pergi satu panel ke bawah | `Control-Command-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Control-Command-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Control-Command-Right` | `gotopaneright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi satu panel ke atas | `Control-Command-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Command-[` | `gototableft` |
| Pergi satu tab ke kanan | `Command-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Command-Option-Shift-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Command-Option-Shift-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Command-Option-Shift-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Command-Option-Shift-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Option-Esc` | `nextpane` |
| Pergi ke tab berikutnya | `Option-Tab` | `nexttab` |
| Pergi ke panel sebelumnya | `Option-Shift-Esc` | `previouspane` |
| Pergi ke tab sebelumnya | `Option-Shift-Tab` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Option-Shift-T` | `reopenLastTab` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan tab saat ini di pohon file | `Command-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Command-0` | `tab0` |
| Pergi ke tab pertama | `Command-1` | `tab1` |
| Pergi ke tab kedua | `Command-2` | `tab2` |
| Pergi ke tab ketiga | `Command-3` | `tab3` |
| Pergi ke tab keempat | `Command-4` | `tab4` |
| Pergi ke tab kelima | `Command-5` | `tab5` |
| Pergi ke tab keenam | `Command-6` | `tab6` |
| Pergi ke tab ketujuh | `Command-7` | `tab7` |
| Pergi ke tab kedelapan | `Command-8` | `tab8` |
| Pergi ke tab kesembilan | `Command` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Command-E\|Command-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Command-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Command-O` | `gotofile` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Command-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Command-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `Control-Esc` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Command-U` | `toggletree` |

## Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Control-Option-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Control-Option-Shi ft-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, | `Control-Option-Down` | `addCursorBelow` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | | |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Control-Option-Shift-Down` | `addCursorBelowSkipCurrent` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Control-Option-A` | `alignCursors` |
| Backspace satu spasi | `Control-Backspace \| Shift-Backspace \| Backspace` | `backspace` |
| Menyeleksi indent satu tab | `Control-]` | `blockindent` |
| Menyeleksi outdent satu tab | `Control-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Command-Z \| Command-Shift-Z \| Command-Y` | `cancelBrowserUndoInAce` |
| Memusatkan seleksi | `Control-L` | `centerselection` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Command-Option-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Command-Option-Up` | `copylinesup` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menghapus satu spasi | `Delete \| Control-D elete \| Shift-Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Command-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Command-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Control-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Command-Option-L \| Command-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat | `Control-Command-Op tion-0` | `foldall` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Command-Option-0` | `foldOther` |
| Turun satu baris | `Down \| Control-N` | `golinedown` |
| Naik satu baris | `Up \| Control-P` | `golineup` |
| Pergi ke akhir file | `Command-End \| Command-Down` | `gotoend` |
| Pergi ke kiri satu spasi | `Left \| Control-B` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Command-Right \| End \| Control-E` | `gotolineend` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke awal baris saat ini | `Command-Left \| Home \| Control-A` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `F4` | `goToNextError` |
| Turun satu halaman | `Page Down \| Control-V` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Shift-F4` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right \| Control-F` | `gotoright` |
| Pergi ke awal file | `Command-Home \| Command-Up` | `gotostart` |
| Pergi satu kata ke kiri | `Option-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Option-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Control-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Command-+ \| Command-=` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Option-Down` | `movelinesdown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memindahkan seleksi ke atas satu baris | `Option-Up` | `movelinesup` |
| Meng-outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Turun satu halaman | `Option-Page Down` | `pagedown` |
| Naik satu halaman | `Option-Page Up` | `pageup` |
| Menghapus baris saat ini | `Command-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Control-K` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Command-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Option-Backspace \| Control-Option-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Option-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Command-Shift-E` | `replaymacro` |
| Memilih semua konten yang dapat dipilih | `Command-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down \| Control-Shift-N` | `selectdown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left \| Control-Shift-B` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Control-Option-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Control-Option-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Control-Option-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Control-Option-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Control-G` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Control-Shift-G` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Command-Shift-End \| Command-Shift-Down` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Command-Shift-Right \| Shift-End \| Control-Shift-E` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Command-Shift-Left \| Control-Shift-A` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Control-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Command-Shift-Home \| Command-Shift-Up` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up \| Control-Shift-Up` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Option-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Option-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Command-,` | `showSettingsMenu` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Command--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Command-Option-S` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Control-Option-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Control-O` | `splitline` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Command-Shift-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Command-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Option-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Command-Option-E` | `togglerecording` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Control-W` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Control-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Control-U` | `touppercase` |
| Mentranspos seleksi | `Control-T` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Command-Option-Shift-L \| Command-Shift-F1` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Command-Option-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Command-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Control-Option-E` | `emmet_expand_abbreviation` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Command-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Command-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Control-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Option-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Option-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Command-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8 \| Command-\` | `resume` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menjalankan atau men-debug aplikasi saat ini | `Option-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11 \| Command-;` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11 \| Command-Shift-'` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10 \| Command-'` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Control-Shift-C` | `stopbuild` |

# Referensi Keybindings Sublime macOS untuk IDE AWS Cloud9

Langkah ini menunjukkan cara memilih MacOS Sublime Keybindings Reference untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, pilih Keybinding.
3. Untuk Mode Keyboard pilih Sublime.
4. Untuk Sistem Operasi, memilih MacOS.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Sublime untuk sistem operasi macOS di IDE: AWS Cloud9

- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug


## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Command-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Control-Space \| Option-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Control-Shift-Space \| Option-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Command-C` | `copy` |
| Memotong pemilihan ke clipboard | `Command-X` | `cut` |
| Menghapus dari kursor ke awal baris | `Command-K Command-Backspace \| Command-Backspace` | `delete_to_hard_bol` |
| Menghapus dari kursor ke akhir baris | `Command-K Command-K \| Command-Delete \| Control-K` | `delete_to_hard_eol` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memperluas kode, jika berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Command-F` | `find` |
| Menyorot semua kecocokan untuk seleksi | `Control-Command-G` | `find_all_under` |
| Menyorot kecocokan berikutnya untuk seleksi | `Option-Command-G` | `find_under` |
| Sorot di sekitar kursor dan semua kecocokan untuk sorotan | `Command-D` | `find_under_expand` |
| Sorot di sekitar kursor dan garis besar semua kecocokan untuk sorotan | `Command-K Command-D` | `find_under_expand_skip` |
| Menyorot kecocokan berikutnya untuk seleksi | `Shift-Option-Command-G` | `find_under_previous` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Control-Option-G` | `findAll` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Command-G` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Shift-Command-G` | `findprevious` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Control-Option-F` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Control-G` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F12 \| Command-Option-Down` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba unggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Command-Shift-U` | `lambdaUploadFunction` |
| Pergi ke akhir kata saat ini | `Option-Right` | `moveToWordEndRight` |
| Pergi ke awal kata saat ini | `Option-Left` | `moveToWordStartLeft` |
| Membuat file baru | `Control-N` | `newfile` |
| Menampilkan tab Preferensi | `Command-,` | `openpreferences` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Command-Option-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Command-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Command-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Command-Shift-Z \| Command-Y` | `redo` |
| Merefresh panel pratinjau | `Command-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Option-Command-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ganti dengan ekspresi | `Command-Option-F` | `replace` |
| Ganti semua menemukan ekspresi cocok dengan ganti dengan ekspresi di find dan replace bar | `Control-Option-Enter` | `replaceall` |
| Ganti berikutnya menemukan ekspresi cocok dengan ganti dengan ekspresi di find dan replace bar | `Command-Option-E` | `replacenext` |
| Menjalankan kembali skrip inisialisasi Anda | `Command-Enter` | `rerunInitScript` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memulai ulang lingkungan | `Command-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Control-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Option-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Command-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Command-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Command-Shift-F` | `searchinfiles` |
| Sertakan dari kursor ke akhir kata dalam pemilihan | `Option-Shift-Right` | `selectToWordEndRight` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Option-Shift-Left` | `selectToWordStartLeft` |
| Menampilkan kotak dialog Daftar Proses | `Command-Option-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Command-Z` | `undo` |

# Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Option-Control-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Option-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Command-Control-W` | `closepane` |
| Menutup tab saat ini | `Option-W` | `closetab` |
| Pergi satu panel ke bawah | `Control-Command-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Control-Command-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Control-Command-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Control-Command-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Command-Shift-[ \| Command-Option-Left` | `gototableft` |
| Pergi satu tab ke kanan | `Command-Shift-] \| Command-Option-Right` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Command-Option-Shift-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada | `Command-Option-Shift-Left` | `movetableft` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| di paling kiri, membuat tab terpisah di sana | | |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Command-Option-Shift-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Command-Option-Shift-Up` | `movetabup` |
| Pergi ke tab berikutnya | `Control-Tab` | `nexttab` |
| Pergi ke panel sebelumnya | `Option-Shift-Esc` | `previouspane` |
| Pergi ke tab sebelumnya | `Control-Shift-Tab` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Command-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Command-E` | `revealtab` |
| Pergi ke tab kesepuluh | `Command-0` | `tab0` |
| Pergi ke tab pertama | `Command-1` | `tab1` |
| Pergi ke tab kedua | `Command-2` | `tab2` |
| Pergi ke tab ketiga | `Command-3` | `tab3` |
| Pergi ke tab keempat | `Command-4` | `tab4` |
| Pergi ke tab kelima | `Command-5` | `tab5` |
| Pergi ke tab keenam | `Command-6` | `tab6` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke tab ketujuh | `Command-7` | `tab7` |
| Pergi ke tab kedelapan | `Command-8` | `tab8` |
| Pergi ke tab kesembilan | `Command` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Command-E\|Command-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Command-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Command-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Command-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Command-Shift-R` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `Control-` ` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Command-K Command-B` | `toggletree` |

# Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Control-Shift-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Control-Option-Shi ft-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Control-Shift-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Control-Option-Shi ft-Down` | `addCursorBelowSkip Current` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Control-Option-A` | `alignCursors` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Backspace satu spasi | `Control-Backspace \| Shift-Backspace \| Backspace` | `backspace` |
| Meng-indent seleksi satu tab | `Control-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Control-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Command-Z \| Command-Shift-Z \| Command-Y` | `cancelBrowserUndoInAce` |
| Memusatkan seleksi | `Command-K Command-C \| Control-L` | `centerselection` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Command-Option-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Command-Option-Up` | `copylinesup` |
| Menghapus satu spasi | `Delete \| Control-Delete \| Shift-Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Command-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Command-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Control-Shift-M` | `expandToMatching` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Command-Option-L \| Command-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat | `Control-Command-Option-0` | `foldall` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Command-K Command-1` | `foldOther` |
| Turun satu baris | `Down \| Control-N` | `golinedown` |
| Naik satu baris | `Up \| Control-P` | `golineup` |
| Pergi ke akhir file | `Command-End \| Command-Down` | `gotoend` |
| Pergi ke kiri satu spasi | `Left \| Control-B` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Command-Right \| End \| Control-E` | `gotolineend` |
| Pergi ke awal baris saat ini | `Command-Left \| Home \| Control-A` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `Control-F6` | `goToNextError` |
| Turun satu halaman | `Page Down \| Control-V` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Control-Shift-F6` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right \| Control-F` | `gotoright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke awal file | `Command-Home \| Command-Up` | `gotostart` |
| Pergi satu kata ke kiri | `Option-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Option-Right` | `gotowordright` |
| Indent seleksi satu tab | `Tab` | `indent` |
| Gabungkan baris yang dipilih ke dalam satu baris | `Command-J` | `joinlines` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Control-M` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Command-= \| Command-+` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Option-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Control-Command-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Control-Command-Up` | `movelinesup` |
| Outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Turun satu halaman | `Option-Page Down` | `pagedown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Naik satu halaman | `Option-Page Up` | `pageup` |
| Menghapus isi dari baris saat ini | `Control-Shift-K` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Control-K` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Command-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Option-Backspace \| Control-Option-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Option-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Control-Shift-Q` | `replaymacro` |
| Memilih semua konten yang dapat dipilih | `Command-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down \| Control-Shift-N` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left \| Control-Shift-B` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Control-Option-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Control-Option-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Control-Option-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Control-Option-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Control-G` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Control-Shift-G` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Command-Shift-End \| Command-Shift-Down` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Command-Shift-Right \| Shift-End \| Control-Shift-E` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Command-Shift-Left \| Control-Shift-A` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Control-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Command-Shift-Home \| Command-Shift-Up` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up \| Control-Shift-P` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Option-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Option-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Command-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Command--` | `smallerfont` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `F5` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Command-Shift-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Control-O` | `splitline` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Command-Option-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Command-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `Command-Option-[` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Option-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Control-Q` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Control-W` | `toggleWordWrap` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengubah seleksi ke semua huruf kecil | `Command-K Command-L` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Command-K Command-U` | `touppercase` |
| Menranspos seleksi | `Control-T` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Command-Option-]` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Command-K Command-0 \| Command-K Command-J` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Command-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Control-Option-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Command-.` | `emmet_select_next_item` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Command-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Control-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Option-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Option-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `F7 \| Command-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8 \| Command-\` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Command-Shift-B` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11 \| Command-;` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11 \| Command-Shift-'` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10 \| Command-'` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Control-Break` | `stopbuild` |

# Referensi Keybindings Default Windows/Linux untuk IDE AWS Cloud9

Langkah ini menunjukkan bagaimana memilih Referenc Keybindings Default Windows/Linux untuk.. AWS Cloud9

1. Pada bilah menu, AWS Cloud9, Preferensi.
2. Pada tab Preferensi, memilih Keybinding.
3. Untuk Mode Keyboard pilih Default.
4. Untuk Sistem OperasiPilih, pilih Windows / Linux.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard default untuk sistem operasi Windows/Linux di AWS Cloud9 Integrated Development Environment (IDE).

- Umum
- Tab

- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Ctrl-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Ctrl-Space \| Alt-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Ctrl-Shift-Space \| Alt-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Ctrl-C` | `copy` |
| Memotong seleksi ke clipboard | `Ctrl-X` | `cut` |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Ctrl-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Ctrl-Alt-K` | `findall` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-K` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-Shift-K` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Ctrl-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Ctrl-G` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Ctrl-Shift-U` | `lambdaUploadFunction` |
| Membuat file baru | `Alt-N` | `newfile` |
| Menampilkan tab Preferensi | `Ctrl-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Alt-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Ctrl-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Ctrl-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Ctrl-Shift-Z \|Ctrl-Y` | `redo` |
| Merefresh panel pratinjau | `Ctrl-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Ctrl-Alt-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Alt-Shift-F \|Ctrl-H` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Ctrl-Enter` | `rerunInitScript` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memulai ulang lingkungan | `Ctrl-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Ctrl-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Alt-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Ctrl-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Ctrl-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Ctrl-Shift-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Ctrl-Alt-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Ctrl-Z` | `undo` |

## Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Ctrl-Alt-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Alt-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Ctrl-W` | `closepane` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup tab saat ini | `Alt-W` | `closetab` |
| Pergi satu panel ke bawah | `Ctrl-Meta-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Ctrl-Meta-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Ctrl-Meta-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Ctrl-Meta-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Ctrl-[` | `gototableft` |
| Pergi satu tab ke kanan | `Ctrl-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Ctrl-Meta-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Ctrl-Meta-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Ctrl-Meta-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Ctrl-Meta-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Ctrl-` ` | `nextpane` |
| Pergi ke tab berikutnya | `Ctrl-Tab \| Alt-` ` | `nexttab` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke panel sebelumnya | `Ctrl-Shift-`` | `previouspane` |
| Pergi ke tab sebelumnya | `Ctrl-Shift-Tab \| Alt-Shift-`` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Alt-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Ctrl-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Ctrl-0` | `tab0` |
| Pergi ke tab pertama | `Ctrl-1` | `tab1` |
| Pergi ke tab kedua | `Ctrl-2` | `tab2` |
| Pergi ke tab ketiga | `Ctrl-3` | `tab3` |
| Pergi ke tab keempat | `Ctrl-4` | `tab4` |
| Pergi ke tab kelima | `Ctrl-5` | `tab5` |
| Pergi ke tab keenam | `Ctrl-6` | `tab6` |
| Pergi ke tab ketujuh | `Ctrl-7` | `tab7` |
| Pergi ke tab kedelapan | `Ctrl-8` | `tab8` |
| Pergi ke tab kesembilan | `Ctrl-9` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Ctrl-E\|Ctrl-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Ctrl-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Ctrl-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Ctrl-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Ctrl-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `F6` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Ctrl-I` | `toggletree` |

## Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk | `Ctrl-Alt-Up` | `addCursorAbove` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| an, menambahkan kursor lain di atas kursor itu | | |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Ctrl-Alt-Shift-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Ctrl-Alt-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Ctrl-Alt-Shift-Down` | `addCursorBelowSkip Current` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Ctrl-Alt-A` | `alignCursors` |
| Backspace satu spasi | `Shift-Backspace \| Backspace` | `backspace` |
| Meng-indent seleksi satu tab | `Ctrl-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Ctrl-[` | `blockoutdent` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Ctrl-Z \| Ctrl-Shift-Z \| Ctrl-Y` | `cancelBrowserUndoInAce` |
| Memusatkan seleksi | `Ctrl-L` | `centerselection` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Alt-Shift-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Alt-Shift-Up` | `copylinesup` |
| Memotong seleksi, atau jika tidak ada seleksi, hapus satu spasi | `Shift-Delete` | `cut_or_delete` |
| Menghapus satu spasi | `Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Ctrl-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Ctrl-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Ctrl-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Alt-L \| Ctrl-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat | `Ctrl-Command-Option-0` | `foldall` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Alt-0` | `foldOther` |
| Turun satu baris | `Down` | `golinedown` |
| Naik satu baris | `Up` | `golineup` |
| Pergi ke akhir file | `Ctrl-End` | `gotoend` |
| Pergi ke kiri satu spasi | `Left` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Alt-Right \| End` | `gotolineend` |
| Pergi ke awal baris saat ini | `Alt-Left \| Home` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `Alt-E` | `goToNextError` |
| Turun satu halaman | `Page Down` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Alt-Shift-E` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right` | `gotoright` |
| Pergi ke awal file | `Ctrl-Home` | `gotostart` |
| Pergi satu kata ke kiri | `Ctrl-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Ctrl-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Ctrl-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Ctrl-+ \| Ctrl-=` | `largerfont` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Alt-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Alt-Up` | `movelinesup` |
| Meng-outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Turun satu halaman | `Option-Page Down` | `pagedown` |
| Naik satu halaman | `Option-Page Up` | `pageup` |
| Menghapus isi dari baris saat ini | `Ctrl-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Alt-Delete` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Alt-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Ctrl-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Ctrl-Delete` | `removewordright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Ctrl-Shift-E` | `replaymacro` |
| Menggulir file saat ini ke bawah satu baris | `Ctrl-Down` | `scrolldown` |
| Menggulir file saat ini ke atas satu baris | `Ctrl-Up` | `scrollup` |
| Memilih semua konten yang dapat dipilih | `Ctrl-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Ctrl-Alt-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Ctrl-Alt-Left` | `selectMoreBefore` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Ctrl-Alt-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Ctrl-Alt-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Alt-K` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Alt-Shift-K` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Ctrl-Shift-End` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Alt-Shift-Right` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Alt-Shift-Left` | `selecttolinestart` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Ctrl-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Ctrl-Shift-Home` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Ctrl-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Ctrl-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Ctrl--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Ctrl-Alt-S` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Ctrl-Alt-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Ctrl-O` | `splitline` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Ctrl-Shift-/` | `toggleBlockComment` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Ctrl-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Alt-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Ctrl-Alt-E` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Ctrl-Q` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Ctrl-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Ctrl-U` | `touppercase` |
| Menranspos seleksi | `Alt-X` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Alt-Shift-L \| Ctrl-Shift-F1` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Alt-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Ctrl-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Ctrl-Alt-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Ctrl-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Ctrl-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Ctrl-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Alt-T` | `openterminal` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Beralih antara editor dan tab Terminal | `Alt-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Ctrl-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Alt-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Ctrl-Shift-C` | `stopbuild` |

# Referensi Keybindings Vim Windows/Linux untuk IDE AWS Cloud9

Langkah ini menunjukkan bagaimana memilih Referensi Keybindings Vim Windows/Linux untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, pilih Keybinding.
3. Untuk Mode Keyboard pilih Vim.
4. Untuk Sistem OperasiPilih, pilih Windows / Linux.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Vim untuk sistem operasi Windows/Linux di IDE: AWS Cloud9


- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug


## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Ctrl-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Ctrl-Space` \| `Alt-Space` | `complete` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Kode selesai, dan kemudian menimpa | `Ctrl-Shift-Space \| Alt-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Ctrl-C` | `copy` |
| Memotong seleksi ke clipboard | `Ctrl-X` | `cut` |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Ctrl-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Ctrl-Alt-K` | `findall` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-K` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-Shift-K` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memformat ulang kode yang dipilih JavaScript | `Ctrl-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Ctrl-G` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Ctrl-Shift-U` | `lambdaUploadFunction` |
| Membuat file baru | `Alt-N` | `newfile` |
| Menampilkan tab Preferensi | `Ctrl-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Alt-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Ctrl-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Ctrl-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Ctrl-Shift-Z \|Ctrl-Y` | `redo` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Merefresh panel pratinjau | `Ctrl-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Ctrl-Alt-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Alt-Shift-F \| Ctrl-H` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Ctrl-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Ctrl-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Ctrl-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Alt-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Ctrl-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Ctrl-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Ctrl-Shift-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Ctrl-Alt-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Ctrl-Z` | `undo` |

# Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Ctrl-Alt-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Alt-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Ctrl-W` | `closepane` |
| Menutup tab saat ini | `Alt-W` | `closetab` |
| Pergi satu panel ke bawah | `Ctrl-Meta-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Ctrl-Meta-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Ctrl-Meta-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Ctrl-Meta-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Ctrl-[` | `gototableft` |
| Pergi satu tab ke kanan | `Ctrl-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Ctrl-Meta-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Ctrl-Meta-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah | `Ctrl-Meta-Right` | `movetabright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| berada di paling kanan, membuat tab terpisah di sana | | |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Ctrl-Meta-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Ctrl-` ` | `nextpane` |
| Pergi ke tab berikutnya | `Ctrl-Tab \| Alt-` ` | `nexttab` |
| Pergi ke panel sebelumnya | `Ctrl-Shift-` ` | `previouspane` |
| Pergi ke tab sebelumnya | `Ctrl-Shift-Tab \| Alt-Shift-` ` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Alt-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Ctrl-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Ctrl-0` | `tab0` |
| Pergi ke tab pertama | `Ctrl-1` | `tab1` |
| Pergi ke tab kedua | `Ctrl-2` | `tab2` |
| Pergi ke tab ketiga | `Ctrl-3` | `tab3` |
| Pergi ke tab keempat | `Ctrl-4` | `tab4` |
| Pergi ke tab kelima | `Ctrl-5` | `tab5` |
| Pergi ke tab keenam | `Ctrl-6` | `tab6` |
| Pergi ke tab ketujuh | `Ctrl-7` | `tab7` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke tab kedelapan | `Ctrl-8` | `tab8` |
| Pergi ke tab kesembilan | `Ctrl-9` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Ctrl-E\|Ctrl-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Ctrl-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Ctrl-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Ctrl-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Ctrl-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `F6` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Ctrl-I` | `toggletree` |

# Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Ctrl-Alt-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Ctrl-Alt-Shift-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Ctrl-Alt-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Ctrl-Alt-Shift-Down` | `addCursorBelowSkip Current` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Ctrl-Alt-A` | `alignCursors` |
| Backspace satu spasi | `Shift-Backspace \| Backspace` | `backspace` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Meng-indent seleksi satu tab | `Ctrl-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Ctrl-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Ctrl-Z \| Ctrl-Shift-Z \| Ctrl-Y` | `cancelBrowserUndoInAce` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Alt-Shift-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Alt-Shift-Up` | `copylinesup` |
| Memotong seleksi. Jika tidak ada seleksi, hapus satu spasi | `Shift-Delete` | `cut_or_delete` |
| Menghapus satu spasi | `Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Ctrl-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Ctrl-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Ctrl-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Alt-L \| Ctrl-F1` | `fold` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Alt-0` | `foldOther` |
| Turun satu baris | `Down` | `golinedown` |
| Naik satu baris | `Up` | `golineup` |
| Pergi ke akhir file | `Ctrl-End` | `gotoend` |
| Pergi ke kiri satu spasi | `Left` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Alt-Right \| End` | `gotolineend` |
| Pergi ke awal baris saat ini | `Alt-Left \| Home` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `Alt-E` | `goToNextError` |
| Turun satu halaman | `Page Down` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Alt-Shift-E` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right` | `gotoright` |
| Pergi ke awal file | `Ctrl-Home` | `gotostart` |
| Pergi satu kata ke kiri | `Ctrl-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Ctrl-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Ctrl-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Ctrl-+ \| Ctrl-=` | `largerfont` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Alt-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Alt-Up` | `movelinesup` |
| Meng-outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Menghapus isi dari baris saat ini | `Ctrl-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Alt-Delete` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Alt-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Ctrl-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Ctrl-Delete` | `removewordright` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Ctrl-Shift-E` | `replaymacro` |
| Menggulir file saat ini ke bawah satu baris | `Ctrl-Down` | `scrolldown` |
| Menggulir file saat ini ke atas satu baris | `Ctrl-Up` | `scrollup` |
| Memilih semua konten yang dapat dipilih | `Ctrl-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Ctrl-Alt-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Ctrl-Alt-Left` | `selectMoreBefore` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Ctrl-Alt-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Ctrl-Alt-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Alt-K` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Alt-Shift-K` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Ctrl-Shift-End` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Alt-Shift-Right` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Alt-Shift-Left` | `selecttolinestart` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Ctrl-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Ctrl-Shift-Home` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Ctrl-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Ctrl-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Ctrl--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Ctrl-Alt-S` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Ctrl-Alt-L` | `splitIntoLines` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Ctrl-Shift-/` | `toggleBlockComment` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Ctrl-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Alt-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Ctrl-Alt-E` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Ctrl-Q` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Ctrl-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Ctrl-U` | `touppercase` |
| Menranspos seleksi | `Alt-X` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Alt-Shift-L \| Ctrl-Shift-F1` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Alt-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Ctrl-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Ctrl-Alt-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Ctrl-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Ctrl-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Ctrl-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Alt-T` | `openterminal` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Beralih antara editor dan tab Terminal | `Alt-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Ctrl-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Alt-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Ctrl-Shift-C` | `stopbuild` |

# Referensi Keybindings Windows/Linux Emacs untuk IDE AWS Cloud9

Langkah ini menunjukkan bagaimana memilih Referensi Keybindings Windows /Linux Emacs untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, memilih Keybinding.
3. Untuk Mode Keyboard memilih Emacs.
4. Untuk Sistem OperasiPilih, memilih Windows / Linux.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Emacs untuk sistem operasi Windows/Linux di IDE: AWS Cloud9

- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

## Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Ctrl-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan menu konteks penyelesaian kode | `Ctrl-Space \| Alt-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Ctrl-Shift-Space \| Alt-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Ctrl-C` | `copy` |
| Memotong seleksi ke clipboard | `Ctrl-X` | `cut` |
| Memperluas kode, jika dapat berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Ctrl-F` | `find` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Ctrl-Alt-K` | `findall` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-K` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Ctrl-Shift-K` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Ctrl-Shift-B` | `formatcode` |
| Menampilkan kotak pergi ke baris | `Ctrl-G` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F3` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba mengunggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Ctrl-Shift-U` | `lambdaUploadFunction` |
| Membuat file baru | `Alt-N` | `newfile` |
| Menampilkan tab Preferensi | `Ctrl-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Alt-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Ctrl-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Ctrl-F3` | `quickfix` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengulangi tindakan terakhir | `Ctrl-Shift-Z \| Ctrl-Y` | `redo` |
| Merefresh panel pratinjau | `Ctrl-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Ctrl-Alt-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ekspresi ganti dengan | `Alt-Shift-F \| Ctrl-H` | `replace` |
| Menjalankan kembali skrip inisialisasi Anda | `Ctrl-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Ctrl-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Ctrl-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Alt-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Ctrl-S` | `save` |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Ctrl-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Ctrl-Shift-F` | `searchinfiles` |
| Menampilkan kotak dialog Daftar Proses | `Ctrl-Alt-P` | `showprocesslist` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengurungkan tindakan terakhir | `Ctrl-Z` | `undo` |

## Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Ctrl-Alt-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Alt-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Ctrl-W` | `closepane` |
| Menutup tab saat ini | `Alt-W` | `closetab` |
| Pergi satu panel ke bawah | `Ctrl-Meta-Down` | `gotopanedown` |
| Pergi satu panel ke kiri | `Ctrl-Meta-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Ctrl-Meta-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Ctrl-Meta-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Ctrl-[` | `gototableft` |
| Pergi satu tab ke kanan | `Ctrl-]` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Ctrl-Meta-Down` | `movetabdown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Ctrl-Meta-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Ctrl-Meta-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Ctrl-Meta-Up` | `movetabup` |
| Pergi ke panel berikutnya | `Ctrl-` ` | `nextpane` |
| Pergi ke tab berikutnya | `Ctrl-Tab \| Alt-` ` | `nexttab` |
| Pergi ke panel sebelumnya | `Ctrl-Shift-` ` | `previouspane` |
| Pergi ke tab sebelumnya | `Ctrl-Shift-Tab \| Alt-Shift-` ` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |
| Membuka tab terakhir lagi | `Alt-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Ctrl-Shift-L` | `revealtab` |
| Pergi ke tab kesepuluh | `Ctrl-0` | `tab0` |
| Pergi ke tab pertama | `Ctrl-1` | `tab1` |
| Pergi ke tab kedua | `Ctrl-2` | `tab2` |
| Pergi ke tab ketiga | `Ctrl-3` | `tab3` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi ke tab keempat | `Ctrl-4` | `tab4` |
| Pergi ke tab kelima | `Ctrl-5` | `tab5` |
| Pergi ke tab keenam | `Ctrl-6` | `tab6` |
| Pergi ke tab ketujuh | `Ctrl-7` | `tab7` |
| Pergi ke tab kedelapan | `Ctrl-8` | `tab8` |
| Pergi ke tab kesembilan | `Ctrl-9` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Ctrl-E\|Ctrl-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Ctrl-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Ctrl-O` | `gotofile` |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Ctrl-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Ctrl-Shift-E` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `F6` | `toggleconsole` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Ctrl-I` | `toggletree` |

## Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Ctrl-Alt-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Ctrl-Alt-Shift-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | `Ctrl-Alt-Down` | `addCursorBelow` |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Ctrl-Alt-Shift-Down` | `addCursorBelowSkip Current` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Ctrl-Alt-A` | `alignCursors` |
| Backspace satu spasi | `Shift-Backspace \| Backspace` | `backspace` |
| Meng-indent seleksi satu tab | `Ctrl-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Ctrl-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Ctrl-Z \| Ctrl-Shift-Z \| Ctrl-Y` | `cancelBrowserUndoInAce` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Alt-Shift-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Alt-Shift-Up` | `copylinesup` |
| Memotong seleksi, atau jika tidak ada seleksi, hapus satu spasi | `Shift-Delete` | `cut_or_delete` |
| Menghapus satu spasi | `Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Ctrl-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Ctrl-Shift-L` | `expandtoline` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Ctrl-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Alt-L \| Ctrl-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Alt-0` | `foldOther` |
| Turun satu baris | `Down` | `golinedown` |
| Naik satu baris | `Up` | `golineup` |
| Pergi ke akhir file | `Ctrl-End` | `gotoend` |
| Pergi ke kiri satu spasi | `Left` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Alt-Right \| End` | `gotolineend` |
| Pergi ke awal baris saat ini | `Alt-Left \| Home` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `Alt-E` | `goToNextError` |
| Turun satu halaman | `Page Down` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Alt-Shift-E` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right` | `gotoright` |
| Pergi ke awal file | `Ctrl-Home` | `gotostart` |
| Pergi satu kata ke kiri | `Ctrl-Left` | `gotowordleft` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi satu kata ke kanan | `Ctrl-Right` | `gotowordright` |
| Meng-indent seleksi satu tab | `Tab` | `indent` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Ctrl-P` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Ctrl-+ \| Ctrl-=` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Ctrl-Shift-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Alt-Down` | `movelinesdown` |
| Memindahkan seleksi ke atas satu baris | `Alt-Up` | `movelinesup` |
| Outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Menghapus isi dari baris saat ini | `Ctrl-D` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Alt-Delete` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Alt-Backspace` | `removetolinestart` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menghapus kata di sebelah kiri kursor | `Ctrl-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Ctrl-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Ctrl-Shift-E` | `replaymacro` |
| Menggulir file saat ini ke bawah satu baris | `Ctrl-Down` | `scrolldown` |
| Menggulir file saat ini ke atas satu baris | `Ctrl-Up` | `scrollup` |
| Memilih semua konten yang dapat dipilih | `Ctrl-A` | `selectall` |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Ctrl-Alt-Right` | `selectMoreAfter` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Ctrl-Alt-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Ctrl-Alt-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Ctrl-Alt-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Alt-K` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Alt-Shift-K` | `selectOrFindPrevious` |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Ctrl-Shift-End` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Alt-Shift-Right` | `selecttolineend` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Alt-Shift-Left` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Ctrl-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Ctrl-Shift-Home` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-Shift-Left` | `selectwordleft` |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Ctrl-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Ctrl-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Ctrl--` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `Ctrl-Alt-S` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Ctrl-Alt-L` | `splitIntoLines` |
| Memindahkan isi kursor ke akhir baris, ke barisnya sendiri | `Ctrl-O` | `splitline` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Ctrl-Shift-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Ctrl-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `F2` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Alt-F2` | `toggleParentFoldWidget` |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Ctrl-Alt-E` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Ctrl-Q` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Ctrl-Shift-U` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Ctrl-U` | `touppercase` |
| Menranspos seleksi | `Alt-X` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Alt-Shift-L \| Ctrl-Shift-F1` | `unfold` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Alt-Shift-0` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Ctrl-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, XML, atau CSS, tergantung pada sintaks file saat ini | `Ctrl-Alt-E` | `emmet_expand_abbreviation` |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Ctrl-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Ctrl-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Ctrl-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Alt-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Alt-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `Ctrl-B` | `build` |
| Melanjutkan proses terjeda saat ini | `F8` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Alt-F5` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Berhenti membangun file saat ini | `Ctrl-Shift-C` | `stopbuild` |

# Referensi Keybindings Sublime Windows/Linux untuk IDE AWS Cloud9

Langkah ini menunjukkan bagaimana memilih Referensi Keybindings Sublime Windows/Linux untuk. AWS Cloud9

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, pilih Keybinding.
3. Untuk Mode Keyboard pilih Sublime.
4. Untuk Sistem OperasiPilih, pilih Windows / Linux.

Untuk informasi selengkapnya tentang ikatan kunci, lihat Bekerja dengan Pengikatan Kunci.

Berikut ini adalah daftar keybinding mode keyboard Sublime untuk sistem operasi Windows/Linux di IDE: AWS Cloud9

- Umum
- Tab
- Panel
- Editor Kode
- emmet
- Terminal
- Jalankan dan Debug

# Umum

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan seleksi sebagai ekspresi tontonan | `Ctrl-Shift-C` | `addwatchfromselection` |
| Menghapus seleksi potongan dari clipboard | `Esc` | `clearcut` |
| Menampilkan menu konteks penyelesaian kode | `Ctrl-Space` | `complete` |
| Kode selesai, dan kemudian menimpa | `Ctrl-Shift-Space \| Alt-Shift-Space` | `completeoverwrite` |
| Menyalin seleksi ke clipboard | `Ctrl-C` | `copy` |
| Memotong pemilihan ke clipboard | `Ctrl-X` | `cut` |
| Menghapus dari kursor ke awal baris | `Ctrl-Shift-Backspace \| Ctrl-K Ctrl-Backspace` | `delete_to_hard_bol` |
| Menghapus dari kursor ke akhir baris | `Ctrl-Shift-Delete \| Ctrl-K Ctrl-K` | `delete_to_hard_eol` |
| Memperluas kode, jika berlaku | `Tab` | `expandSnippet` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini | `Ctrl-F` | `find` |
| Menyorot semua kecocokan untuk seleksi | `Alt-F3` | `find_all_under` |
| Menyorot kecocokan berikutnya untuk seleksi | `Ctrl-F3` | `find_under` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyorot di sekitar kursor dan semua kecocokan untuk disorot | `Ctrl-D` | `find_under_expand` |
| Menyorot di sekitar kursor dan outline semua kecocokan untuk disorot | `Ctrl-K Ctrl-D` | `find_under_expand_skip` |
| Menyorot kecocokan sebelumnya untuk seleksi | `Ctrl-Shift-F3` | `find_under_prev` |
| Memilih semua temuan kecocokan di dokumen saat ini | `Ctrl-Alt-K` | `findAll` |
| Pergi ke kecocokan berikutnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `F3` | `findnext` |
| Pergi ke kecocokan sebelumnya dalam dokumen saat ini untuk menemukan kueri yang Anda masukkan terakhir | `Shift-F3` | `findprevious` |
| Menampilkan semua referensi yang diketahui ke simbol pada titik penyisipan dalam file aktif di editor | `Shift-F3` | `findReferences` |
| Membuka jendela Lingkungan, lalu membuat daftar file aktif | `Shift-Esc` | `focusTree` |
| Memformat ulang kode yang dipilih JavaScript | `Ctrl-Alt-F` | `formatcode` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan kotak pergi ke baris | `Ctrl-G` | `gotoline` |
| Menyembunyikan bilah temukan dan ganti, jika ditampilkan | `Esc` | `hidesearchreplace` |
| Membuka definisi variabel atau fungsi di kursor | `F12` | `jumptodef` |
| Jika fungsi Lambda lokal dipilih di bagian Lambda pada jendela Sumber Daya AWS , coba unggah fungsi tersebut ke Lambda sebagai fungsi jarak jauh | `Ctrl-Shift-U` | `lambdaUploadFunction` |
| Pergi ke akhir kata saat ini | `Ctrl-Right` | `moveToWordEndRight` |
| Pergi ke awal kata saat ini | `Ctrl-Left` | `moveToWordStartLeft` |
| Membuat file baru | `Alt-N` | `newfile` |
| Menampilkan tab Preferensi | `Ctrl-,` | `openpreferences` |
| Membuka tab Terminal, dan kemudian beralih ke folder induk dari file yang dipilih dalam daftar file | `Alt-L` | `opentermhere` |
| Menempelkan isi clipboard saat ini pada kursor | `Ctrl-V` | `paste` |
| Menampilkan saran untuk memperbaiki kesalahan | `Ctrl-F3` | `quickfix` |
| Mengulangi tindakan terakhir | `Ctrl-Shift-Z \|Ctrl-Y` | `redo` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Merefresh panel pratinjau | `Ctrl-Enter` | `reloadpreview` |
| Memulai memfaktor ulang (refactor) ganti nama untuk seleksi | `Ctrl-Alt-R` | `renameVar` |
| Menampilkan bilah temukan dan ganti untuk dokumen saat ini, dengan fokus pada ganti dengan ekspresi | `Ctrl-H` | `replace` |
| Ganti semua menemukan ekspresi cocok dengan ganti dengan ekspresi di find dan replace bar | `Ctrl-Alt-Enter` | `replaceall` |
| Ganti berikutnya menemukan ekspresi cocok dengan ganti dengan ekspresi di find dan replace bar | `Ctrl-Shift-H` | `replacenext` |
| Menjalankan kembali skrip inisialisasi Anda | `Ctrl-Enter` | `rerunInitScript` |
| Memulai ulang lingkungan | `Ctrl-R` | `restartc9` |
| Menyetel ulang file saat ini ke versi terakhir yang disimpan | `Ctrl-Shift-Q` | `reverttosaved` |
| Menyetel ulang setiap file yang terbuka ke versi yang disimpan | `Alt-Shift-Q` | `reverttosavedall` |
| Menyimpan file saat ini ke disk | `Ctrl-S` | `save` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyimpan file saat ini ke disk dengan nama file yang berbeda | `Ctrl-Shift-S` | `saveas` |
| Menampilkan bilah temukan dan ganti untuk banyak file | `Ctrl-Shift-F` | `searchinfiles` |
| Sertakan dari kursor ke akhir kata dalam pemilihan | `Ctrl-Shift-Right` | `selectToWordEndRight` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-Shift-Left` | `selectToWordStartLeft` |
| Menampilkan kotak dialog Daftar Proses | `Ctrl-Alt-P` | `showprocesslist` |
| Mengurungkan tindakan terakhir | `Ctrl-Z` | `undo` |

## Tab

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menutup semua tab yang terbuka di panel saat ini, kecuali tab saat ini | `Ctrl-Alt-W` | `closeallbutme` |
| Menutup semua tab yang terbuka di semua panel | `Alt-Shift-W` | `closealltabs` |
| Menutup panel saat ini | `Ctrl-W` | `closepane` |
| Menutup tab saat ini | `Alt-W` | `closetab` |
| Pergi satu panel ke bawah | `Ctrl-Meta-Down` | `gotopanedown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Pergi satu panel ke kiri | `Ctrl-Meta-Left` | `gotopaneleft` |
| Pergi satu panel ke kanan | `Ctrl-Meta-Right` | `gotopaneright` |
| Pergi satu panel ke atas | `Ctrl-Meta-Up` | `gottopaneup` |
| Pergi satu tab ke kiri | `Ctrl-Page Up` | `gototableft` |
| Pergi satu tab ke kanan | `Ctrl-Page Down` | `gototabright` |
| Memindahkan tab saat ini ke bawah satu panel, atau jika tab sudah berada di paling bawah, membuat tab terpisah di sana | `Ctrl-Meta-Down` | `movetabdown` |
| Memindahkan tab saat ini ke kiri, atau jika tab sudah berada di paling kiri, membuat tab terpisah di sana | `Ctrl-Meta-Left` | `movetableft` |
| Memindahkan tab saat ini ke kanan, atau jika tab sudah berada di paling kanan, membuat tab terpisah di sana | `Ctrl-Meta-Right` | `movetabright` |
| Memindahkan tab saat ini ke atas satu panel, atau jika tab sudah berada di paling atas, membuat tab terpisah di sana | `Ctrl-Meta-Up` | `movetabup` |
| Pergi ke tab berikutnya | `Ctrl-Tab` | `nexttab` |
| Pergi ke panel sebelumnya | `Ctrl-Shift-`` | `previouspane` |
| Pergi ke tab sebelumnya | `Ctrl-Shift-Tab` | `previoustab` |
| Kembali ke tab terakhir | `Esc` | `refocusTab` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab terakhir lagi | `Ctrl-Shift-T` | `reopenLastTab` |
| Menampilkan tab saat ini di pohon file | `Ctrl-E` | `revealtab` |
| Pergi ke tab kesepuluh | `Ctrl-0` | `tab0` |
| Pergi ke tab pertama | `Ctrl-1` | `tab1` |
| Pergi ke tab kedua | `Ctrl-2` | `tab2` |
| Pergi ke tab ketiga | `Ctrl-3` | `tab3` |
| Pergi ke tab keempat | `Ctrl-4` | `tab4` |
| Pergi ke tab kelima | `Ctrl-5` | `tab5` |
| Pergi ke tab keenam | `Ctrl-6` | `tab6` |
| Pergi ke tab ketujuh | `Ctrl-7` | `tab7` |
| Pergi ke tab kedelapan | `Ctrl-8` | `tab8` |
| Pergi ke tab kesembilan | `Ctrl-9` | `tab9` |

## Panel

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Apa Pun | `Ctrl-E\|Ctrl-P` | `gotoanything` |
| Menampilkan jendela Pergi di mode Pergi ke Perintah | `Ctrl-.\|F1` | `gotocommand` |
| Menampilkan jendela Pergi di mode Pergi ke File. | `Ctrl-O` | `gotofile` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menampilkan jendela Pergi di mode Pergi ke Simbol. | `Ctrl-Shift-O` | `gotosymbol` |
| Menampilkan jendela Garis Besar | `Ctrl-R\|Ctrl-Shift-R` | `outline` |
| Menampilkan jendela Konsol jika tersembunyi, atau menyembunyikan jika ditampilkan | `Ctrl-` ` | `toggleconsole` |
| Menampilkan jendela Lingkungan jika tersembun yi, atau menyembunyikan jika ditampilkan | `Ctrl-K Ctrl-B` | `toggletree` |

## Editor Kode

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menambahkan kursor satu baris di atas kursor aktif, atau jika kursor sudah ditambahk an, menambahkan kursor lain di atas kursor itu | `Ctrl-Alt-Up` | `addCursorAbove` |
| Menambahkan kursor kedua satu baris di atas kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahk an kursor kedua ke atas satu baris | `Ctrl-Alt-Shift-Up` | `addCursorAboveSkip Current` |
| Menambahkan kursor satu baris di bawah kursor aktif, | `Ctrl-Alt-Down` | `addCursorBelow` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| atau jika kursor sudah ditambahkan, menambahkan kursor lain di bawah kursor itu | | |
| Menambahkan kursor kedua satu baris di bawah kursor aktif, atau jika kursor kedua sudah ditambahkan, memindahkan kursor kedua ke bawah satu baris | `Ctrl-Alt-Shift-Down` | `addCursorBelowSkipCurrent` |
| Memindahkan semua kursor ke spasi yang sama dengan kursor aktif di setiap barisnya, jika tidak sejajar | `Ctrl-Alt-A` | `alignCursors` |
| Backspace satu spasi | `Shift-Backspace \| Backspace` | `backspace` |
| Meng-indent seleksi satu tab | `Ctrl-]` | `blockindent` |
| Meng-outdent seleksi satu tab | `Ctrl-[` | `blockoutdent` |
| Mengontrol apakah fokus dapat dialihkan dari editor ke tempat lain di IDE | `Ctrl-Z \| Ctrl-Shift-Z \| Ctrl-Y` | `cancelBrowserUndoInAce` |
| Memusatkan seleksi | `Ctrl-K Ctrl-C` | `centerselection` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke bawah | `Alt-Shift-Down` | `copylinesdown` |
| Menyalin isi baris, dan menempelkan isi yang disalin satu baris ke atas | `Alt-Shift-Up` | `copylinesup` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memotong seleksi, atau jika tidak ada seleksi, hapus satu spasi | `Shift-Delete` | `cut_or_delete` |
| Menghapus satu spasi | `Delete` | `del` |
| Menyalin isi seleksi, dan menempelkan isi yang disalin segera setelah seleksi | `Ctrl-Shift-D` | `duplicateSelection` |
| Menyertakan isi baris saat ini dalam seleksi | `Ctrl-Shift-L` | `expandtoline` |
| Menyertakan hingga simbol pencocokan berikutnya dalam seleksi | `Ctrl-Shift-M` | `expandToMatching` |
| Melipat kode yang dipilih; jika unit terlipat yang dipilih, membuka lipatannya | `Alt-L \| Ctrl-F1` | `fold` |
| Melipat semua elemen yang mungkin dapat dilipat, kecuali untuk cakupan seleksi saat ini | `Ctrl-K Ctrl-1` | `foldOther` |
| Turun satu baris | `Down` | `golinedown` |
| Naik satu baris | `Up` | `golineup` |
| Pergi ke akhir file | `Ctrl-End` | `gotoend` |
| Pergi ke kiri satu spasi | `Left` | `gotoleft` |
| Pergi ke akhir baris saat ini | `Alt-Right \| End` | `gotolineend` |
| Pergi ke awal baris saat ini | `Alt-Left \| Home` | `gotolinestart` |
| Pergi ke kesalahan berikutnya | `Ctrl-F6` | `goToNextError` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Turun satu halaman | `Page Down` | `gotopagedown` |
| Naik satu halaman | `Page Up` | `gotopageup` |
| Pergi ke kesalahan sebelumnya | `Ctrl-Shift-F6` | `goToPreviousError` |
| Pergi ke kanan satu spasi | `Right` | `gotoright` |
| Pergi ke awal file | `Ctrl-Home` | `gotostart` |
| Pergi satu kata ke kiri | `Ctrl-Left` | `gotowordleft` |
| Pergi satu kata ke kanan | `Ctrl-Right` | `gotowordright` |
| Indent seleksi satu tab | `Tab` | `indent` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-J` | `joinlines` |
| Pergi ke simbol yang cocok dalam cakupan saat ini | `Ctrl-M` | `jumptomatching` |
| Meningkatkan ukuran huruf | `Ctrl-- \| Ctrl-= \| Ctrl-+` | `largerfont` |
| Mengurangi angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Alt-Down` | `modifyNumberDown` |
| Menambah angka di sebelah kiri kursor sebesar 1, jika itu adalah angka | `Alt-Up` | `modifyNumberUp` |
| Memindahkan seleksi ke bawah satu baris | `Ctrl-Shift-Down` | `movelinesdown` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memindahkan seleksi ke atas satu baris | `Ctrl-Shift-Up` | `movelinesup` |
| Meng-outdent seleksi satu tab | `Shift-Tab` | `outdent` |
| Mengaktifkan mode penimpaan, atau jika aktif, menonaktifkannya | `Insert` | `overwrite` |
| Menghapus isi dari baris saat ini | `Ctrl-Shift-K` | `removeline` |
| Menghapus dari kursor ke akhir baris saat ini | `Alt-Delete` | `removetolineend` |
| Menghapus dari awal baris saat ini hingga kursor | `Alt-Backspace` | `removetolinestart` |
| Menghapus kata di sebelah kiri kursor | `Ctrl-Backspace` | `removewordleft` |
| Menghapus kata di sebelah kanan kursor | `Ctrl-Delete` | `removewordright` |
| Memutar ulang penekanan tombol yang direkam sebelumnya | `Ctrl-Shift-Q` | `replaymacro` |
| Menggulir file saat ini ke bawah satu baris | `Ctrl-Down` | `scrolldown` |
| Menggulir file saat ini ke atas satu baris | `Ctrl-Up` | `scrollup` |
| Memilih semua konten yang dapat dipilih | `Ctrl-A` | `selectall` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan baris berikutnya ke bawah dalam seleksi | `Shift-Down` | `selectdown` |
| Menyertakan spasi berikutnya ke kiri dalam seleksi | `Shift-Left` | `selectleft` |
| Menyertakan sisa baris saat ini dalam seleksi, mulai dari kursor | `Shift-End` | `selectlineend` |
| Menyertakan awal baris saat ini dalam seleksi, hingga kursor | `Shift-Home` | `selectlinestart` |
| Menyertakan lebih banyak seleksi yang cocok yang setelah seleksi | `Ctrl-Alt-Right` | `selectMoreAfter` |
| Menyertakan lebih banyak seleksi yang cocok yang sebelum seleksi | `Ctrl-Alt-Left` | `selectMoreBefore` |
| Menyertakan seleksi pencocokan berikutnya yaitu setelah seleksi | `Ctrl-Alt-Shift-Right` | `selectNextAfter` |
| Menyertakan seleksi pencocokan berikutnya yaitu sebelum seleksi | `Ctrl-Alt-Shift-Left` | `selectNextBefore` |
| Memilih atau menemukan seleksi pencocokan berikutnya | `Alt-K` | `selectOrFindNext` |
| Memilih atau menemukan seleksi pencocokan sebelumnya | `Alt-Shift-K` | `selectOrFindPrevious` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan dari kursor turun ke akhir halaman saat ini dalam seleksi | `Shift-Page Down` | `selectpagedown` |
| Menyertakan dari kursor naik ke awal halaman saat ini dalam seleksi | `Shift-Page Up` | `selectpageup` |
| Menyertakan spasi berikutnya ke kanan kursor dalam seleksi | `Shift-Right` | `selectright` |
| Menyertakan dari kursor turun ke akhir file saat ini dalam seleksi | `Ctrl-Shift-End` | `selecttoend` |
| Menyertakan dari kursor ke akhir baris saat ini dalam seleksi | `Alt-Shift-Right` | `selecttolineend` |
| Menyertakan dari awal baris saat ini hingga kursor dalam seleksi | `Alt-Shift-Left` | `selecttolinestart` |
| Menyertakan dari kursor ke simbol pencocokan berikutnya dalam cakupan saat ini | `Ctrl-Shift-P` | `selecttomatching` |
| Menyertakan dari kursor naik ke awal file saat ini dalam seleksi | `Ctrl-Shift-Home` | `selecttostart` |
| Menyertakan baris berikutnya ke atas dalam seleksi | `Shift-Up` | `selectup` |
| Menyertakan kata berikutnya ke kiri kursor dalam seleksi | `Ctrl-Shift-Left` | `selectwordleft` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Menyertakan kata berikutnya ke kanan kursor dalam seleksi | `Ctrl-Shift-Right` | `selectwordright` |
| Menampilkan tab Preferensi | `Ctrl-,` | `showSettingsMenu` |
| Mengosongkan semua seleksi sebelumnya | `Esc` | `singleSelection` |
| Mengurangi ukuran huruf | `Ctrl-- \| Ctrl-Shift-= \| Ctrl-Shift-+` | `smallerfont` |
| Jika beberapa baris dipilih, mengatur ulang baris menjadi urutan yang diurutkan | `F9` | `sortlines` |
| Menambahkan kursor di akhir baris saat ini | `Ctrl-Shift-L` | `splitIntoLines` |
| Mengelilingi seleksi dengan karakter komentar blok, atau menghapusnya jika sudah ada | `Ctrl-Shift-/` | `toggleBlockComment` |
| Menambahkan karakter komentar baris di awal setiap baris yang dipilih, atau menghapusnya jika sudah ada | `Ctrl-/` | `togglecomment` |
| Melipat kode, atau menghapus lipatan kode jika sudah ada | `Ctrl-Shift-[` | `toggleFoldWidget` |
| Melipat kode induk, atau menghapus lipatan jika sudah ada | `Alt-F2` | `toggleParentFoldWidget` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Memulai perekaman penekanan tombol, atau menghentikan jika sudah merekam | `Ctrl-Q` | `togglerecording` |
| Membungkus kata-kata, atau menghentikan membungkus kata-kata jika sudah membungkus | `Ctrl-Q` | `toggleWordWrap` |
| Mengubah seleksi ke semua huruf kecil | `Ctrl-K Ctrl-L` | `tolowercase` |
| Mengubah seleksi ke semua huruf besar | `Ctrl-K Ctrl-U` | `touppercase` |
| Menranspos seleksi | `Alt-X` | `transposeletters` |
| Membuka lipatan kode yang dipilih | `Ctrl-Shift-]` | `unfold` |
| Membuka lipatan dari pelipatan kode untuk seluruh file | `Ctrl-K Ctrl-0 \| Ctrl-K Ctrl-J` | `unfoldall` |

## Emmet

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Mengevaluasi ekspresi matematika sederhana (seperti `2*4` atau `10/2`), dan mengeluarkan hasilnya | `Shift-Ctrl-Y` | `emmet_evaluate_math_expression` |
| Memperluas singkatan seperti CSS ke dalam kode HTML, | `Ctrl-Alt-E` | `emmet_expand_abbreviation` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| XML, atau CSS, tergantung pada sintaks file saat ini | | |
| Melintasi singkatan seperti CSS yang diperluas, dengan perhentian tab | `Tab` | `emmet_expand_abbreviation_with_tab` |
| Pergi ke bagian kode yang dapat diedit berikutnya | `Shift-Ctrl-.` | `emmet_select_next_item` |
| Pergi ke bagian kode yang dapat diedit sebelumnya | `Shift-Ctrl-,` | `emmet_select_previous_item` |
| Memperluas singkatan, dan kemudian menempatkan seleksi saat ini dalam elemen terakhir dari cuplikan dihasilkan | `Shift-Ctrl-A` | `emmet_wrap_with_abbreviation` |

## Terminal

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membuka tab Terminal baru | `Alt-T` | `openterminal` |
| Beralih antara editor dan tab Terminal | `Alt-S` | `switchterminal` |

## Jalankan dan Debug

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Membangun file saat ini | `F7 \| Ctrl-B` | `build` |

| Deskripsi | Keybinding | Perintah |
| --- | --- | --- |
| Melanjutkan proses terjeda saat ini | `F8` | `resume` |
| Menjalankan atau men-debug aplikasi saat ini | `Ctrl-Shift-B` | `run` |
| Menjalankan atau men-debug file yang dijalankan terakhir | `F5` | `runlast` |
| Melangkah masuk ke fungsi yang ada di tumpukan berikutnya | `F11` | `stepinto` |
| Melangkah keluar dari cakupan fungsi saat ini | `Shift-F11` | `stepout` |
| Melangkahi ekspresi saat ini di tumpukan | `F10` | `stepover` |
| Berhenti menjalankan atau men-debug aplikasi saat ini | `Shift-F5` | `stop` |
| Berhenti membangun file saat ini | `Ctrl-Break` | `stopbuild` |

# Referensi perintah untuk AWS Cloud9 IDE

Untuk menjalankan perintah di AWS Cloud9 Integrated Development Environment (IDE):

1. Pilih tombol Pergi (kaca pembesar) untuk menampilkan jendela Pergi. Jika tombol Pergi tidak terlihat, pilih Jendela, Pergi di bilah menu.
2. Di kotak Pergi ke Apa Pun, mulailah mengetik nama grup perintah (Editor Kode, misalnya). Sebuah grup berisi beberapa perintah yang diatur di sekitar tema umum atau fitur IDE.
3. Di bawah judul Perintah, pilih dari grup perintah khusus untuk dijalankan.

Grup perintah yang tersedia

| Grup perintah | Deskripsi |
| --- | --- |
| `AWS` | Perintah untuk AWS Toolkit |
| `Clipboard` | Perintah untuk menyalin dan menempel konten |
| `Code Editor` | Perintah untuk menavigasi antarmuka editor kode dan berinteraksi dengan konten editor |
| `Emmet` | Perintah untuk bekerja dengan kit alat Emmet yang digunakan untuk konten HTML dan CSS |
| `General` | Berbagai perintah untuk mengelola konfigurasi IDE dan file proyek |
| `Panels` | Perintah untuk mengelola tampilan panel di antarmuka IDE |
| `Run & Debug` | Perintah untuk menjalankan dan men-debug proyek di AWS Cloud9 |
| `Tabs` | Perintah untuk mengelola tampilan dan navigasi tab di antarmuka IDE |
| `Terminal` | Perintah untuk mengelola terminal baris perintah |
| `Window` | Perintah untuk mengelola tata letak panel di jendela IDE |

# Bekerja dengan AWS layanan lain

Saat menggunakan AWS Cloud9, Anda dapat bekerja sama dengan Amazon Lightsail AWS CodeStar,, dan. AWS CodePipeline Topik di bagian ini cara melakukannya.

> **Important**
>
> Fitur AWS Toolkit menyediakan antarmuka visual yang nyaman untuk bekerja dengan AWS layanan utama seperti AWS Lambda, Amazon S3 AWS Serverless Application Model, dan Amazon. Lihat informasi yang lebih lengkap di Bekerja dengan AWS Toolkit.

Topik


- Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9
- Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE
- Bekerja dengan Amazon CodeCatalyst
- Bekerja dengan AWS CDK di AWS Cloud9 IDE


## Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9

Anda dapat menggunakan AWS Cloud9 IDE untuk bekerja dengan kode pada instans Amazon Lightsail yang telah dikonfigurasi sebelumnya dengan aplikasi dan kerangka kerja populer. Mereka termasuk WordPress, LAMP (Linux, Apache, MySQL, dan PHP), Node.js, NGINX, Drupal, dan Joomla. Linux distribusi disertakan seperti Amazon Linux, Ubuntu, Debian, FreeBSD, dan openSUSE.

Lightsail menyediakan solusi server pribadi virtual pengaturan yang mudah dan cepat. Lightsail menyediakan kapasitas komputasi, penyimpanan, dan jaringan serta kemampuan untuk menyebarkan dan mengelola situs web dan aplikasi web di cloud. Anda dapat menggunakan Lightsail untuk meluncurkan proyek Anda dengan cepat dengan harga bulanan yang rendah dan dapat diprediksi. Untuk informasi selengkapnya, lihat Fitur Amazon Lightsail.

Dalam topik ini, Anda membuat dan menyiapkan instance Lightsail berbasis Linux yang kompatibel dengannya. AWS Cloud9 Anda kemudian membuat dan menghubungkan lingkungan pengembangan AWS Cloud9 SSH ke instance Lightsail.

> **Note**
>
> Menyelesaikan prosedur ini dapat mengakibatkan biaya untuk Anda Akun AWS. Ini termasuk kemungkinan biaya untuk layanan seperti Lightsail. Untuk informasi lebih lanjut, lihat Harga Amazon Lightsail.
>
> Untuk menggunakan AWS Cloud9 IDE untuk bekerja dengan EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server yang tidak berisi kode sampel, lihatMemulai: tutorial basic.

- Langkah 1: Buat Instance Lightsail Berbasis Linux
- Langkah 2: Siapkan Instance untuk Menggunakannya AWS Cloud9
- Langkah 3: Buat dan Connect ke Lingkungan Pengembangan AWS Cloud9 SSH
- Langkah 4: Gunakan AWS Cloud9 IDE untuk Mengubah Kode pada Instance

## Langkah 1: Membuat instans Lightsail berbasis Linux

Pada langkah ini, Anda menggunakan konsol Lightsail untuk membuat instance EC2 Amazon yang menjalankan aplikasi dalam distribusi berbasis Linux. Contoh ini secara otomatis mencakup yang berikut:

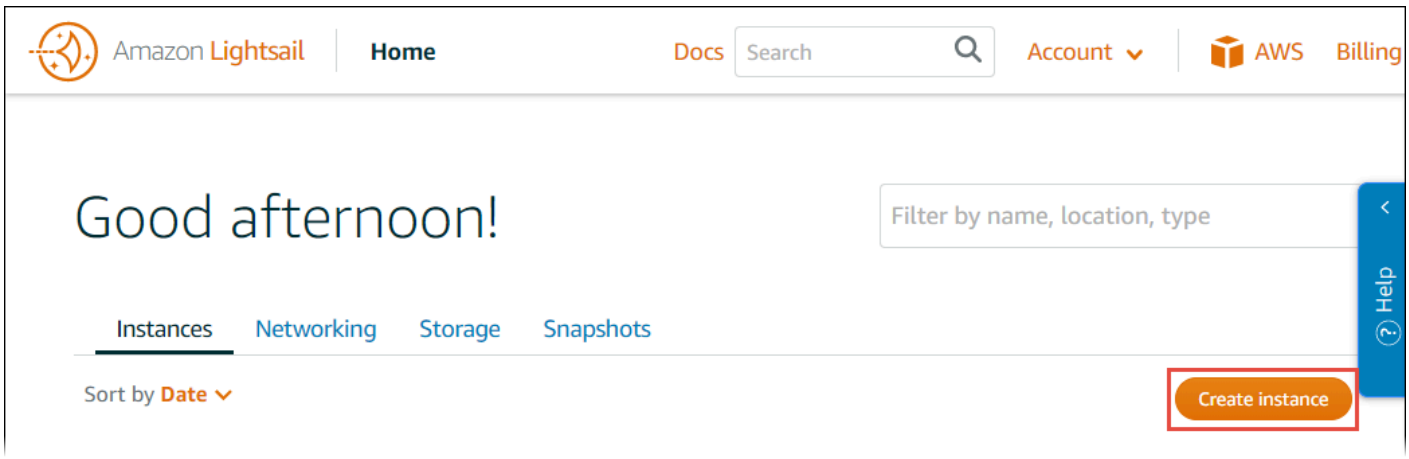
- Alamat IP privat dan publik. (Anda dapat membuat IP publik statis nanti.)
- Akses ke instans menggunakan SSH melalui port 22, HTTP melalui port 80, dan HTTPS melalui port 443. (Anda dapat mengubah pengaturan ini.)
- Disk penyimpanan blok. (Anda dapat melampirkan disk tambahan nanti.)
- Pelaporan sistem bawaan.

Pada konsol Lightsail, Anda dapat mencadangkan, reboot, menghentikan, atau menghapus instance nanti.

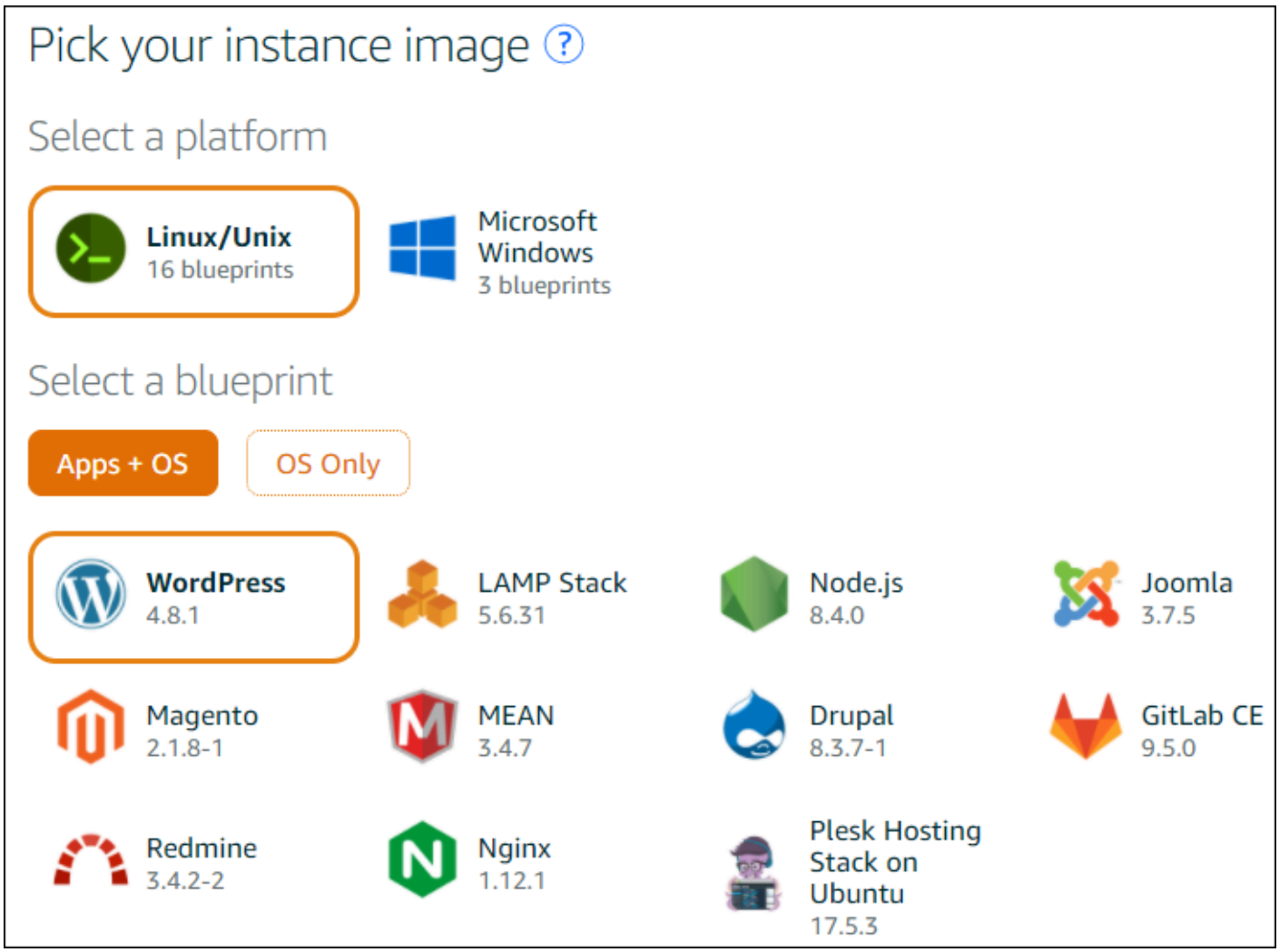
1. Buka dan kemudian masuk ke konsol Lightsail, di. https://lightsail.aws.amazon.com

   Kami menyarankan Anda masuk menggunakan kredensi untuk pengguna administrator IAM di Anda. Akun AWS Jika Anda tidak dapat masuk sebagai pengguna administrator IAM, periksa dengan Akun AWS administrator Anda.
2. Jika diminta, pilih bahasa yang akan digunakan di konsol, lalu pilih Simpan.

3. Jika diminta, pilih Mari kita mulai.
4. Di halaman beranda, dengan tab Instans sudah dipilih, pilih Buat instans.




5. Untuk lokasi Instance, pastikan lokasi Wilayah AWS AWS Cloud9 tersedia yang ingin Anda buat instance. Untuk informasi selengkapnya, lihat AWS Cloud9 di Referensi Umum Amazon Web Services. Untuk mengubah Wilayah AWS, Availability Zone, atau keduanya, pilih Ubah AWS Wilayah dan Availability Zone, lalu ikuti petunjuk di layar.
6. Untuk Pilih citra instans Anda, dengan Linux/UNIX sudah dipilih untuk Pilih platform, dan Aplikasi + OS sudah dipilih untuk Pilih cetak biru, pilih cetak biru.




> **Note**
>
> Jika Anda ingin membuat instans tanpa aplikasi, pilih Hanya OS sebagai gantinya Aplikasi + OS, lalu pilih distribusi.
> Untuk mempelajari tentang pilihan yang tersedia, lihat Memilih citra instans Amazon Lightsail di situs web Lightsail.

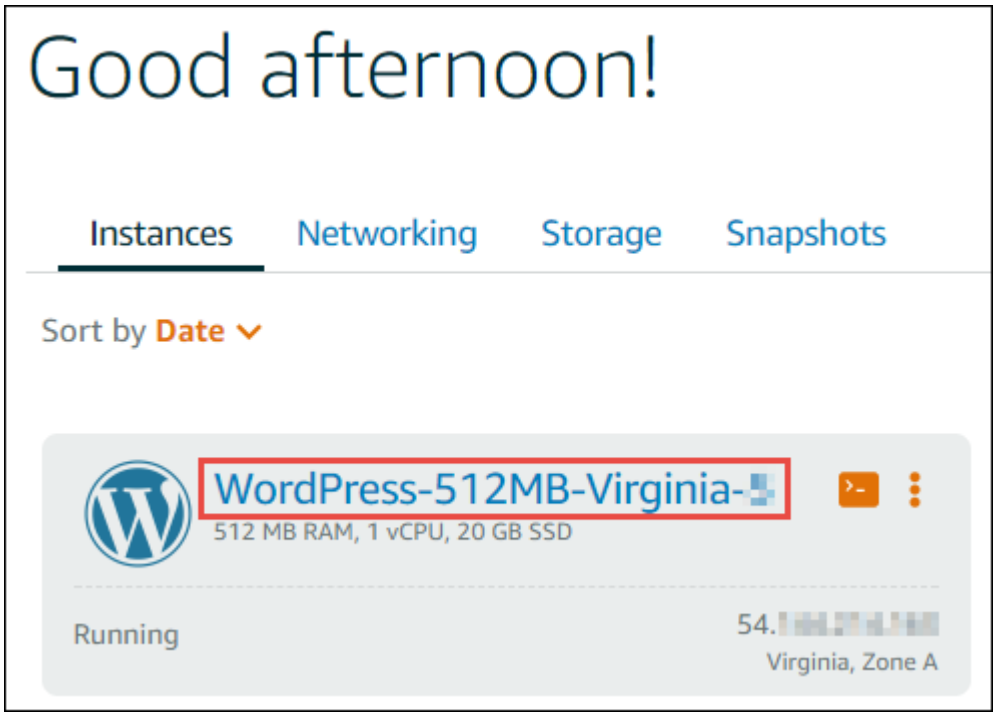
7. Untuk Pilih paket instans Anda, pilih paket, atau biarkan paket default yang dipilih.
8. Untuk Nama instance Anda, masukkan nama untuk instance, atau tinggalkan nama default yang disarankan.
9. Untuk jumlah instance, masukkan jumlah instance yang ingin Anda buat, atau biarkan default dari satu instance (x 1).
10. Pilih Buat.

# Langkah 2: Siapkan instance untuk menggunakannya AWS Cloud9

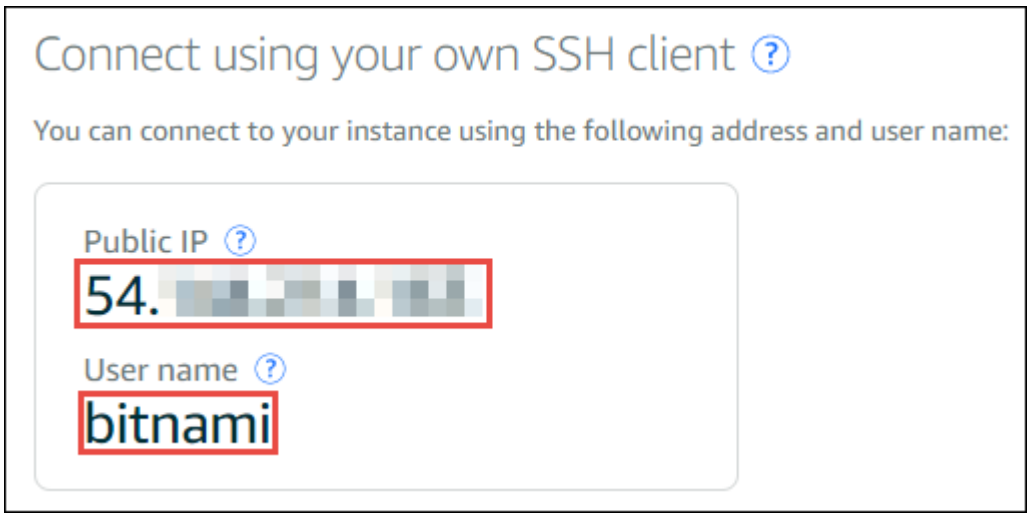
Pada langkah ini, Anda terhubung ke instance yang sedang berjalan dan kemudian mengaturnya sehingga AWS Cloud9 dapat menggunakannya nanti.

> **Note**
>
> Petunjuk berikut mengasumsikan Anda memilih Aplikasi + OS pada langkah sebelumnya. Jika Anda memilih Hanya OS dan distribusi selain Ubuntu, Anda mungkin perlu menyesuaikan instruksi berikut.

1. Dengan konsol Lightsail yang masih terbuka dari langkah sebelumnya, pada tab Instans, di kartu untuk instans, pilih nama instans.




2. Pada tab Connect, untuk Connect menggunakan klien SSH Anda sendiri, perhatikan nilai IP Publik dan nama Pengguna, seperti yang Anda butuhkan nanti.




3. Pilih Hubungkan menggunakan SSH.
4. Pastikan instans memiliki pembaruan sistem terbaru. Untuk melakukan ini, di sesi terminal yang muncul, jalankan perintah **`sudo apt update`** .
5. Periksa untuk melihat apakah Python diinstal, dan jika ya, periksa untuk memastikan versinya 2.7. Untuk memeriksa versi, jalankan perintah **`python --version`** , dan perhatikan nomor versi yang muncul. Jika tidak ada nomor versi yang muncul, atau jika versinya bukan 2.7, instal Python 2.7 pada instance dengan menjalankan perintah **`sudo apt install -y python-minimal`**.
6. Periksa untuk melihat apakah Node.js diinstal, dan jika ya, periksa apakah versinya 0.6.16 atau yang lebih baru. Untuk memeriksa versi, jalankan perintah **`node --version`** , dan perhatikan nomor versi yang muncul. Jika tidak ada nomor versi yang muncul, atau versi tidak 0.6.16 atau yang lebih baru, kami sarankan Anda menggunakan Node Version Manager (nvm) untuk menginstal Node.js pada contoh.

   Untuk melakukan ini, jalankan perintah berikut satu per satu, dalam urutan berikut, untuk memperbarui instance, instal Node Version Manager (nvm) pada instance, aktifkan nvm pada instance, lalu instal versi terbaru Node.js pada contoh.

   ```
   sudo apt update
   curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.0/install.sh | bash
   . ~/.bashrc
   nvm install node
   ```

7. Jalankan perintah **`which node`**, dan perhatikan nilai yang muncul. Anda akan membutuhkannya nanti.

> **Note**
>
> Jika output dari perintah **`which node`**adalah sesuatu seperti`/usr/sbin/node`, tidak AWS Cloud9 dapat menemukan Node.js di jalur itu. Sebagai gantinya, gunakan `nvm` untuk menginstal Node.js, seperti yang dijelaskan pada langkah sebelumnya dalam prosedur ini. Kemudian, jalankan perintah `which node` lagi dan perhatikan nilai baru yang muncul.

8. Unduh dan jalankan AWS Cloud9 Installer pada instance.

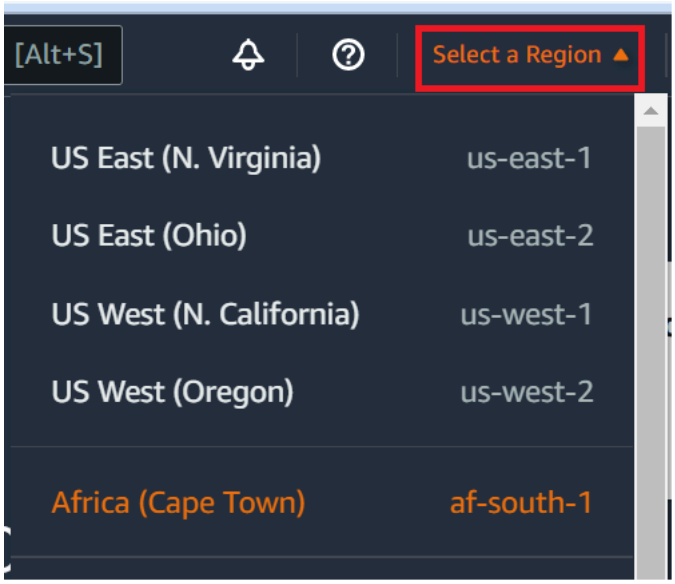
## Langkah 3: Membuat dan menghubungkan ke Lingkungan Pengembangan AWS Cloud9 SSH

Pada langkah ini, Anda menggunakan AWS Cloud9 konsol dan terminal instance untuk membuat lingkungan SSH dan kemudian menghubungkan lingkungan ke instance yang sedang berjalan.

1. Dengan sesi terminal masih terbuka dari langkah sebelumnya, masuk ke AWS Cloud9 konsol, sebagai berikut:
   - Jika Anda satu-satunya individu yang menggunakan Anda Akun AWS atau Anda adalah pengguna IAM dalam satu Akun AWS, buka. https://console.aws.amazon.com/cloud9/
   - Jika organisasi Anda menggunakan AWS IAM Identity Center, lihat Akun AWS administrator untuk petunjuk masuk.

   > **Note**
   >
   > Untuk langkah ini, Anda akan bekerja dengan dua yang berbeda Layanan AWS pada saat yang bersamaan. Sekarang, misalkan Anda masuk ke konsol Lightsail sebagai pengguna administrator IAM, tetapi Anda ingin entitas yang berbeda memiliki lingkungan SSH baru. Untuk kasus ini, kami sarankan membuka browser web yang berbeda dan masuk ke AWS Cloud9 konsol sebagai entitas itu.

2. Di AWS Cloud9 konsol, pilih Wilayah AWS yang cocok dengan yang Anda buat instance dalam kerangka kerja.




3. Jika halaman selamat datang ditampilkan, untuk AWS Cloud9 Lingkungan baru, pilih Buat lingkungan. Jika tidak, pilih Buat lingkungan.

Atau:

4. Pada halaman Nama lingkungan, untuk Nama, memasukkan nama untuk lingkungan Anda.
5. Tambahkan deskripsi ke lingkungan Anda di bidang Deskripsi.
6. Untuk jenis Lingkungan pilih Komputasi yang ada. Ini penting karena Anda perlu memilih opsi ini untuk menampilkan opsi Pengguna dan Host.
7. Untuk Pengguna, masukkan nilai Nama pengguna yang Anda catat sebelumnya.
8. Untuk Host, masukkan nilai IP Publik yang Anda catat sebelumnya.
9. Untuk Port, biarkan nilai default 22.
10. Perluas Detail tambahan.
11. Untuk jalur Lingkungan, masukkan jalur yang AWS Cloud9 dimulai dari setelah login, yaitu~/. Ini adalah root dari direktori home pengguna.

12. Untuk jalur biner Node.js, masukkan nilai perintah **`which node`** yang Anda catat sebelumnya.

13. Biarkan Host lompatan SSH kosong.

14. Simpan kunci SSH publik yang AWS Cloud9 dibuat untuk lingkungan ini di clipboard sistem Anda. Untuk melakukannya, pilih Salin kunci ke clipboard.

> **Note**
>
> Untuk melihat nilai kunci SSH publik yang disalin, perluas Lihat kunci SSH publik.

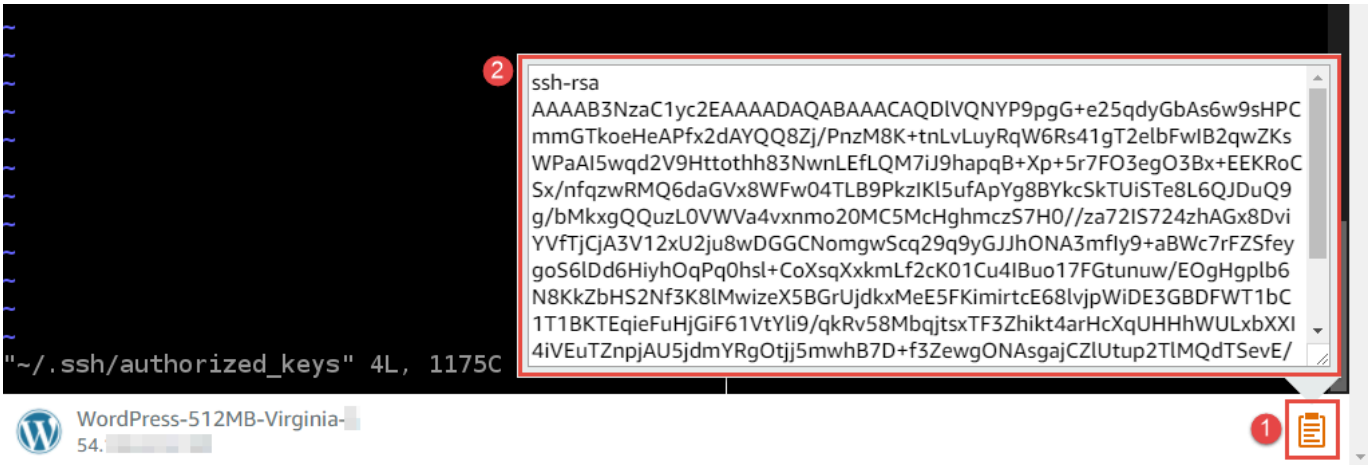
15. Simpan nilai kunci SSH publik yang baru saja Anda salin ke instans. Untuk melakukan ini, gunakan vi, editor teks populer, yang sudah diinstal pada contoh:

    a. Di sesi terminal untuk instans, jalankan perintah **`vi ~/.ssh/authorized_keys`** .

    b. Dalam vi editor yang muncul, pergi ke akhir file, dan beralih ke mode insert. Untuk melakukannya, tekan `I`, kemudian `A`. (-- INSERT -- muncul di bagian bawah vi editor.)

    c. Tambahkan dua carriage return ke akhir file dengan menekan `Enter` dua kali.

    d. Tempel konten clipboard sistem Anda, yang berisi nilai kunci SSH publik yang baru saja Anda salin, ke clipboard sesi terminal. Untuk melakukan ini, di sudut bawah jendela sesi terminal, pilih tombol clipboard, lalu tempel isi clipboard sistem Anda ke dalam kotak.




    e. Tempelkan isi clipboard sesi terminal ke editor vi. Untuk melakukan ini, pada titik penyisipan di editor vi, tekan `Ctrl + Shift + V`.

    f. Simpan file tersebut. Untuk melakukan ini, tekan `Esc` untuk masuk ke mode perintah. (— INSERT — menghilang dari bagian bawah editor vi.) Ketik `:wq` (`write`ke file dan kemudian `quit` vi editor), lalu tekan`Enter`.

16. Kembali ke AWS Cloud9 konsol, pilih Langkah selanjutnya.

17. Di halaman Tinjauan pilihan, pilih Buat lingkungan. Tunggu sementara AWS Cloud9 menciptakan lingkungan Anda dan kemudian menampilkan AWS Cloud9 IDE untuk lingkungan. Ini dapat memakan waktu beberapa menit.

Setelah AWS Cloud9 menciptakan lingkungan Anda, ini menampilkan AWS Cloud9 IDE untuk lingkungan.

Jika AWS Cloud9 tidak menampilkan IDE setelah setidaknya lima menit, mungkin ada masalah dengan browser web Anda, izin AWS akses Anda, instance, atau virtual private cloud (VPC) terkait. Untuk kemungkinan perbaikan, lihat Tidak Dapat Membuka Lingkungan di Pemecahan Masalah.

## Langkah 4: Gunakan AWS Cloud9 IDE untuk mengubah kode pada instance

Sekarang setelah IDE muncul untuk lingkungan baru, Anda dapat menggunakan sesi terminal di IDE sebagai ganti sesi terminal Lightsail. IDE menyediakan pengalaman pengeditan kode yang kaya dengan dukungan untuk beberapa bahasa pemrograman dan debugger runtime. IDE juga mencakup tema warna, keybinding pintasan, pewarnaan sintaks khusus bahasa pemrograman dan pemformatan kode.

Untuk mempelajari cara menggunakan IDE, lihat Tur AWS Cloud9 IDE.

Untuk mempelajari cara mengubah kode pada instans Anda, kami merekomendasikan sumber daya berikut:

- AllMendapatkan kata sandi aplikasi untuk 'powered by Anda Bitnami'Gambar Lightsail di situs web Lightsail
- Drupal: BitnamiDrupal Untuk AWS Cloud di Bitnami situs web, dan Tutorial dan resep situs di Drupal situs web
- GitLab CE: BitnamiGitLab CE untuk AWS Cloud di Bitnami situs web, dan GitLab Dokumentasi pada GitLab situs web
- Joomla: BitnamiJoomla! Untuk AWS Cloud di Bitnami situs web, dan Memulai dengan Joomla!pada Joomla! situs web
- LAMP Tumpukan: BitnamiLAMP untuk AWS Cloud di Bitnami situs web
- Magento: BitnamiMagento Untuk AWS Cloud di Bitnami situs web, dan Panduan Pengguna Magento di Magento situs web
- MEAN: BitnamiMEAN Untuk AWS Cloud di Bitnami situs web

- NGINX: BitnamiNGINX Untuk AWS Cloud di Bitnami situs web, dan NGINX Wiki di NGINX situs web
- Node.js: BitnamiNode.Js Untuk AWS Cloud di Bitnami situs web, dan Panduan Memulai di situs web Node.js
- Plesk Hosting Stack on Ubuntu: Siapkan dan konfigurasikan Plesk di Amazon Lightsail.
- Redmine: Bitnami Redmine Untuk AWS Cloud di Bitnami situs web, dan Memulai di Redmine situs web
- WordPress: Memulai menggunakan WordPress dari instans Amazon Lightsail Anda di situs web Lightsail, dan Bitnami WordPress Untuk AWS Cloud di Bitnami situs web

# Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE

Anda dapat menggunakan AWS Cloud9 Integrated Development Environment (IDE) untuk bekerja dengan kode sumber di repositori yang kompatibel dengannya. AWS CodePipeline

CodePipeline adalah layanan pengiriman berkelanjutan yang dapat Anda gunakan untuk memodelkan, memvisualisasikan, dan mengotomatiskan langkah-langkah yang diperlukan untuk merilis perangkat lunak Anda dan perubahan berkelanjutan yang Anda buat untuk itu. Anda dapat menggunakannya CodePipeline untuk memodelkan dan mengonfigurasi berbagai tahapan proses rilis perangkat lunak dengan cepat. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS CodePipeline.

> **Note**
>
> Menyelesaikan prosedur ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2, Amazon S3 CodePipeline, dan AWS layanan yang didukung oleh. CodePipeline Untuk informasi selengkapnya, lihat EC2Harga Amazon, AWS CodePipeline Harga, Harga Amazon S3, dan Harga Layanan Cloud.

- Langkah 1: Buat atau Identifikasi Repositori Kode Sumber Anda
- Langkah 2: Buat Lingkungan AWS Cloud9 Pengembangan, Hubungkan ke Repositori Kode, dan Unggah Kode Anda
- Langkah 3: Bersiaplah untuk Bekerja dengan AWS CodePipeline
- Langkah 4: Buat Pipeline di AWS CodePipeline

## Langkah 1: Buat atau identifikasi repositori kode sumber Anda

Pada langkah ini, Anda membuat atau mengidentifikasi repositori kode sumber yang kompatibel dengannya. CodePipeline

Kemudian dalam topik ini, Anda mengunggah kode sumber perangkat lunak Anda ke repositori itu. CodePipelineakan membangun, menguji, dan menyebarkan kode sumber yang diunggah di repositori itu dengan menggunakan pipeline terkait yang juga Anda buat.

Repositori kode sumber Anda harus salah satu jenis repositori berikut yang mendukung: CodePipeline

- AWS CodeCommit. Jika Anda sudah memiliki repositori CodeCommit yang ingin Anda gunakan, lewati ke Langkah 2: Buat Lingkungan AWS Cloud9 Pengembangan, Hubungkan ke Repositori Kode, dan Unggah Kode Anda. Jika tidak, untuk menggunakan CodeCommit, ikuti petunjuk ini dalam AWS CodeCommit Sampel dalam urutan ini, dan kemudian kembali ke topik ini:
  - Langkah 1: Siapkan Grup IAM Anda dengan Izin Akses yang Diperlukan
  - Langkah 2: Buat Repositori di AWS CodeCommit
- Amazon S3. Jika Anda sudah memiliki bucket di Amazon S3 yang ingin Anda gunakan, lanjutkan ke Langkah 2: Buat Lingkungan AWS Cloud9 Pengembangan, Hubungkan ke Repositori Kode, dan Unggah Kode Anda. Jika tidak, untuk menggunakan Amazon S3, ikuti petunjuk ini di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon dalam urutan ini, lalu kembali ke topik ini:
  - Mendaftar untuk Amazon S3
  - Buat Bucket
- GitHub. Jika Anda sudah memiliki repositori GitHub, Anda dapat mengkloningnya dan membuat salinan lokal di lingkungan pengembangan Anda menggunakan antarmuka panel Git. Jika Anda belum menyiapkan akun atau repositori GitHub, lihat dokumentasi yang relevan untuk mendapatkan instruksi.

## Langkah 2: Buat Lingkungan AWS Cloud9 Pengembangan, hubungkan ke repositori kode, dan unggah kode Anda

Pada langkah ini, Anda membuat lingkungan AWS Cloud9 pengembangan di AWS Cloud9 konsol. Anda kemudian menghubungkan lingkungan ke repositori yang CodePipeline akan digunakan. Terakhir, Anda menggunakan AWS Cloud9 IDE untuk lingkungan untuk mengunggah kode sumber Anda ke repositori.

Untuk membuat lingkungan, ikuti instruksi di Membuat Lingkungan, lalu kembali ke topik ini. (Jika Anda sudah memiliki lingkungan, Anda dapat menggunakannya. Anda tidak perlu membuat yang baru.)

Untuk menghubungkan lingkungan ke repositori, dan kemudian mengunggah kode sumber Anda ke repositori jika belum ada, gunakan salah satu dari rangkaian instruksi berikut. Rangkaian yang Anda pilih tergantung pada jenis repositori yang menyimpan kode sumber.

| Jenis repositori | Instruksi |
| --- | --- |
| CodeCommit | Ikuti petunjuk ini dalam Sampel AWS CodeCommit :<br>• Langkah 3: Hubungkan Lingkungan Anda ke Repositori Jarak Jauh<br>• Langkah 4: Kloning Repositori Jarak Jauh ke Lingkungan Anda<br>• Langkah 5: Tambahkan File ke Repositori, mengganti kode sumber Anda sendiri untuk langkah ini |
| Amazon S3 | • Instal dan konfigurasikan AWS CLI atau AWS CloudShell di lingkungan, seperti yang dijelaskan dalam AWS CLI dan AWS CloudShell Sampel.<br>• Untuk mengunggah kode sumber Anda ke bucket, gunakan AWS CLI atau AWS CloudShell di lingkungan untuk menjalankan perintah aws s3 cp. (Untuk AWS CloudShell, Anda dapat menghapus `aws` dari perintah.) |
| GitHub | Anda dapat mengkloning repositori yang di-host GitHub dan berinteraksi dengan menggunakan antarmuka panel Git. |

Setelah Anda menghubungkan lingkungan ke repositori, setiap kali Anda mendorong perubahan kode sumber dari AWS Cloud9 IDE ke repositori, CodePipeline secara otomatis mengirimkan

perubahan tersebut melalui pipeline terkait untuk dibangun, diuji, dan digunakan. Anda membuat alur terkait nanti di topik ini.

### Langkah 3: Bersiaplah untuk bekerja dengan AWS CodePipeline

Pada langkah ini, Anda melampirkan kebijakan AWS terkelola tertentu ke grup IAM yang Anda buat atau identifikasi dalam Pengaturan Tim. Hal ini memungkinkan pengguna grup untuk mulai membuat dan bekerja dengan pipeline di CodePipeline.

Jika Anda pernah menggunakan CodePipeline sebelumnya, lanjutkan ke Langkah 4: Buat Pipeline di AWS CodePipeline.

Untuk langkah ini, ikuti petunjuk ini di Langkah 3: Gunakan Kebijakan Terkelola IAM untuk Menetapkan AWS CodePipeline; Izin untuk Pengguna IAM di Panduan AWS CodePipeline Pengguna, lalu kembali ke topik ini.

### Langkah 4: Buat alur di AWS CodePipeline

Pada langkah ini, Anda membuat pipeline CodePipeline yang menggunakan repositori yang Anda buat atau identifikasi sebelumnya dalam topik ini.

Untuk langkah ini, ikuti instruksi di Membuat Alur di AWS CodePipeline di Panduan Pengguna AWS CodePipeline .

Setelah Anda membuat pipeline, CodePipeline kirimkan versi kode sumber saat ini di repositori melalui pipeline yang akan dibangun, diuji, dan digunakan. Kemudian, setiap kali Anda mendorong perubahan kode sumber dari AWS Cloud9 IDE ke repositori, CodePipeline secara otomatis mengirimkan perubahan tersebut melalui pipeline yang akan dibangun, diuji, dan diterapkan.

Untuk melihat alur, ikuti petunjuk di Lihat Detail dan Riwayat Alur di AWS CodePipeline di Panduan Pengguna AWS CodePipeline .

## Bekerja dengan Amazon CodeCatalyst

Amazon CodeCatalyst adalah ruang kolaborasi berbasis cloud untuk tim pengembangan perangkat lunak. CodeCatalyst adalah tempat terpadu untuk bekerja, berkolaborasi dalam kode, dan membangun, menguji, dan menyebarkan aplikasi dengan alat berkelanjutanintegration/delivery (CI/CD). Anda dapat menghubungkan AWS sumber daya dengan proyek Anda dengan menghubungkan Anda Akun AWS ke CodeCatalyst ruang Anda. Anda juga dapat menggunakannya CodeCatalyst untuk mengirimkan perangkat lunak dengan cara yang cepat dan percaya diri. Untuk informasi

selengkapnya CodeCatalyst, lihat Apa itu Amazon CodeCatalyst? di  CodeCatalyst Panduan Amazon.

Lingkungan Dev adalah lingkungan pengembangan berbasis cloud yang dapat Anda gunakan CodeCatalyst untuk mengerjakan kode yang disimpan di repositori sumber proyek Anda. Anda dapat membuat Lingkungan Pengembang di CodeCatalyst. Kemudian, saat berada di sana, Anda dapat mengerjakan kode dalam proyek khusus CodeCatalyst dengan lingkungan pengembangan terintegrasi (IDE) yang didukung. Atau, buat Lingkungan Dev kosong untuk mengkloning kode dari repositori pihak ketiga untuk dikerjakan dengan IDE yang didukung.

AWS Cloud9 IDE yang digunakan untuk mengakses Lingkungan Dev Anda di CodeCatalyst konsol berbeda dari AWS Cloud9 IDE yang berjalan AWS. Di CodeCatalyst AWS Cloud9 IDE, Anda secara otomatis masuk CodeCatalyst dan dapat mengakses layanan menggunakan opsi aws-explorer dalam IDE. Untuk informasi selengkapnya tentang AWS Toolkit, lihat AWS Toolkit untuk AWS Cloud9 di Panduan.AWS Cloud9

Topik

- Memulai dengan Amazon CodeCatalyst di AWS Cloud9
- Bermigrasi dari AWS Cloud9 ke Amazon CodeCatalyst
- Menggunakan alat replikasi
- FAQs tentang proses replikasi
- Lingkungan Pengembang di Amazon CodeCatalyst

## Memulai dengan Amazon CodeCatalyst di AWS Cloud9

Bagian ini memberikan ikhtisar tentang cara mulai menggunakan CodeCatalyst. Topik di bagian ini mencakup cara menggunakan AWS Cloud9 di Amazon CodeCatalyst dan cara mereplikasi AWS Cloud9 lingkungan Anda. CodeCatalyst Topik selanjutnya juga merinci cara membuat Lingkungan CodeCatalyst Dev dan cara mengakses Lingkungan Dev Anda menggunakan IDE. AWS Cloud9

AWS Toolkit adalah kit pengembangan perangkat lunak khusus IDE (SDKs) yang menyediakan akses cepat ke AWS Cloud akun, layanan, dan sumber daya. Dari CodeCatalyst akun Anda di AWS Toolkit, Anda dapat melihat, mengedit, dan mengelola Lingkungan CodeCatalyst Dev, Spasi, dan proyek Anda dalam antarmuka yang nyaman. Untuk mempelajari lebih lanjut tentang AWS Cloud layanan dan fitur yang tersedia melalui AWS Toolkit, lihat Apa itu? AWS Toolkit for Visual Studio Code , AWS Toolkit untuk AWS Cloud9, dan Apa itu? AWS Toolkit for JetBrains . Apa AWS Toolkit for JetBrains panduannya.

Untuk menggunakan CodeCatalyst AWS Cloud9 IDE, Anda harus memiliki Space, project, dan Dev Environment yang sudah ada yang Anda buat di dalam CodeCatalyst konsol.

> **Note**
>
> Jangan membuat subfolder bernama proyek dalam folder dengan nama yang sama dalam Sistem File AWS Cloud9 IDE untuk CodeCatalyst. Jika Anda melakukannya, Anda tidak dapat mengakses file apa pun dalam direktori ini. Masalah ini memengaruhi jalur file /projects/projects. Jalur file seperti /test/projects dan/projects/test/projectstidak terpengaruh oleh masalah ini. Ini adalah masalah yang diketahui dan hanya mempengaruhi AWS Cloud9 IDE File Explorer.

> **Note**
>
> Saat ini tidak mungkin untuk membuat subfolder bernama proyek dalam folder dengan nama yang sama, menggunakan Sistem File AWS Cloud9 IDE untuk CodeCatalyst. Anda tidak akan dapat mengakses file apa pun dalam direktori ini dari AWS Cloud9 IDE File Explorer, tetapi Anda akan dapat mengaksesnya menggunakan baris perintah. Silakan gunakan nama folder alternatif. Masalah ini hanya memengaruhi jalur file /projects/projects, jalur file seperti /test/projects dan/seharusnya berfungsi. projects/test/projects Ini adalah masalah yang diketahui dan hanya mempengaruhi AWS Cloud9 IDE File Explorer.

## Bermigrasi dari AWS Cloud9 ke Amazon CodeCatalyst

AWS Cloud9 in CodeCatalyst memberikan pengalaman yang dikelola sepenuhnya untuk berinteraksi dengan AWS Cloud9. Anda dapat mereplikasi sumber daya AWS Cloud9 kode Anda saat ini secara manual di Amazon CodeCatalyst. Prosesnya dirinci di bagian berikut. Untuk memindahkan sumber daya kode Anda dan mereplikasinya, buat Spasi di dalamnya CodeCatalyst. Ruang mewakili perusahaan, departemen, atau grup Anda. Anda perlu membuat spasi untuk menambahkan proyek, anggota, dan sumber daya cloud terkait yang Anda buat CodeCatalyst. Ketika pengguna menerima undangan ke proyek, CodeCatalyst secara otomatis menambahkannya ke ruang. Pengguna dengan peran administrator Space dapat mengelola ruang.

Dalam ruang ini, Anda membuat proyek dan menambahkan repositori sumber Anda. Proyek adalah ruang kolaborasi CodeCatalyst yang mendukung tim dan tugas pengembangan. Setelah membuat

proyek, Anda dapat menambahkan, memperbarui, atau menghapus sumber daya. Anda juga dapat menyesuaikan dasbor proyek Anda, dan memantau kemajuan pekerjaan tim Anda. Anda dapat memiliki beberapa proyek dalam satu ruang. Jumlah repositori sumber yang Anda tambahkan tergantung pada jumlah repositori yang sudah Anda gunakan di lingkungan Anda. AWS Cloud9 Setelah Anda membuat proyek ini dan menambahkan repositori sumber yang berlaku, Anda mungkin perlu kembali ke AWS Cloud9 lingkungan Anda dan mereplikasi data lingkungan ke repositori baru ini di. CodeCatalyst Apa yang Anda lakukan tergantung pada jenis repositori sumber yang Anda miliki. AWS Cloud9

Setelah Anda membuat ruang, proyek, dan repositori sumber, Anda dapat meluncurkan lingkungan Anda dalam CodeCatalyst menggunakan AWS Cloud9 dengan Lingkungan Dev. Dev Environment adalah lingkungan pengembangan berbasis cloud. Anda dapat menggunakan Dev Environment CodeCatalyst untuk mengerjakan kode yang disimpan di repositori sumber proyek Anda. Anda juga dapat membuat Lingkungan Dev CodeCatalyst untuk mengerjakan kode di Lingkungan Dev khusus proyek dengan lingkungan pengembangan terintegrasi (IDE) yang didukung.

Anda juga dapat mereplikasi sumber daya AWS Cloud9 kode Anda saat ini untuk CodeCatalyst menggunakan alat replikasi. Ini adalah alat yang Anda unduh dan jalankan di AWS Cloud9 lingkungan Anda. Jika Anda sudah mendaftar CodeCatalyst, dan membuat spasi, alat ini secara otomatis membuat proyek dalam ruang ini dan mereplikasi sumber daya kode Anda ke repositori baru di. CodeCatalyst Mirip dengan proses replikasi manual. Ini tergantung pada jenis repositori sumber yang Anda miliki. AWS Cloud9 Misalnya, jika Anda memiliki GitHub repositori, Anda masih perlu mereplikasi repositori ini menggunakan GitHub ekstensi  di CodeCatalyst konsol.

- Langkah 1. Mendaftar ke Amazon CodeCatalyst dan membuat Space
- Langkah 2. Membuat proyek di Ruang Anda
- Langkah 3. Membuat repositori sumber di proyek Anda
- Langkah 4. Mereplikasi sumber daya AWS Cloud9 kode Anda ke sumber repositori di CodeCatalyst
- Langkah 5. Membuat Lingkungan Dev dalam CodeCatalyst menggunakan AWS Cloud9

## Langkah 1. Mendaftar ke Amazon CodeCatalyst dan membuat Space

Anda dapat mendaftar ke Amazon CodeCatalyst tanpa undangan ke ruang atau proyek yang ada. Saat Anda mendaftar, Anda membuat ruang dan proyek. Anda dapat memasukkan Akun AWS ID yang ada yang Anda gunakan untuk AWS Cloud9. Hal yang sama Akun AWS dapat digunakan untuk tujuan penagihan. Untuk informasi tentang cara menemukan Akun AWS ID Anda, lihat Akun AWS ID

Anda dan aliasnya. Ikuti prosedur ini untuk mendaftar ke CodeCatalyst profil Amazon Anda, membuat spasi, dan menambahkan akun untuk ruang Anda.

Untuk mendaftar sebagai pengguna baru

1. Buka konsol CodeCatalyst.
2. Pada halaman selamat datang, pilih Daftar.

   Halaman Create your AWS Builder ID ditampilkan. AWS Builder ID Anda adalah identitas yang Anda buat untuk masuk. ID ini tidak sama dengan Akun AWS ID. Untuk mempelajari lebih lanjut tentang ID AWS Pembuat, lihat ID AWS Pembuat dan AWS kredensyal lainnya di Panduan Pengguna AWS Masuk.
3. Untuk alamat email Anda, masukkan alamat email yang ingin Anda kaitkan CodeCatalyst. Lalu, pilih Selanjutnya.
4. Untuk nama Anda, masukkan nama depan dan belakang yang ingin ditampilkan di aplikasi tempat Anda menggunakan AWS Builder ID.

   Nama ini adalah nama profil AWS Builder ID Anda. Jika mau, Anda bisa mengganti namanya nanti.

   Pilih Berikutnya. Halaman verifikasi Email akan muncul. Kode verifikasi dikirim ke email yang Anda tentukan.
5. Untuk kode Verifikasi, masukkan kode yang Anda terima, lalu pilih Verifikasi.

   Jika Anda tidak menerima kode setelah 5 menit dan tidak dapat menemukannya di folder spam atau sampah, pilih Kirim ulang kode.
6. Setelah kode Anda diverifikasi, masukkan kata sandi dan pilih Konfirmasi kata sandi.

   Pilih kotak centang yang mengonfirmasi bahwa Anda membaca dan mengakui Perjanjian AWS Pelanggan dan Ketentuan AWS Layanan, lalu pilih Buat profil saya.
7. Pada halaman Buat alias Anda, masukkan alias yang akan digunakan. CodeCatalyst CodeCatalyst Pengguna lain akan menggunakan alias ini untuk @mention Anda di komentar dan permintaan tarik. CodeCatalyst Profil Anda akan berisi nama lengkap Anda dari AWS Builder ID dan CodeCatalyst alias Anda. Anda tidak dapat mengubah CodeCatalyst alias Anda.

   Nama lengkap Anda dan alias Anda akan ditampilkan di berbagai area di CodeCatalyst. Misalnya, nama profil Anda muncul di feed aktivitas Anda, tetapi anggota proyek akan menggunakan alias Anda untuk @mention Anda.

Pilih Buat alias. Halaman diperbarui untuk menampilkan bagian Buat ruang Anda.

8. Untuk nama Spasi, masukkan nama spasi Anda, lalu pilih Berikutnya.

   Anda tidak dapat mengubah nama ini.

9. Untuk Akun AWS ID, tautkan ID dua belas digit untuk akun yang ingin Anda sambungkan ke ruang Anda.

   Dalam token Akun AWS verifikasi, salin ID token yang dihasilkan. Token secara otomatis disalin untuk Anda. Tapi, Anda mungkin ingin menyimpannya saat Anda menyetujui permintaan AWS koneksi.

10. Pilih Verifikasi di AWS.

11. Halaman  CodeCatalyst spasi Verifikasi Amazon terbuka di AWS Management Console.

    Ini adalah halaman Amazon CodeCatalyst Spaces. Anda mungkin perlu masuk untuk mengakses halaman.

    Untuk mengakses halaman, masuk ke Amazon CodeCatalyst Spaces di halaman AWS Management Console.

    Bidang token verifikasi di dalam AWS Management Console secara otomatis diisi dengan token yang dihasilkan di CodeCatalyst.

12. Pilih Verifikasi ruang.

    Pesan sukses terverifikasi Akun ditampilkan untuk menunjukkan bahwa akun telah ditambahkan ke ruang.

    Anda akan menggunakan tingkat CodeCatalyst gratis secara default. Jika Anda ingin mengubah, pilih Untuk mengaktifkan tingkat Standar atau menambahkan peran IAM untuk ruang ini, lihat detail ruang.

    Untuk informasi selengkapnya tentang tingkatan CodeCatalyst harga, lihat Amazon CodeCatalyst - Harga.

    Halaman detail CodeCatalyst spasi terbuka di AWS Management Console. Ini adalah halaman Amazon CodeCatalyst Spaces. Anda mungkin perlu masuk untuk mengakses halaman.

13. Pilih Pergi ke Amazon CodeCatalyst.

14. Pada halaman pembuatan di CodeCatalyst, pilih Buat ruang.

Pesan status ditampilkan saat ruang Anda sedang dibuat. Saat ruang dibuat, CodeCatalyst buka halaman untuk ruang Anda. Tampilan default ke tab Projects.

> **Note**
>
> Jika kesalahan izin atau spanduk ditampilkan, maka segarkan halaman dan coba lihat halaman lagi.

Setelah Anda mendaftar CodeCatalyst dan membuat spasi, langkah selanjutnya dalam proses replikasi adalah membuat proyek dalam ruang ini.

## Langkah 2. Membuat proyek di Ruang Anda

Langkah-langkah berikut menguraikan cara membuat proyek kosong di dalam ruang yang Anda buat pada langkah sebelumnya. Dengan proyek ini, Anda dapat menambahkan sumber daya yang Anda inginkan secara manual di lain waktu. Sebelum Anda membuat proyek, Anda harus memiliki peran administrator Space, dan Anda harus bergabung dengan ruang tempat Anda ingin membuat proyek. Saat Anda membuat spasi, CodeCatalyst secara otomatis memberi Anda peran administrator Space. Peran administrator Space adalah peran yang paling kuat dalam CodeCatalyst. Untuk informasi selengkapnya tentang peran ini dan izinnya, lihat Peran administrator ruang.

Untuk membuat proyek kosong

1. Arahkan ke ruang tempat Anda ingin membuat proyek.
2. Di dasbor ruang, pilih Buat proyek.
3. Pilih Mulai dari awal.
4. Di bawah Berikan nama untuk proyek Anda, masukkan nama yang ingin Anda tetapkan ke proyek Anda. Nama harus unik di dalam ruang Anda.
5. Pilih Buat proyek.

Setelah Anda membuat proyek, langkah selanjutnya dalam proses replikasi adalah membuat satu atau lebih repositori sumber.

## Langkah 3. Membuat repositori sumber di proyek Anda

Dalam proyek yang baru saja Anda buat, Anda perlu membuat repositori sumber. Repositori ini berisi satu file, file README.md, yang dapat Anda edit atau hapus kapan saja. Bergantung pada pilihan yang Anda buat saat membuat repositori sumber, mungkin juga berisi file. `.gitignore`

Untuk membuat repositori sumber

1. Buka konsol CodeCatalyst.
2. Arahkan ke proyek Anda.
3. Di panel navigasi, pilih Kode, lalu pilih Repositori sumber.
4. Pilih Tambahkan repositori, lalu pilih Buat repositori.
5. Dalam nama Repositori, berikan nama untuk repositori.

   Nama repositori harus unik dalam sebuah proyek. Untuk informasi selengkapnya tentang persyaratan untuk nama repositori, lihat Kuota untuk repositori sumber di. CodeCatalyst
6. (Opsional) Dalam Deskripsi, tambahkan deskripsi untuk repositori yang membantu pengguna lain dalam proyek memahami untuk apa repositori digunakan.
7. (Opsional) Tambahkan `.gitignore` file untuk jenis kode yang Anda rencanakan untuk dorong.
8. Pilih Buat.

   > **Note**
   >
   > CodeCatalyst menambahkan `README.md` file ke repositori Anda saat Anda membuatnya. CodeCatalyst juga membuat komit awal untuk repositori di cabang default bernama main. Anda dapat mengedit atau menghapus file README.md, tetapi Anda tidak dapat mengubah atau menghapus cabang default.
9. Untuk mendapatkan URL klon repositori sumber dan PAT, pilih Repositori klon.
10. Untuk menyalin setiap URL klon HTTPS dan PAT, pilih Salin. Kemudian, simpan URL klon dan PAT di suatu tempat di mana Anda dapat mengambilnya.

    URL klon dan PAT akan digunakan pada langkah 4, dan direferensikan sebagai `CODECATALYST_SOURCE_REPO_CLONE_URL` dan. `CODECATALYST_PAT`

Setelah Anda membuat repositori sumber dalam proyek Anda, replikasi AWS Cloud9 data Anda ke repositori sumber ini.

## Langkah 4. Mereplikasi sumber daya AWS Cloud9 kode Anda ke sumber repositori di CodeCatalyst

Jenis repositori sumber yang Anda miliki di AWS Cloud9 lingkungan Anda menentukan metode replikasi yang Anda ikuti untuk mendapatkan sumber daya kode Anda ke dalam repositori CodeCatalyst sumber yang Anda buat. Opsinya adalah sebagai berikut:

- Menggunakan GitHub repositori di AWS Cloud9
- Menggunakan non-GitHub, misalnya GitLab atau Bitbucket, repositori di AWS Cloud9
- Menggunakan repositori kosong di. AWS Cloud9 Opsi ini berarti Anda tidak akan menggunakan repositori sumber apa pun di. AWS Cloud9

### Penggunaan GitHub repositori di CodeCatalyst

Dengan GitHub ekstensi repositori, Anda dapat menggunakan link GitHub repositori dari dalam proyek AWS Cloud9 Amazon CodeCatalyst . Langkah-langkah berikut menguraikan cara menginstal GitHub ekstensi dari CodeCatalyst katalog. Langkah-langkahnya juga menunjukkan cara menghubungkan yang sudah ada GitHub akun ke CodeCatalyst ruang Anda dan tautkan GitHub repositori untuk proyek Anda CodeCatalyst .

Langkah pertama dalam metode ini adalah menginstal GitHub ekstensi repositori dari katalog. CodeCatalyst Untuk menginstal ekstensi, lakukan langkah-langkah berikut.

> **Important**
>
> Sebagai bagian dari menginstal dan mengkonfigurasi Github ekstensi repositori, Anda harus menginstal ekstensi ke GitHub akun. Untuk melakukan ini, Anda harus menjadi GitHub administrator akun dan administrator CodeCatalyst ruang.

#### Langkah 1. Untuk menginstal ekstensi dari CodeCatalyst katalog

1. Buka konsol CodeCatalyst.
2. Arahkan ke ruang Anda.

> **Tip**
>
> Jika Anda memiliki lebih dari satu spasi, Anda dapat memilih ruang mana yang akan dilihat di bilah navigasi atas.

3. Arahkan ke CodeCatalyst katalog dengan memilih ikon Katalog di bilah menu atas di sebelah bilah pencarian. Anda dapat mencari GitHub repositori atau ekstensi filter berdasarkan kategori.
4. (Opsional) Untuk melihat detail lebih lanjut tentang ekstensi, seperti izin yang terkait dengannya, pilih GitHub nama ekstensi repositori.
5. Pilih Instal. Tinjau izin yang diperlukan oleh ekstensi, dan jika Anda ingin melanjutkan, pilih Instal lagi.

Setelah menginstal GitHub ekstensi repositori, Anda dibawa ke GitHub halaman detail ekstensi repositori tempat Anda dapat melihat dan mengelola terhubung GitHub akun dan tertaut GitHub repositori.

Setelah Anda menginstal GitHub ekstensi repositori, sambungkan GitHub akun ke CodeCatalyst ruang Anda. Untuk menghubungkan GitHub akun, lakukan langkah-langkah berikut.

Langkah 2. Untuk menghubungkan GitHub akun ke CodeCatalyst

1. Di Terhubung Github tab akun, pilih Connect GitHub akun untuk pergi ke situs eksternal untuk GitHub.
2. Masuk ke GitHub akun menggunakan GitHub kredensialnya, lalu pilih akun tempat Anda ingin menginstal Amazon. CodeCatalyst
3. Pilih apakah Anda ingin mengizinkan CodeCatalyst untuk mengakses semua repositori saat ini dan masa depan. Atau, sebagai alternatif, pilih yang spesifik GitHub repositori yang ingin Anda gunakan. CodeCatalyst Opsi default adalah semua GitHub repositori di GitHub ruang.
4. Tinjau izin yang diberikan CodeCatalyst, lalu pilih Instal.

Setelah menghubungkan GitHub akun ke CodeCatalyst, Anda dapat melihat akun yang terhubung di GitHub tab akun dari GitHub halaman detail ekstensi repositori.

Langkah terakhir untuk menggunakan GitHub repositori di CodeCatalyst adalah untuk menautkan repositori ke CodeCatalyst proyek tempat Anda ingin menggunakannya. Untuk menautkan GitHub

repositori ke CodeCatalyst proyek, lakukan langkah-langkah berikut yang diuraikan dalam Langkah 3 dari keseluruhan proses:

Langkah 3. Untuk menautkan a GitHub repositori ke CodeCatalyst proyek dari GitHub halaman detail ekstensi repositori

1. Dalam  Tertaut GitHub tab repositori, pilih Tautan  GitHub repositori. .
2. Untuk GitHub akun, pilih GitHub akun yang berisi repositori yang ingin Anda tautkan.
3. Untuk GitHub repositori, pilih repositori yang ingin Anda tautkan ke proyek. CodeCatalyst
4. Untuk CodeCatalyst proyek, pilih CodeCatalyst proyek yang ingin Anda tautkan GitHub repositori ke.
5. Pilih Tautkan.

CodeCatalyst Repositori Anda sekarang harus memiliki file yang diperbarui dan commit yang baru saja Anda dorong. Anda sekarang dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9. Untuk informasi rinci tentang Lingkungan Pengembang, lihat Lingkungan Pengembang di. CodeCatalyst

Anda sekarang dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9. Langkah-langkah untuk melakukan ini diuraikan dalam Langkah 5: Membuat Lingkungan Pengembang menggunakan di AWS Cloud9 CodeCatalyst

Menggunakan non-GitHub repositori di CodeCatalyst

Anda perlu membuat token akses pribadi (PAT) di Amazon CodeCatalyst sebelum mereplikasi lingkungan Anda dari AWS Cloud9 menggunakan non-GitHub repositori. Bagian berikut menguraikan cara membuat token ini.

Membuat token akses pribadi di Amazon CodeCatalyst

Anda dapat mengakses repositori sumber yang Anda buat dalam proyek Anda di komputer lokal Git klien atau dalam lingkungan pengembangan terintegrasi (IDE). Untuk melakukan ini, Anda harus memasukkan kata sandi khusus aplikasi. Anda dapat membuat token akses pribadi (PAT) untuk digunakan untuk tujuan ini. Token akses pribadi (PATs) yang Anda buat dikaitkan dengan identitas pengguna Anda di semua ruang dan proyek CodeCatalyst. Anda dapat melihat nama dan tanggal kedaluwarsa PATs yang Anda buat, dan Anda dapat menghapus Anda PATs yang tidak lagi Anda butuhkan. Anda hanya dapat menyalin rahasia PAT pada saat Anda membuatnya.

Untuk membuat token akses pribadi (PAT)

1. Buka CodeCatalyst konsol di https://codecatalyst.aws/.
2. Di bilah menu atas, pilih lencana profil Anda, lalu pilih Pengaturan saya.

> **Tip**
>
> Anda juga dapat menemukan profil pengguna Anda. Untuk melakukan ini, pada halaman anggota untuk proyek atau spasi, pilih nama Anda dari daftar anggota.

3. Di bawah Token akses pribadi, pilih Buat.
4. Dalam nama PAT, masukkan nama deskriptif untuk token akses pribadi (PAT) Anda.
5. Di Tanggal kedaluwarsa, pertahankan tanggal default atau pilih ikon kalender untuk memilih tanggal kustom. Tanggal kedaluwarsa default menjadi 1 tahun dari tanggal saat ini.
6. Pilih Buat.

> **Tip**
>
> Anda juga dapat membuat token ini ketika Anda memilih Repositori klon untuk repositori sumber.

7. Untuk menyalin rahasia PAT pilih Salin. Simpan rahasia PAT di suatu tempat di mana Anda dapat mengambilnya.

> **Important**
>
> Rahasia PAT hanya ditampilkan sekali. Anda tidak dapat mengambilnya setelah Anda menutup jendela. Jika Anda tidak menyimpan rahasia PAT di lokasi yang aman, Anda dapat membuat yang lain.

Setelah Anda membuat PAT untuk repositori sumber Anda, replikasi data Anda dari lingkungan Anda CodeCatalyst dengan menambahkan repositori jarak jauh di AWS Cloud9 lingkungan Anda dan mendorong data Anda AWS Cloud9 ke repositori ini, seperti yang diuraikan pada bagian di bawah ini.

Menambahkan repositori jarak jauh di lingkungan Anda AWS Cloud9

Katakan bahwa Anda menjalankan repositori yang tidak GitHub repositori. Anda dapat menambahkan repositori jarak jauh di AWS Cloud9 lingkungan Anda dan mendorong data Anda ke repositori sumber Anda. CodeCatalyst Untuk menyelesaikan proses ini, jalankan perintah berikut.

Dari dalam AWS Cloud9 IDE Anda, tambahkan repositori jarak jauh yang menunjuk ke repositori sumber yang Anda buat di langkah 3 dari proses replikasi di. CodeCatalyst Ganti `CODECATALYST_SOURCE_REPO_CLONE_URL` dalam perintah dengan URL Klon yang Anda simpan di langkah 10 Langkah 3. Membuat repositori sumber di proyek Anda.

```
git remote add codecatalyst CODECATALYST_SOURCE_REPO_CLONE_URL
```

Dorong cabang baru ke repositori sumber dengan menggunakan perintah berikut. Ketika diminta untuk memasukkan kata sandi, gunakan `CODECATALYST_PAT` yang Anda simpan di langkah 10 dari Langkah 3. Membuat repositori sumber di proyek Anda:

```
git checkout -b replication && git push codecatalyst replication
```

Berikut ini adalah contoh output command run yang diharapkan:

```
Switched to a new branch 'replication'
Password for 'https://[aws-account-id]@[aws-region].codecatalyst.aws/v1/
MySpace222581768915/Replication/Repository':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 982 bytes | 122.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://[aws-account-id].codecatalyst.aws/v1/MySpace222581768915/Replication/
Repository
* [new branch] replication # replication
```

Cabang ini tersedia di repositori sumber yang Anda buat. CodeCatalyst Anda dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9. Untuk informasi selengkapnya tentang Lingkungan Pengembang, lihat Lingkungan Pengembang di. CodeCatalyst

Anda sekarang dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9. Langkah-langkah untuk melakukan ini diuraikan dalam Langkah 5: Membuat Lingkungan Pengembang menggunakan di AWS Cloud9 CodeCatalyst

Menggunakan repositori kosong di AWS Cloud9

Pertama buat token akses pribadi (PAT) di Amazon CodeCatalyst sebelum Anda dapat mereplikasi lingkungan Anda dari AWS Cloud9 menggunakan repositori kosong. Bagian berikut menguraikan cara membuat token ini.

Membuat token akses pribadi di Amazon CodeCatalyst

Anda dapat mengakses repositori sumber yang Anda buat dalam proyek Anda di komputer lokal Git klien atau dalam lingkungan pengembangan terintegrasi (IDE). Untuk melakukan ini, Anda harus memasukkan kata sandi khusus aplikasi. Anda dapat membuat token akses pribadi (PAT) untuk digunakan untuk tujuan ini. Token akses pribadi (PATs) yang Anda buat dikaitkan dengan identitas pengguna Anda di semua ruang dan proyek CodeCatalyst. Anda dapat melihat nama dan tanggal kedaluwarsa PATs yang Anda buat, dan Anda dapat menghapus Anda PATs yang tidak lagi Anda butuhkan. Anda hanya dapat menyalin rahasia PAT pada saat Anda membuatnya.

Untuk membuat token akses pribadi (PAT)

1. Buka CodeCatalyst konsol di https://codecatalyst.aws/.
2. Di bilah menu atas, pilih lencana profil Anda, lalu pilih Pengaturan saya.

   > **Tip**
   >
   > Anda juga dapat menemukan profil pengguna Anda. Untuk melakukan ini, pada halaman anggota untuk proyek atau spasi, pilih nama Anda dari daftar anggota.

3. Di bawah Token akses pribadi, pilih Buat.
4. Dalam nama PAT, masukkan nama deskriptif untuk token akses pribadi (PAT) Anda.
5. Di Tanggal kedaluwarsa, pertahankan tanggal default atau pilih ikon kalender untuk memilih tanggal kustom. Tanggal kedaluwarsa default menjadi 1 tahun dari tanggal saat ini.
6. Pilih Buat.

> **Tip**
>
> Anda juga dapat membuat token ini ketika Anda memilih Repositori klon untuk repositori sumber.

7. Untuk menyalin rahasia PAT pilih Salin. Simpan rahasia PAT di suatu tempat di mana Anda dapat mengambilnya.

> **Important**
>
> Rahasia PAT hanya ditampilkan sekali. Anda tidak dapat mengambilnya setelah Anda menutup jendela. Jika Anda tidak menyimpan rahasia PAT di lokasi yang aman, Anda dapat membuat yang lain.

Setelah Anda membuat PAT untuk repositori sumber Anda, replikasi data Anda dari lingkungan Anda CodeCatalyst dengan memulai repositori kosong di AWS Cloud9 lingkungan Anda dan menunjuk ke repositori sumber yang Anda AWS Cloud9 buat, seperti yang diuraikan pada bagian di bawah ini. CodeCatalyst

Memulai repositori kosong di AWS Cloud9

Jika Anda tidak memiliki repositori sumber apa pun yang disiapkan AWS Cloud9, maka mulailah repositori kosong di. AWS Cloud9 Selain itu, arahkan ke repositori sumber yang Anda buat CodeCatalyst, dan tambahkan dan dorong file yang ingin Anda tiru Git. Lakukan langkah-langkah berikut dan jalankan perintah berikut untuk mereplikasi AWS Cloud9 file Anda. CodeCatalyst

1. Dari AWS Cloud9 lingkungan Anda, memulai repositori kosong dengan menjalankan perintah berikut:

```
git init -b main
```

   Kemudian, Anda melihat output serupa seperti yang ditunjukkan di bawah ini:

```
Initialized empty Git repository in /home/ec2-user/environment/.git/
```

2. Kloning URL repositori sumber dari. CodeCatalyst Arahkan ke CodeCatalyst proyek yang Anda buat di dalam CodeCatalyst konsol, dan, di panel navigasi, pilih Kode, lalu pilih Repositori sumber.
3. Pilih repositori dari daftar repositori sumber yang Anda inginkan, dan pilih Repositori klon untuk menyalin URL klon.
4. Tambahkan CodeCatalyst repositori menggunakan URL yang Anda kloning, dan dorong konten yang sudah ada di repositori kosong di: CodeCatalyst

```
git remote add origin [...]
git push origin --force
```

5. Tambahkan file yang ingin Anda tiru. Jika Anda ingin mereplikasi semua file di direktori lingkungan Anda, jalankan`git add -A`:

```
git add -A .
git commit -m "replicate"
```

6. Gabungkan dua sejarah yang tidak terkait. Mengatasi konflik gabungan jika terjadi:

```
git merge origin/main --allow-unrelated-histories
```

7. Dorong perubahan kembali ke repositori sumber CodeCatalyst dengan menjalankan perintah berikut. Ketika diminta untuk memasukkan kata sandi, masukkan token akses pribadi (`CODECATALYST_PAT`) yang Anda buat langkah 10 dari Langkah 3. Membuat repositori sumber di proyek Anda:

```
Admin:~/environment (main) $ git push origin main
Password for 'https://222581768915@git.us-west-2.codecatalyst.aws/v1/
MySpace222581768915/Replication/Replication':
```

Setelah menyelesaikan prosedur ini, CodeCatalyst repositori Anda memiliki file yang diperbarui dan komit yang baru saja Anda dorong. Anda sekarang dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9. Langkah-langkah untuk melakukan ini diuraikan di bagian di bawah ini.

## Langkah 5: Membuat Lingkungan Dev menggunakan AWS Cloud9 in CodeCatalyst

Prosedur berikut menguraikan cara membuat Lingkungan Dev dalam CodeCatalyst menggunakan AWS Cloud9 dan data yang baru saja Anda replikasi.

Untuk membuat Lingkungan Dev menggunakan AWS Cloud9

1. Buka CodeCatalyst konsol di https://codecatalyst.aws/.
2. Arahkan ke proyek tempat Anda ingin membuat Lingkungan Dev.
3. Di panel navigasi, Pilih Ikhtisar, lalu arahkan ke bagian Lingkungan Pengembang Saya.
4. Pilih Buat Lingkungan Pengembang.
5. Pilih AWS Cloud9 dari menu drop-down.
6. Pilih Kloning repositori.

> **Note**
>
> Saat ini, CodeCatalyst tidak mendukung kloning repositori pihak ketiga, tetapi Anda dapat membuat Lingkungan Dev dan mengkloning repositori pihak ketiga ke dalamnya dari IDE pilihan Anda.

7. Lakukan salah satu hal berikut ini:
   a. Pilih repositori yang akan dikloning, pilih Bekerja di cabang yang ada, lalu pilih cabang dari menu drop-down cabang yang ada.
   b. Pilih repositori untuk dikloning, pilih Bekerja di cabang baru, masukkan nama cabang ke bidang Nama cabang, dan pilih cabang untuk membuat cabang baru dari Buat cabang dari menu drop-down.
8. Secara opsional, tambahkan alias untuk Lingkungan Dev.
9. Secara opsional, pilih tombol edit konfigurasi Lingkungan Dev untuk mengedit konfigurasi komputasi, penyimpanan, atau batas waktu Dev Environment.
10. Pilih Buat. Saat Dev Environment Anda sedang dibuat, kolom status Dev Environment akan menampilkan Mulai, dan kolom status akan ditampilkan Running setelah Dev Environment telah dibuat.

## Menggunakan alat replikasi

AWS Cloud9 in CodeCatalyst memberikan pengalaman yang dikelola sepenuhnya untuk berinteraksi dengan AWS Cloud9. Untuk memungkinkan pelanggan mencoba menggunakannya AWS Cloud9 CodeCatalyst, kami telah membuat alat replikasi. Setelah Anda menyalin dan menjalankan skrip di AWS Cloud9 lingkungan Anda, ikuti petunjuk untuk menjalankannya dan mereplikasi sumber daya

kode Anda dari AWS Cloud9 ke. CodeCatalyst Untuk informasi lebih lanjut tentang alat dan proses replikasi, lihat FAQ tentang proses replikasi yang diuraikan di bawah ini.

> **Note**
>
> Proses replikasi ini tidak akan berpengaruh pada AWS Cloud9 lingkungan Anda yang ada. Setelah proses replikasi selesai, Anda dapat menghapus Lingkungan Dev, repositori sumber, proyek dan ruang, dan itu tidak akan berpengaruh pada lingkungan Anda. AWS Cloud9 Alat ini hanya akan menyalin sumber daya kode Anda ke AWS Cloud9 dalam CodeCatalyst, itu tidak akan menghapus atau mengkonfigurasi AWS Cloud9 lingkungan Anda yang ada. Alat replikasi ini telah dirilis ke grup AWS akun pilihan awal. Akibatnya, mungkin tidak muncul di AWS akun tertentu.

> **Note**
>
> Disarankan agar Anda mendaftar ke Amazon CodeCatalyst dan membuat ruang sebelum Anda mengunduh alat. Untuk informasi tentang mendaftar CodeCatalyst, lihat Mendaftar ke Amazon CodeCatalyst dan membuat Spasi.

## Manfaat menggunakan AWS Cloud9 di Amazon CodeCatalyst

Bagian berikut menguraikan beberapa manfaat kinerja dan fitur yang disempurnakan yang akan Anda alami saat menggunakan AWS Cloud9 : CodeCatalyst

- CodeCatalyst memberikan pengalaman terintegrasi yang memungkinkan Anda menggunakan Lingkungan Dev yang dikelola sepenuhnya untuk mengelola seluruh siklus hidup pengembangan perangkat lunak dari satu lokasi.
- Opsi ukuran volume Amazon EBS yang disempurnakan saat diluncurkan.
- Support untuk lingkungan sementara dan kemampuan untuk menskalakan komputasi Lingkungan Dev Anda sesuai permintaan.
- Dukungan AMI khusus yang tersedia melalui spesifikasi gambar khusus.
- Dukungan Devfile yang memungkinkan Anda untuk menggambarkan konfigurasi sebagai kode.

## Mereplikasi sumber daya AWS Cloud9 kode Anda dalam CodeCatalyst menggunakan alat replikasi

Prosedur berikut merinci cara menyalin dan menjalankan alat replikasi untuk menyelesaikan proses replikasi.

1. Salin skrip di bawah ini dan pastikan Anda menjalankannya dalam AWS Cloud9 lingkungan:

   ```
   curl https://dx5z5embsyrja.cloudfront.net -o /tmp/replicate-tool.tar.gz && tar
    --no-same-owner --no-same-permissions -xvf /tmp/replicate-tool.tar.gz -C /tmp &&
    node /tmp/cloud9-replication-tools
   ```

2. [Opsional] Alat replikasi menggunakan Akun AWS ID Anda untuk telemetri. Tujuannya adalah untuk membantu kami mengidentifikasi masalah yang mungkin Anda temui dengan lebih baik saat menggunakan alat ini. Kami memancarkan acara telemetri untuk`tool starts`,,`tool fails`, `tool is cancelled by user` dan. `tool completes successfully` `tool creates a Dev Environment for the user` Jika Anda ingin menonaktifkan telemetri dengan alat replikasi, lihat Menonaktifkan telemetri untuk alat replikasi di bawah ini.
3. Setelah Anda menyalin dan menjalankan alat replikasi di AWS Cloud9 lingkungan Anda, Anda harus menautkan Anda Akun AWS dengan AWS Builder ID dengan menavigasi ke URL akses di browser dan mengklik Izinkan dalam waktu 10 menit. Pastikan Anda hanya membuka tautan sekali, jika Anda membukanya beberapa kali itu akan menyebabkan kesalahan dan Anda harus memulai lagi. Untuk informasi selengkapnya tentang AWS Builder ID, lihat Masuk dengan AWS Builder ID di Panduan Pengguna AWS Masuk. Ini akan memberikan akses alat replikasi ke sumber daya kode Anda untuk tujuan mereplikasi mereka. CodeCatalyst
4. Pilih Space yang ingin Anda gunakan. Jika Anda hanya memiliki satu ruang, ruang itu dipilih. Untuk informasi selengkapnya tentang spasi, lihat Spasi CodeCatalyst di Panduan CodeCatalyst Pengguna Amazon.
5. Pilih apakah Anda ingin mereplikasi kode Anda CodeCatalyst atau mencobanya dengan Lingkungan Dev baru. Kami merekomendasikan untuk mereplikasi kode Anda secara langsung di CodeCatalyst. Untuk informasi selengkapnya tentang Lingkungan Pengembang, lihat Lingkungan Pengembang CodeCatalyst di CodeCatalyst Panduan Pengguna Amazon.
6. Masukkan nama untuk proyek Anda atau tekan enter untuk menggunakan nama default yang disediakan.
7. Saat diminta, pilih bagaimana Anda ingin menyalin file Anda ke repositori sumber baru di. CodeCatalyst Anda dapat memilih untuk mendorong folder root ke satu CodeCatalyst repositori, atau mendorong sub-folder Anda ke repositori yang berbeda. CodeCatalyst

8. Setelah alat selesai, navigasikan ke proyek di dalam CodeCatalyst konsol melalui URL yang disediakan di pesan terminal untuk mengakses sumber daya kode Anda CodeCatalyst.

Setelah menyelesaikan prosedur ini, CodeCatalyst repositori Anda memiliki file yang diperbarui dan komit yang baru saja Anda dorong. Anda sekarang dapat membuat Lingkungan Dev dari cabang ini dan membukanya dengan AWS Cloud9.

## Menonaktifkan telemetri untuk alat replikasi

Langkah-langkah berikut menguraikan cara mengatur variabel lingkungan untuk menonaktifkan telemetri untuk alat replikasi.

1. Buka terminal di AWS Cloud9 lingkungan Anda
2. Jalankan salah satu dari perintah berikut:

   ```
   export CLOUD9_REPLICATION_TOOL_TELEMETRY=off
   ```

   atau

   ```
   export CLOUD9_REPLICATION_TOOL_TELEMETRY=0
   ```

3. Setelah Anda menjalankan salah satu perintah di atas, variabel lingkungan akan diatur dan telemetri untuk alat replikasi akan dinonaktifkan. Setelah Anda menonaktifkan telemetri, Anda harus menyalin dan menjalankan kembali skrip alat replikasi lagi untuk memulai proses.

## Umpan balik alat replikasi

Jika Anda mengalami masalah, atau ingin memberikan umpan balik tentang pengalaman Anda menggunakan alat replikasi, silakan buat dan kirimkan kasus dukungan. Untuk informasi tentang cara membuat kasus dukungan, lihat Membuat kasus dukungan dan manajemen kasus.

## Perbedaan antara AWS Cloud9 dan Amazon CodeCatalyst

Tabel berikut menguraikan beberapa perbedaan antara AWS Cloud9 dan AWS Cloud9 seterusnya CodeCatalyst.

| AWS Cloud9 | AWS Cloud9 di Amazon CodeCatalyst |
| --- | --- |
| VPC pribadi bekerja sangat baik dengan. AWS Cloud9 | Penggunaan VPC pribadi saat ini tidak didukung untuk AWS Cloud9 on. CodeCatalyst |
| AWS Cloud9 mendukung kredensi AWS terkelola yang telah dikonfigurasi sebelumnya. | Kredenal perlu dikonfigurasi secara manual untuk AWS Cloud9 on. CodeCatalyst |
| Dimungkinkan untuk memiliki interval 30 menit hingga 7 hari dan untuk menonaktifkan shutdown dengan. AWS Cloud9 | Dimungkinkan untuk memiliki interval 15 menit hingga 20 jam untuk AWS Cloud9 aktif CodeCatalyst dan Anda tidak dapat menonakti fkan shutdown. |
| AWS Cloud9 mendukung platform Ubuntu dan AL2 OS. | AWS Cloud9 on CodeCatalyst mendukung gambar MDE Universal dan gambar khusus yang dapat mencakup Ubuntu dan AL2. Untuk informasi selengkapnya tentang ini, lihat Gambar devfile universal di Panduan CodeCatalyst Pengguna Amazon. |
| Mengunggah dan mengunduh didukung di AWS Cloud9 | Mengunggah dan mengunduh saat ini tidak didukung untuk AWS Cloud9 aktif. CodeCatal yst Pengguna harus mengunggah dan mengunduh menggunakan bucket Amazon S3. |
| Kolaborasi tersedia di AWS Cloud9 | Kolaborasi saat ini tidak tersedia untuk AWS Cloud9 di CodeCatalyst. |

## FAQs tentang proses replikasi

Bagian berikut bertujuan untuk menjawab beberapa yang FAQs terkait dengan alat replikasi, dan proses replikasi.

Pertanyaan: Jika saya mereplikasi AWS Cloud9 lingkungan saya CodeCatalyst, apakah AWS Cloud9 lingkungan saya akan terpengaruh?

Jawaban: Tidak, replikasi lingkungan Anda hanya akan menyalin sumber daya kode Anda untuk AWS Cloud9 CodeCatalyst memungkinkan Anda untuk terus bekerja. Sumber daya kode dan lingkungan Anda tidak AWS Cloud9 akan terpengaruh dengan cara apa pun.

Pertanyaan: Jika saya ingin melakukan rollback, apakah AWS Cloud9 lingkungan saya akan terpengaruh?

Jawaban: Tidak, Anda dapat menghapus Lingkungan CodeCatalyst Dev, repositori sumber, proyek dan ruang dan itu tidak akan berpengaruh pada lingkungan Anda. AWS Cloud9

Pertanyaan: Apakah lokasi baru akan sesuai dengan standar seperti HIPAA, SOC, dll?

Jawaban: Lingkungan Dev on saat CodeCatalyst ini tidak sesuai dengan standar ini. Kepatuhan terhadap standar ini ada di peta jalan.

Pertanyaan: Ke mana sumber daya kode saya akan pergi?

Jawaban: Sumber daya kode Anda akan disalin ke repositori sumber dalam proyek Anda di. CodeCatalyst

Pertanyaan: Apakah penggunaan saya akan terbatas?

Jawaban: Sebagai bagian dari proses replikasi, Anda akan membuat Lingkungan Dev yang memiliki 16 GB dalam tingkat gratis. Ini berarti Anda dapat memiliki maksimal 4 Lingkungan Pengembang. Untuk informasi selengkapnya tentang harga, penyimpanan, dan berbagai tingkatan yang tersedia, lihat Amazon CodeCatalyst - Harga.

Pertanyaan: Ke mana komputasi saya akan pergi?

Jawaban: Tidak akan ada perubahan pada komputasi Anda yang ada. Itu akan tetap seperti apa adanya.

Pertanyaan: Dapatkah saya menggunakan kredensi AWS akun saya yang ada di CodeCatalyst, dan apakah kredensyal tersebut akan ditransfer secara otomatis?

Jawaban: Anda dapat mengonfigurasi kredensi AWS akun Anda secara manual di. CodeCatalyst Mereka tidak akan ditransfer secara otomatis.

Pertanyaan: Berapa biayanya?

Jawaban: Anda dapat mulai menggunakan secara CodeCatalyst gratis. Untuk informasi selengkapnya tentang harga dan berbagai tingkatan yang tersedia, lihat Amazon CodeCatalyst - Harga.

Pertanyaan: Apakah proses replikasi data dan penyimpanan data CodeCatalyst aman?

Jawaban: Ya, kami akan menggunakan git push dengan https to copy sumber daya kode dan menyimpan data dengan CodeCatalyst aman di dalam layanan. Semua data dienkripsi dalam perjalanan dan saat istirahat. Untuk informasi selengkapnya tentang perlindungan data di CodeCatalyst, lihat Perlindungan data CodeCatalyst di Amazon di Panduan CodeCatalyst Pengguna Amazon.

Pertanyaan: Pendekatan replikasi mana yang harus saya pilih?

Jawaban: Alat replikasi menawarkan dua pendekatan; Anda dapat menyalin sumber daya kode Anda dari AWS Cloud9 ke CodeCatalyst dengan mendorongnya ke repositori CodeCatalyst sumber tunggal, atau setiap subfolder diterjemahkan ke dalam repositori sumber yang berbeda. CodeCatalyst Kami merekomendasikan menggunakan pendekatan pertama karena tidak memerlukan pengetahuan sebelumnya tentang CodeCatalyst konsep seperti repositori sumber. Pendekatan ini adalah titik awal yang baik untuk mengeksplorasi AWS Cloud9 pengalaman CodeCatalyst, sambil bekerja dengan pengaturan serupa yang biasa Anda gunakan AWS Cloud9.

Opsi kedua paling baik dipilih ketika Anda menggunakan subfolder yang terletak di bawah folder AWS Cloud9 lingkungan root secara independen. Dengan pendekatan ini, file apa pun di bawah folder root tidak akan direplikasi. Untuk informasi selengkapnya tentang repositori sumber di CodeCatalyst, lihat Repositori sumber di dalam Panduan Pengguna CodeCatalyst Amazon. CodeCatalyst

Pertanyaan: Apa token akses pribadi yang dihasilkan dalam proses replikasi dan mengapa saya membutuhkannya? Bisakah saya membuatnya lagi jika saya kehilangannya?

Jawaban: Token akses pribadi dikaitkan dengan identitas pengguna Anda di CodeCatalyst. Ini diperlukan sebagai kata sandi saat Anda mendorong perubahan lokal dengan git ke repositori CodeCatalyst sumber. Untuk informasi selengkapnya tentang token dan cara membuatnya, lihat Mengelola token akses pribadi di Amazon CodeCatalyst di Panduan CodeCatalyst Pengguna Amazon.

Pertanyaan: Apa yang terjadi jika ada kesalahan selama proses replikasi?

Jawaban: Jika terjadi kesalahan saat menggunakan alat replikasi, Anda harus terlebih dahulu mencoba alat ini lagi. Jika kesalahan terkait dengan repositori sumber, Anda dapat secara manual mendorong sumber daya kode Anda ke repositori CodeCatalyst sumber setelah direplikasi. Ini seharusnya berfungsi karena repositori lokal telah dikonfigurasi untuk bekerja dengan CodeCatalyst

upstream. Jika masalah berlanjut, harap buat dan kirimkan kasus dukungan. Untuk informasi tentang cara membuat kasus dukungan, lihat Membuat kasus dukungan dan manajemen kasus.

Pertanyaan: Mengapa saya perlu mengautentikasi dan memberikan izin ke alat replikasi menggunakan builderId saya? AWS

Jawaban: Selama proses replikasi, alat replikasi perlu membaca dan menulis beberapa sumber daya (proyek, Lingkungan Pengembang, repositori sumber) CodeCatalyst dan menyalin konten lokal atas nama pengguna, sehingga memerlukan izin Anda untuk melakukan ini.

Pertanyaan: Apakah akan ada perubahan latensi jika saya pindah ke? CodeCatalyst

Jawaban: Tergantung pada tindakan yang Anda lakukan, Anda mungkin melihat pengurangan latensi. Ini karena CodeCatalyst server yang di-host di wilayah PDX.

Pertanyaan: Apakah semua perangkat lunak yang saya instal akan ditransfer?

Jawaban: Tidak, hanya sumber daya kode Anda yang akan ditransfer. Binari, konfigurasi, dan perangkat lunak yang diinstal tidak akan ditransfer.

## Lingkungan Pengembang di Amazon CodeCatalyst

Bagian berikut menguraikan cara membuat dan mengelola Lingkungan Dev Anda dengan CodeCatalyst menggunakan IDE. AWS Cloud9

- Membuat Lingkungan Dev
- Membuka pengaturan Lingkungan Dev
- Melanjutkan Lingkungan Pengembang
- Menghapus Lingkungan Dev
- Mengedit devfile repositori untuk Lingkungan Dev
- Mengkloning repositori
- Memecahkan Masalah Lingkungan Pengembang

### Membuat Lingkungan Dev

Anda dapat membuat Lingkungan Pengembang dengan berbagai cara:

- Buat Lingkungan Pengembang CodeCatalyst dengan repositori CodeCatalyst sumber dari halaman repositori Ringkasan, Lingkungan Pengembang, atau Sumber.

- Buat Lingkungan Dev kosong yang tidak terhubung ke repositori sumber CodeCatalyst dari Dev Environments.
- Buat Lingkungan Pengembang di IDE pilihan Anda dan kloning repositori CodeCatalyst sumber ke Lingkungan Dev.

Anda dapat membuat satu Lingkungan Dev untuk setiap cabang dan repositori. Sebuah proyek dapat memiliki beberapa repositori. Lingkungan Pengembang Anda hanya terkait dengan CodeCatalyst akun Anda, dan hanya dapat dikelola oleh CodeCatalyst akun Anda. Anda dapat membuka Lingkungan Dev dan bekerja dengannya dengan salah satu yang didukung IDEs. Setelah Anda memilih IDE tertentu, Anda hanya dapat membuka Lingkungan Dev dengan IDE yang dipilih. Jika Anda ingin menggunakan IDE yang berbeda, Anda dapat mengubah IDE dengan memilih Lingkungan Dev di bilah navigasi dan memilih Edit, atau dengan membuat Lingkungan Dev baru. Secara default, Dev Environments dibuat dengan prosesor 2-core, RAM 4 GB, dan penyimpanan persisten 16 GB.

Untuk informasi selengkapnya tentang cara membuat Lingkungan Pengembang di CodeCatalyst, lihat Membuat Lingkungan Pengembang di panduan Amazon CodeCatalyst .

Untuk informasi dan langkah-langkah dalam membuat Lingkungan Pengembang di CodeCatalyst, lihat Membuat Lingkungan Pengembang di  CodeCatalyst Panduan Pengguna Amazon.

> **Note**
>
> Anda sekarang dapat membuat Lingkungan Dev dengan repositori sumber pihak ketiga. Untuk informasi tentang menautkan repositori sumber pihak ketiga ke project di dalamnya CodeCatalyst, lihat Menautkan repositori sumber di Panduan Pengguna Amazon. CodeCatalyst

## Membuka pengaturan Lingkungan Dev

Setelah Anda membuat Lingkungan Dev di CodeCatalyst konsol, Anda dapat melihat pengaturan Lingkungan Dev tertentu:

1. Di CodeCatalyst konsol, navigasikan ke Lingkungan Dev Anda melalui AWS Cloud9 IDE.
2. Pilih aws-explorer dari sidebar. AWS Cloud9
3. Di panel navigasi Alat Pengembang, perluas CodeCatalystdan pilih Buka Pengaturan untuk membuka tampilan Pengaturan Lingkungan Dev.

4. Dari tampilan Pengaturan Lingkungan Dev, bagian berikut berisi opsi untuk Lingkungan Pengembang Anda:

   - Alias: Lihat dan ubah Alias yang ditetapkan ke Lingkungan Pengembang Anda.
   - Status: Lihat status Lingkungan Pengembang Anda saat ini, proyek yang telah ditetapkan, dan hentikan Lingkungan Pengembang Anda.
   - Devfile: Lihat nama dan lokasi Devfile untuk Lingkungan Dev Anda. Buka Anda Devfile dengan memilih Buka di Editor.
   - Pengaturan Komputasi: Ubah ukuran dan Panjang Batas Waktu default untuk Lingkungan Pengembang Anda.

> **Note**
>
> Anda tidak dapat mengubah jumlah ruang penyimpanan yang ditetapkan ke Lingkungan Pengembang Anda setelah dibuat.

> **Note**
>
> Saat menggunakan Amazon CodeCatalyst AWS CLI dari terminal, Anda harus memastikan bahwa Anda menyetel AWS_PROFILE=codecatalyst sebelum menjalankan perintah apa pun. CodeCatalyst

## Melanjutkan Lingkungan Pengembang

Segala sesuatu di `$HOME` direktori Dev Environment disimpan terus-menerus. Anda dapat berhenti bekerja di Lingkungan Pengembang jika perlu, dan melanjutkan bekerja di Lingkungan Pengembang Anda di lain waktu. Misalkan Lingkungan Dev dibiarkan menganggur selama lebih dari jumlah waktu yang dipilih di bidang Timeout saat Lingkungan Dev dibuat. Dalam hal ini, sesi secara otomatis berhenti.

Anda hanya dapat melanjutkan Lingkungan Pengembang dari CodeCatalyst. Untuk informasi selengkapnya tentang cara melanjutkan Lingkungan Pengembang, lihat Melanjutkan Lingkungan Pengembang di panduan Amazon. CodeCatalyst

> **Note**
>
> Melanjutkan Lingkungan Pengembang mungkin memakan waktu beberapa menit.

## Menghapus Lingkungan Dev

Setelah selesai mengerjakan konten yang disimpan di Lingkungan Pengembang, Anda dapat menghapus konten tersebut. Sebelum Anda menghapus Lingkungan Dev, pastikan Anda melakukan dan mendorong perubahan kode Anda ke repositori sumber asli. Setelah Anda menghapus Lingkungan Dev, tagihan komputasi dan penyimpanan untuk Lingkungan Pengembang berakhir.

Anda hanya dapat menghapus Dev Environment dari halaman Dev Environments di. CodeCatalyst Untuk informasi selengkapnya tentang cara menghapus Lingkungan Pengembang, lihat Menghapus Lingkungan Pengembang di panduan Amazon. CodeCatalyst

## Mengedit repositori devfile untuk Lingkungan Pengembang

Untuk mengubah konfigurasi Lingkungan Pengembang, edit devfile. Anda dapat menggunakandevfiles untuk menstandarisasi Lingkungan Pengembang pengembangan Anda di seluruh tim Anda. Anda dapat mengedit devfile dari akar repositori sumber di. CodeCatalyst Atau, Anda dapat mengedit devfile dalam IDE yang didukung. Jika Anda mengedit devfile dalam IDE yang didukung, komit dan dorong perubahan Anda ke repositori sumber atau buat permintaan tarik. Dengan begitu, anggota tim dapat meninjau dan menyetujui devfile suntingan.

> **Note**
>
> Anda hanya dapat menyertakan gambar kontainer publik di devfile.

> **Note**
>
> Jika dependensi hilang, beberapa fitur AWS Cloud9 IDE mungkin tidak berfungsi di kustom devfile. Mungkin diperlukan upaya tambahan untuk membuatnya bekerja di beberapa platform selain Linux x64.

Untuk mengedit repositori devfile untuk Lingkungan Pengembang di AWS Cloud9

1. Di CodeCatalyst konsol, navigasikan ke Lingkungan Dev Anda melalui AWS Cloud9 IDE.
2. Dari AWS Cloud9 sidebar, pilih aws-explorer.
3. Di panel navigasi Alat Pengembang, pilih menu CodeCatalyst toolkit.
4. Pilih Buka Devfile.
5. Edit devfile, dan simpan filenya.
6. Pilih Source Control, yang merupakan Git ekstensi dari menu side-bar.
7. Di bidang Teks pesan, masukkan pesan sebelum melakukan perubahan pementasan.
8. Untuk mempersiapkan komit, pilih ikon Tahap Semua Perubahan (+).
9. Untuk melihat Git perintah, pilih ikon menu yang ada di sebelah nama repositori.
10. Pilih Commit dan Push.
11. Pilih Perbarui Lingkungan Pengembang dari AWS Toolkit menu.

    Pilih Commit dan Push. Yang diperbarui devfile telah disimpan dan perubahan telah dilakukan dan didorong.

> **Note**
>
> Katakan bahwa Lingkungan Pengembang yang ingin Anda luncurkan menggunakan kustom devfile tidak bekerja. Ini mungkin karena devfile tidak kompatibel dengan AWS Cloud9. Untuk memecahkan masalah, tinjau devfile. Jika masalah berlanjut, hapus dan coba buat yang baru.

Anda juga dapat mengedit devfile untuk Lingkungan Pengembang melalui CodeCatalyst. Untuk informasi selengkapnya, lihat Mengonfigurasi Lingkungan Pengembang Anda di panduan Amazon CodeCatalyst .

## Mengkloning repositori

Untuk bekerja secara efektif dengan banyak file, cabang, dan komit di repositori sumber, Anda dapat mengkloning repositori sumber ke komputer lokal Anda. Kemudian, gunakan Git klien atau IDE untuk membuat perubahan. Dari CodeCatalyst, Anda dapat menggunakan AWS Cloud9 IDE Gitekstensi dengan cara yang sama seperti yang lainnya Git penyedia host dan juga dengan menggunakan baris

perintah. Untuk mempelajari cara mengkloning repositori pihak ketiga, lihat Menginisialisasi atau mengkloning Git repositori.

Untuk informasi selengkapnya tentang membuat Lingkungan Pengembang dari repositori sumber dan mengkloningnya CodeCatalyst, lihat Konsep repositori sumber di panduan Amazon. CodeCatalyst

### Memecahkan Masalah Lingkungan Pengembang

Jika Anda mengalami masalah dengan Lingkungan Pengembang, lihat Memecahkan masalah dengan Lingkungan Pengembang di panduan Amazon. CodeCatalyst

> **Note**
>
> Saat menggunakan Amazon CodeCatalyst AWS CLI dari terminal, Anda harus memastikan bahwa Anda menyetel AWS_PROFILE=codecatalyst sebelum menjalankan perintah apa pun. CodeCatalyst

Jika Anda mengalami masalah dengan Lingkungan Pengembang, lihat Memecahkan masalah dengan Lingkungan Pengembang di panduan Amazon. CodeCatalyst

## Bekerja dengan AWS CDK di AWS Cloud9 IDE

AWS CDK Layanan ini memungkinkan Anda untuk bekerja dengan AWS Cloud Development Kit (AWS CDK)aplikasi, atau aplikasi. Anda dapat menemukan informasi rinci tentang AWS CDK di Panduan AWS Cloud Development Kit (AWS CDK) Pengembang.

AWS CDK aplikasi terdiri dari blok bangunan yang dikenal sebagai konstruksi. Blok bangunan ini mencakup definisi untuk AWS CloudFormation tumpukan Anda dan AWS sumber daya di dalamnya. Menggunakan AWS CDK Explorer, Anda dapat melihat tumpukan dan sumber daya yang ditentukan dalam tampilan AWS CDK pohon. Anda dapat mengakses tampilan ini di panel Alat Pengembang di dalam AWS Cloud9 editor.

Bagian ini memberikan informasi tentang cara mengakses dan menggunakan AWS CDK di AWS Cloud9 editor.

# Bekerja dengan AWS CDK aplikasi

Gunakan AWS CDK Explorer di lingkungan pengembangan AWS Cloud9 terintegrasi (IDE) untuk memvisualisasikan dan bekerja dengan AWS CDK aplikasi.

## Prasyarat

Instal antarmuka baris AWS CDK perintah. Untuk petunjuk, lihat Memulai AWS CDK di Panduan AWS Cloud Development Kit (AWS CDK) Pengembang.

> **Important**
>
> AWS CDK Versi yang Anda instal harus 1.17.0 atau yang lebih baru. Anda dapat memeriksa versi mana yang Anda jalankan menggunakan **`cdk --version`**perintah.

## Visualisasikan aplikasi AWS CDK

Menggunakan AWS Cloud9 IDE AWS CDK Explorer, Anda dapat mengelola tumpukan dan sumber daya yang disimpan dalam konstruksi CDK aplikasi Anda. AWS CDK Explorer menampilkan sumber daya Anda dalam tampilan pohon menggunakan informasi yang ditentukan dalam `tree.json` file. File ini dibuat saat Anda menjalankan **`cdk synth`**perintah. Secara default, `tree.json` file tersebut terletak di `cdk.out` direktori aplikasi.

Untuk mulai menggunakan Toolkit AWS CDK Explorer, buat aplikasi CDK.

1. Selesaikan beberapa langkah pertama dari Hello World Tutorial di Panduan AWS CDK Pengembang.

   > **Important**
   >
   > Saat Anda mencapai langkah Deploying the Stack, berhenti dan kembali ke panduan ini.

   > **Note**
   >
   > Anda dapat menjalankan perintah yang disediakan dalam tutorial, seperti **`mkdir`**dan **`cdk init`**, pada antarmuka baris perintah sistem operasi atau di jendela Terminal di dalam editor VS Code.

2. Setelah Anda menyelesaikan langkah-langkah yang diperlukan dari tutorial CDK, buka konten CDK yang Anda buat di editor AWS Cloud9 IDE.
3. Di panel AWS navigasi, perluas judul CDK. Aplikasi CDK Anda dan sumber daya terkaitnya sekarang ditampilkan di tampilan pohon CDK Explorer. Anda juga dapat menjalankan perintah berikut di terminal di dalam AWS Cloud9 untuk mengonfirmasi bahwa fitur CDK berfungsi:

```
mkdir mycdkapp
cd mycdkapp
cdk init app --language=typescript
cdk synth
cdk bootstrap
```

Catatan penting

- Saat memuat aplikasi CDK ke AWS Cloud9 editor, Anda dapat memuat beberapa folder sekaligus. Setiap folder dapat berisi beberapa aplikasi CDK, seperti yang ditunjukkan pada gambar sebelumnya. AWS CDK Explorer menemukan aplikasi di direktori root proyek dan subdirektori langsungnya.

- Ketika Anda melakukan beberapa langkah pertama dari tutorial, Anda mungkin memperhatikan bahwa perintah terakhir yang Anda jalankan adalah **`cdk synth`**. Perintah ini mensintesis CloudFormation template dengan menerjemahkan AWS CDK aplikasi Anda ke CFN. Sebagai produk sampingan, itu juga menghasilkan `tree.json` file. Jika Anda membuat perubahan pada aplikasi CDK, jalankan **`cdk synth`**perintah lagi untuk melihat perubahan yang tercermin dalam tampilan pohon. Salah satu contoh perubahan adalah menambahkan lebih banyak sumber daya ke aplikasi.

## Melakukan operasi lain di AWS CDK aplikasi

Anda dapat menggunakan AWS Cloud9 editor untuk melakukan operasi lain pada aplikasi CDK dengan cara yang sama seperti Anda menggunakan antarmuka baris perintah. Misalnya, Anda dapat memperbarui file kode di editor dan menerapkan aplikasi dengan menggunakan jendela AWS Cloud9 Terminal.

Untuk mencoba jenis tindakan ini, gunakan AWS Cloud9 editor untuk melanjutkan Hello World Tutorial di Panduan AWS CDK Pengembang. Pastikan Anda melakukan langkah terakhir,

Menghancurkan Sumber Daya Aplikasi. Jika tidak, Anda mungkin dikenakan biaya tak terduga untuk Anda Akun AWS.

# Kontrol sumber visual dengan panel Git

Panel Git untuk AWS Cloud9 menyediakan antarmuka visual yang nyaman untuk menggunakan fitur Git penting.

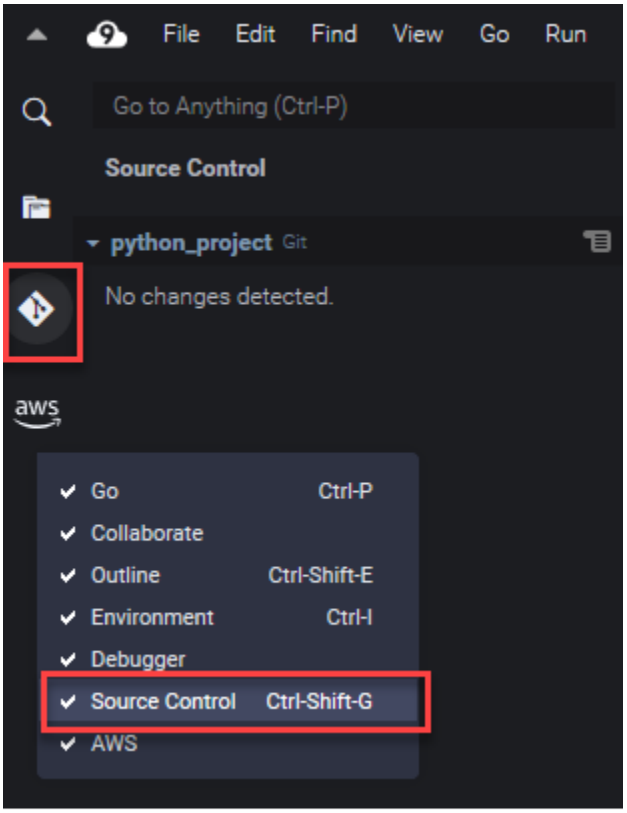
Dengan menggunakan opsi dari antarmuka panel Git, Anda dapat mengelola siklus hidup kontrol sumber lengkap: menginisialisasi repositori atau mengkloning repo jarak jauh, menambahkan file ke area persiapan, berkomitmen terhadap file persiapan ke direktori kerja, lalu mendorong perubahan ke repositori upstream.

Kolaborasi inti dan fitur manajemen proyek Git, seperti membuat dan menggabungkan cabang, dapat dengan cepat diimplementasikan dengan beberapa klik di antarmuka panel Git. Selain itu, konflik gabungan dapat diidentifikasi dan diselesaikan menggunakan jendela editor IDE.

> **Important**
>
> Panel Git hanya tersedia di AWS Cloud9 lingkungan yang dibuat dengan EC2 instance Amazon. Fitur ini tidak dapat diakses jika Anda menggunakan lingkungan pengembangan SSH, bukan EC2 lingkungan.
>
> Selain itu, panel Git tersedia secara default hanya di AWS Cloud9 lingkungan baru yang dibuat setelah 11 Desember 2020. Kami sedang berupaya mengaktifkan panel Git untuk lingkungan pengembangan yang dibuat sebelum tanggal ini.

Untuk mengakses dan berinteraksi dengan antarmuka, pilih Window, Kontrol Sumber. Atau, Anda bisa masuk ke Source Control dengan mengklik kanan di mana saja di panel samping IDE dan memilih Source Control. Kemudian, setelah ini, pilih ikon Git yang ditampilkan di antarmuka IDE.

Kombinasi tombol Ctrl-Shift-G juga dapat digunakan untuk beralih tampilan panel Git.




> **Note**
>
> Tangkapan layar untuk dokumentasi panel Git menunjukkan AWS Cloud9 IDE dengan tema Jett Dark diterapkan. Beberapa elemen antarmuka ditampilkan secara berbeda jika Anda menggunakan IDE dengan tema yang berbeda. Untuk membuka panel Git, Anda dapat memilih tautan dengan label Kontrol Sumber alih-alih ikon Git.

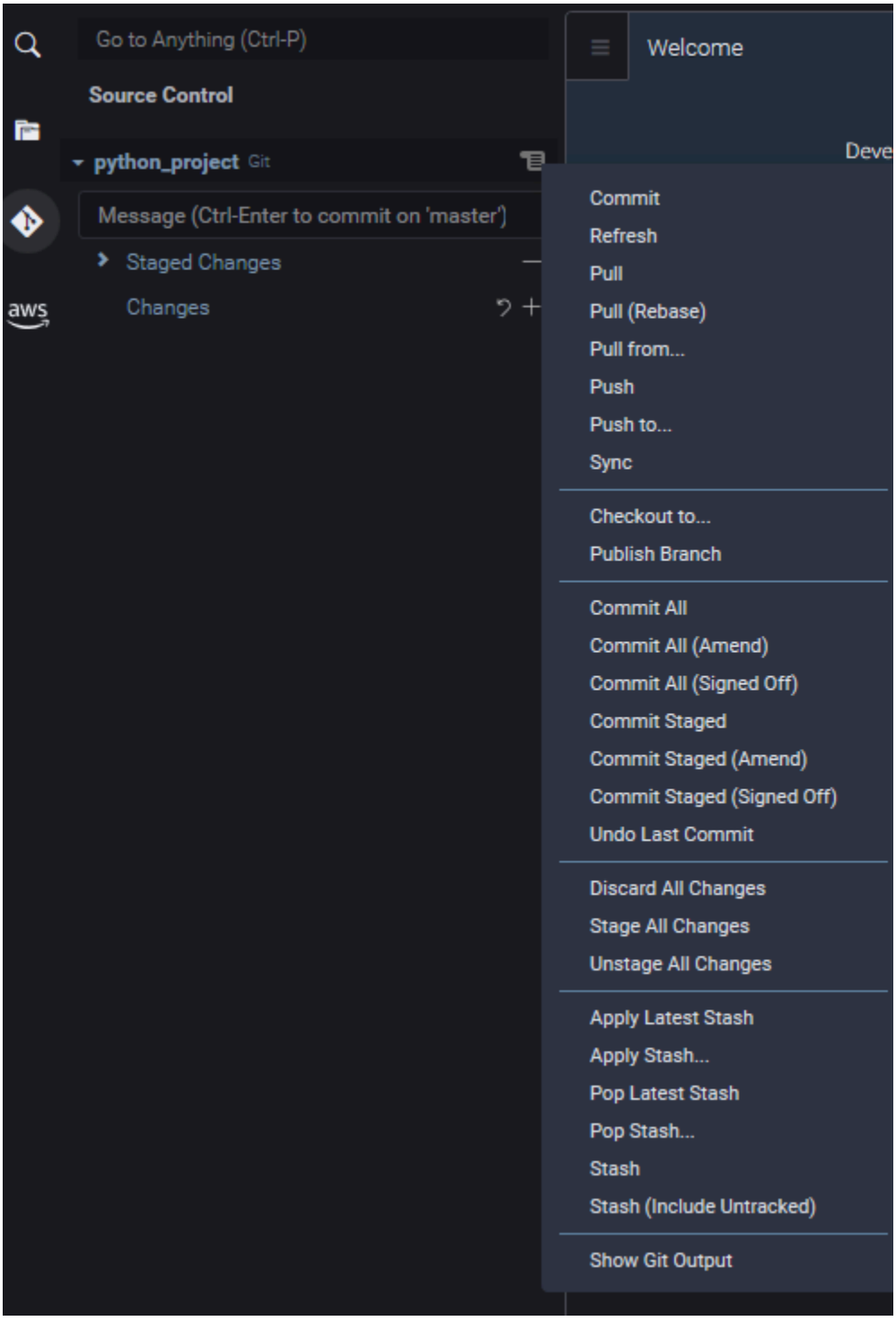
Topik


- Mengelola kontrol sumber dengan panel Git
- Referensi: Perintah Git tersedia di panel Git


# Mengelola kontrol sumber dengan panel Git

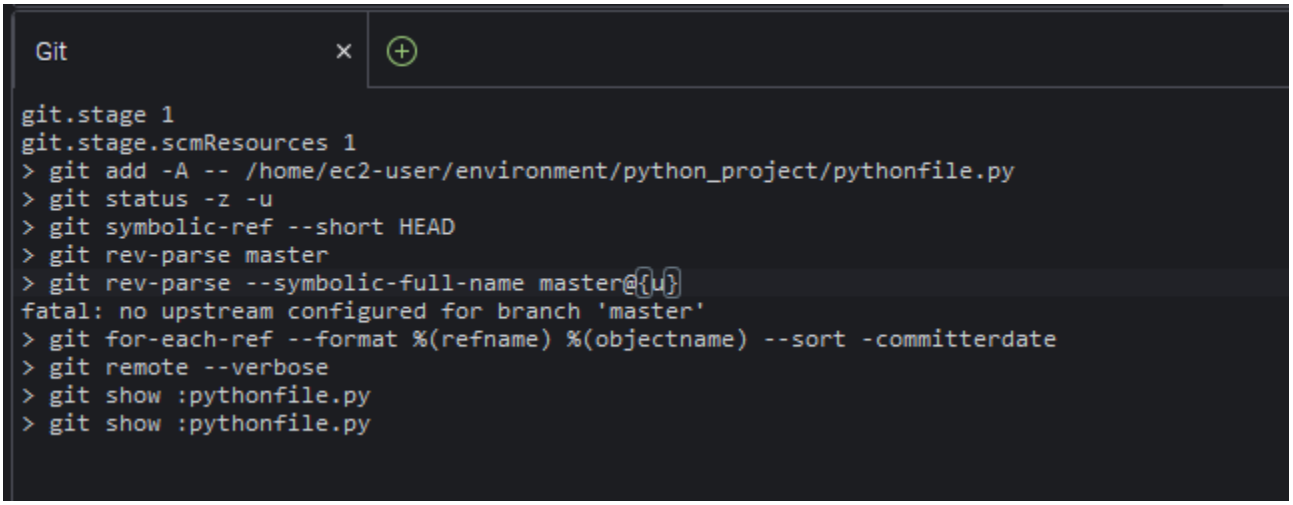
Ekstensi panel Git untuk AWS Cloud9 menyediakan akses antarmuka pengguna yang nyaman ke perintah Git inti dan lanjutan.

Bagian ini menunjukkan bagaimana untuk mengakses fitur access key Git kunci untuk mengelola kontrol sumber. Prosedur berfokus pada penggunaan menu Panel Git untuk menjalankan perintah Git terhadap repositori Anda dan isinya.




Anda juga dapat mengakses perintah Git yang di support dengan mulai memasukkan nama di kotak pencarian panel Git:

Dan Anda dapat melihat perintah Git aktual yang dijalankan ketika Anda berinteraksi dengan antarmuka panel Git. Untuk melihat aktivitas baris perintah, pergi ke menu Panel Git dan memilih Tampilkan Output Git.

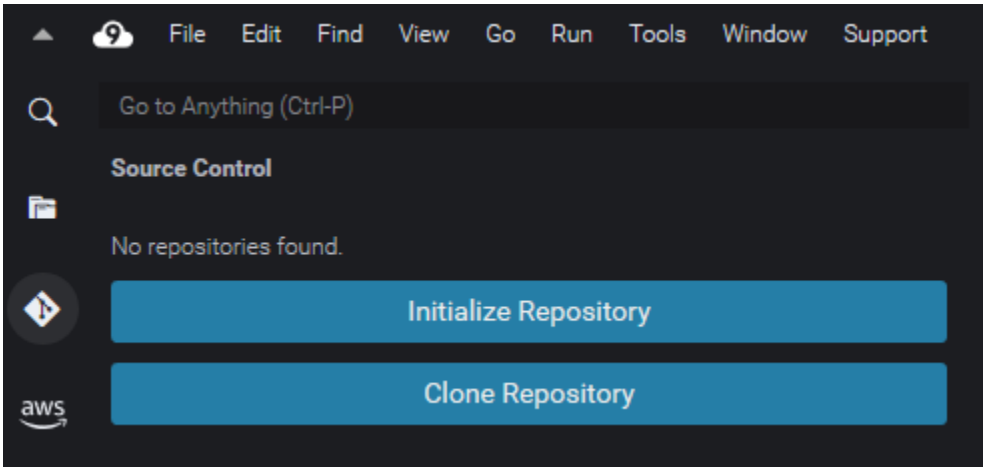


# Menginisialisasi atau klon repositori Git

Sebuah repositori Git ("repo") berisi riwayat lengkap proyek dari awal. Repositori terdiri dari semua snapshot konten proyek yang diambil setiap kali Anda melakukan file bertahap ke repo itu.

Panel Git men-support kedua cara mendapatkan repositori Git:

- Menginisialisasi direktori yang ada sebagai repositori Git.
- Klon repositori yang ada dan menyalinnya ke direktori lokal.




> **Note**
>
> Pilihan antarmuka untuk menginisialisasi atau kloning repo tersedia hanya jika Anda belum memiliki repositori Git ditambahkan ke folder workspace di lingkungan Anda. Jika Anda sudah memiliki direktori kerja untuk repositori, jendela panel Git menampilkan status direktori kerja dan area persiapan. Menu Panel Git juga tersedia untuk menyediakan akses ke perintah Git yang dapat Anda jalankan terhadap repositori.

Untuk menginisialisasi atau klon repositori Git

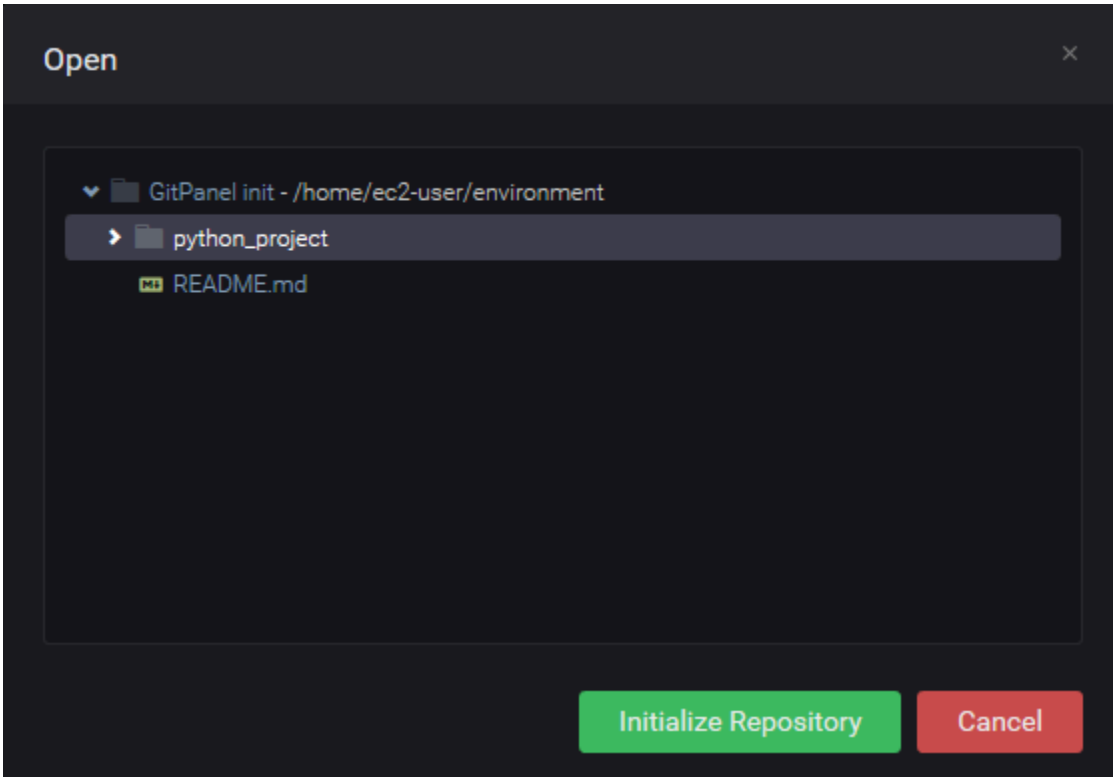
1. Jika panel Git belum tersedia, akses dengan memilih Jendela, Kontrol Sumber, dan kemudian memilih ikon Git.

> **Note**
>
> Anda juga dapat membuka panel Git menggunakan shortcut keyboard Ctrl+Shift+G.

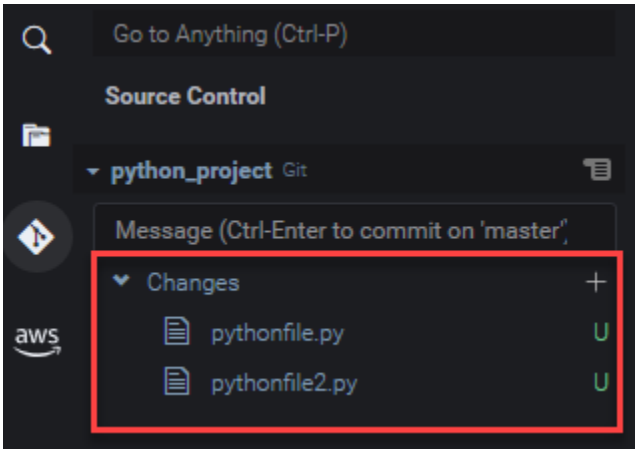
2. Memilih apakah akan menginisialisasi repo baru atau klon yang sudah ada.

   Initialize a repository

   - Dalam panel Git, memilih Inisialisasi Repositori.
   - Selanjutnya, memilih folder workspace tempat repo Git Anda akan diinisialisasi. Anda dapat memasukkan path ke folder, memilih path, atau memilih folder di kotak dialog.
   - Jika Anda menggunakan kotak dialog, pilih folder tujuan dan memilih Inisialisasi Repositori.




   Setelah Anda menginisialisasi repo Git dalam folder yang dipilih, panel Git menampilkan file yang sudah dalam folder tersebut sebagai untracked dan siap untuk ditambahkan ke area persiapan Git.




Clone a repository

- Di jendela panel Git, memilih Klon Repositori.
- Selanjutnya masukkan URL untuk repo jarak jauh yang ingin Anda kloning (**`https://github.com/my-own-repo/my-repo-project-name.git`**, misalnya, untuk mengkloning repo yang dihosting GitHub) dan tekan Return.
- Dalam kotak dialog yang memaparkan, pilih folder workspace untuk repo kloning dan memilih Pilih Lokasi Repositori.

> **Note**
>
> Jika Anda mengakses repositori yang dihosting di situs eksternal (misalnya)GitHub, Anda juga perlu memasukkan kredensi masuk untuk situs untuk menyelesaikan proses.

Setelah Anda klon repo jauh di folder yang dipilih, Anda dapat menjalankan perintah `git pull` untuk menyinkronkan repositori lokal Anda dengan perubahan terbaru dalam repositori jarak jauh. Untuk informasi selengkapnya, lihat Bekerja dengan repositori jauh.

## Staging dan commiting file

Setelah Anda memperoleh repositori Git, Anda kemudian dapat mulai mengisinya dengan konten menggunakan proses dua langkah:

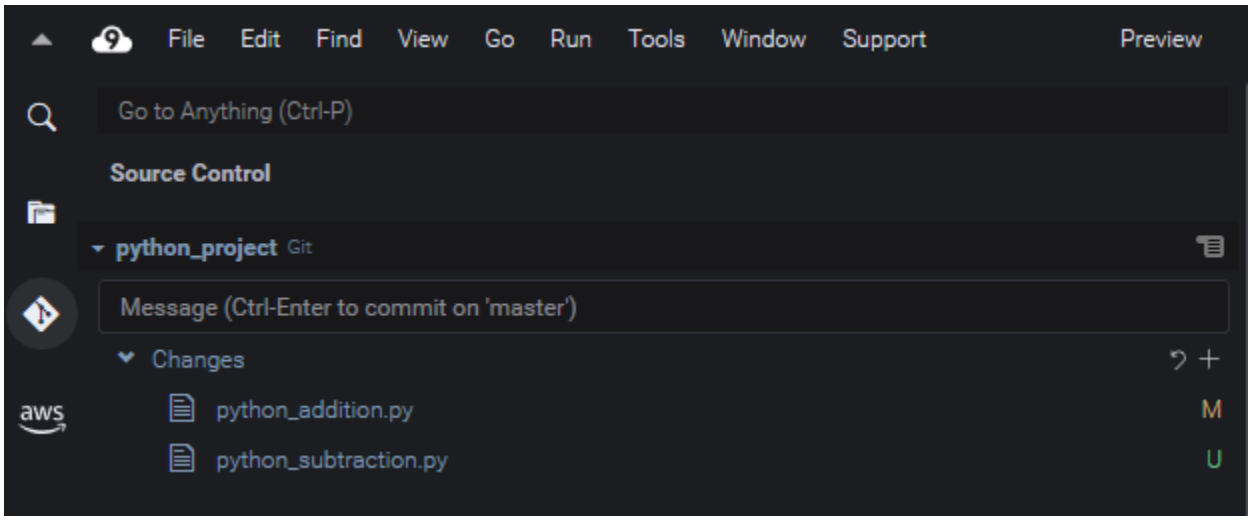
1. Menambahkan konten yang belum dilacak atau baru dimodifikasi ke area persiapan.

2. Commit file di area persiapan ke direktori kerja.

> **Important**
>
> Anda mungkin tidak ingin commit setiap file dalam direktori kerja Anda ke repositori. Misalnya, Anda tidak mungkin ingin menambahkan file yang dihasilkan selama runtime ke repositori proyek Anda. Dengan panel Git, Anda dapat menandai file untuk diabaikan dengan menambahkan mereka ke daftar di file `.gitignore`.
> Untuk memperbarui daftar di `.gitignore`, klik kanan file yang belum ditambahkan ke area persiapan dan pilih Menambahkan File ke .gitignore. IDE membuka file `.gitignore` dan nama file yang dipilih ditambahkan ke daftar file yang diabaikan.
> Untuk informasi tentang penggunaan pencocokan pola di `.gitignore` untuk mengecualikan jenis file, lihat referensi di situs git-scm.com.

Stage files

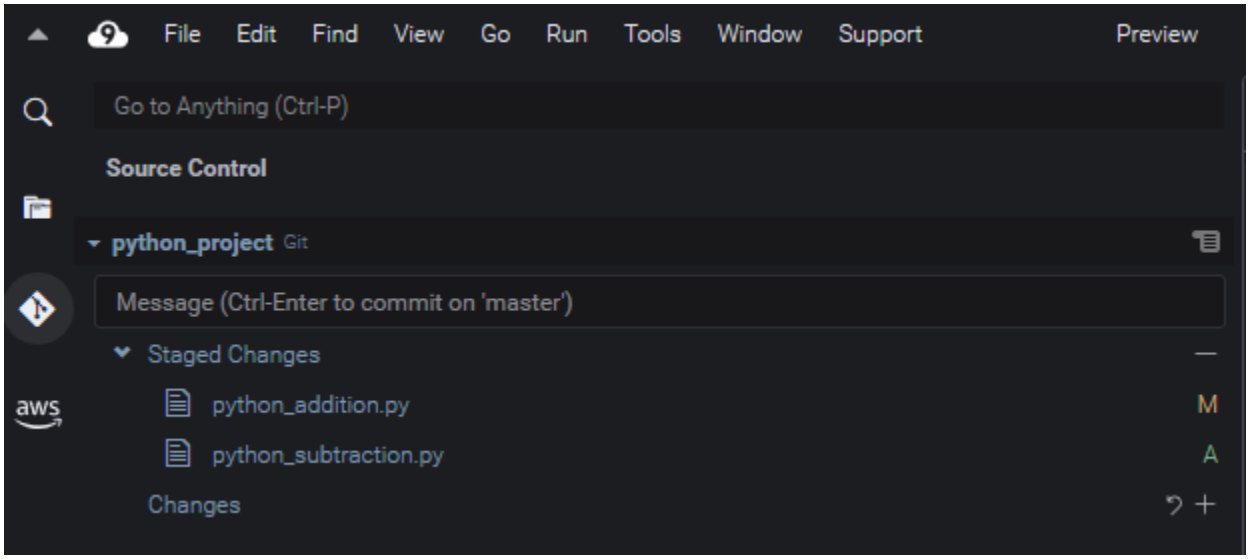
File yang tidak dilacak (berlabel “U”) dan file yang dimodifikasi (berlabel “M”) yang belum ditambahkan ke area pementasan tercantum di bawah Perubahan di panel panel Git.




Menggunakan antarmuka panel Git, Anda dapat menambahkan file tertentu atau semua file yang tidak dilacak dan dimodifikasi ke area persiapan:

- File tertentu: Jeda pada file dan kemudian memilih +untuk menambahkan ke area persiapan. Atau, klik kanan file dan memilih Stage Perubahan.
- Semua file: Pergi ke menu Panel Git dan memilih Stage Semua Perubahan.

File yang ditambahkan ke indeks repositori tercantum di bawah Perubahan yang di-Stage. File yang sebelumnya tidak dilacak diberi label “A” untuk menunjukkan bahwa file tersebut telah dipentaskan.




> **Note**
>
> Anda juga dapat membatalkan stage perubahan tertentu atau semua perubahan. Untuk satu file, jeda pada file dan kemudian memilih -. Atau, klik kanan file dan memilih Unstage Perubahan. Untuk unstage semua perubahan, pergi ke menu Panel Git dan memilih Unstage Semua Perubahan.

## Commit files

Anda dapat menggunakan perintah `commit` Git untuk menangkap file yang di-stage sebagai snapshot permanen di repositori. Menggunakan antarmuka panel Git, Anda dapat memilih file mana yang akan di-commit:

- Commit file di area persiapan: Buka menu Panel Git dan memilih Commit atau Commit Staged.
- Commit semua file dalam direktori kerja: Pergi ke menu Panel Git dan memilih Commit Semua. (Opsi ini menggunakan `git add` untuk menambahkan file ke area persiapan sebelum memanggil `git commit`.)

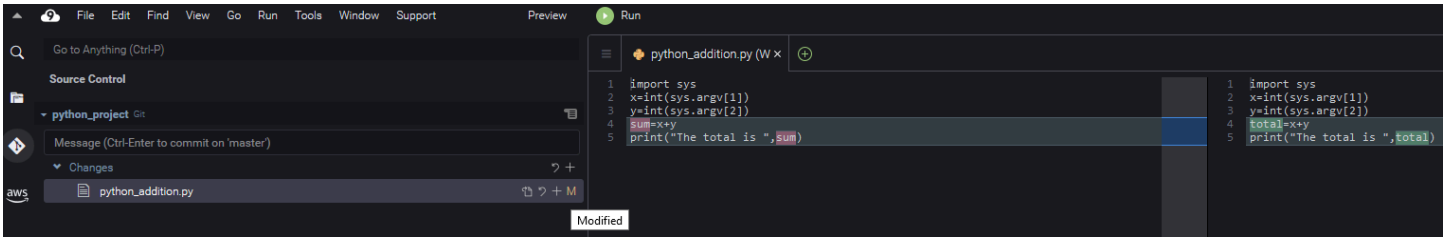
> **Note**
>
> Anda juga dapat menggunakan `amend` dan `signed-off` ketika melakukan commit file dengan panel Git. Pilihan `amend` memodifikasi olahpesan commit dari commit terbaru. Pilihan `sign-off` dapat mengidentifikasi siapa yang melakukan commit di log Git. Anda juga dapat membalikkan commit dengan pergi ke menu Panel Git dan memilih Urungkan Commit Terakhir

## Melihat versi file yang berbeda

Anda dapat membandingkan versi file yang telah dimodifikasi setelah di-stage atau di-commit.

- File yang tercantum di bawah Perubahan: Memilih "M" untuk melihat perbedaan antara versi dalam direktori kerja dan versi yang terakhir di-stage atau di-commit ke repo.
- File yang tercantum di bawah Perubahan Staged: Memilih "M" untuk melihat perbedaan antara versi dalam area persiapan dan versi yang terakhir di-stage atau di-commit ke repo.

Setelah Anda memilih "M", jendela IDE menampilkan perbedaan antara dua versi file. Satu sisi menunjukkan versi yang dilacak sebagai saat ini dalam repositori. Sisi lain menunjukkan versi modifikasi yang belum di-commit.



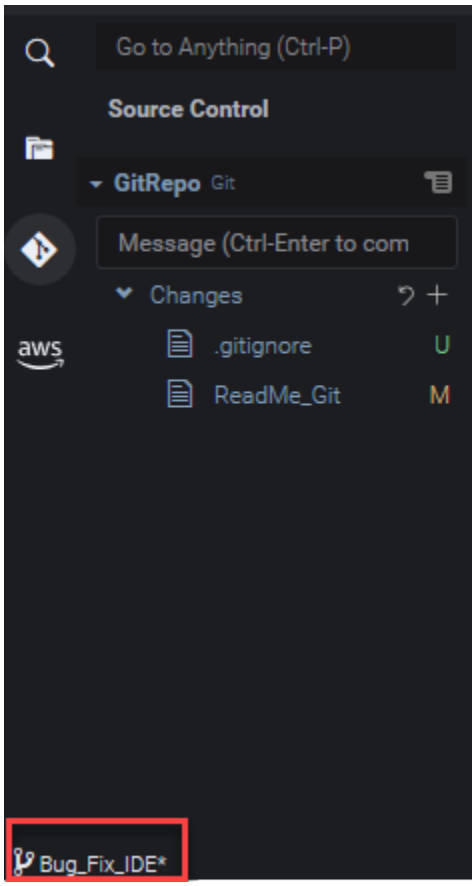

## Bekerja dengan cabang

Git sangat memudahkan pengelolaan alur kerja dengan mengizinkan Anda untuk bekerja pada fitur-fitur baru di cabang-cabang yang independen dari cabang utama repo. Anda dapat beralih dengan mulus di antara beberapa cabang sambil memastikan Anda selalu memiliki kode ready-to-build sumber di cabang utama.

### Buatlah sebuah cabang

Membuat cabang melibatkan penamaan cabang dan memilih titik awal.

1. Di menu Panel Git, memilih Checkout ke. Atau, Anda dapat memilih nama cabang saat ini yang ditampilkan di bagian bawah panel Git.




2. Memilih opsi untuk membuat cabang baru:
   - Buat cabang baru: Cabang baru dimulai dari commit terakhir cabang saat ini.
   - Buat cabang baru dari: Cabang baru dimulai dari commit terakhir cabang yang Anda pilih di layar berikutnya.
3. Memasukkan nama cabang baru.
4. Jika Anda menentukan cabang tertentu sebagai titik awal untuk cabang Anda, pilih salah satu dari daftar.

Setelah beralih ke cabang baru, Anda dapat memeriksa nama cabang saat ini dengan melihat bagian bawah panel Git.

> **Note**
>
> Jika Anda bekerja dengan repositori jarak jauh, mempublikasikan cabang baru ke repositori remote upstream untuk mengizinkan orang lain mengakses konten Anda.

### Beralih cabang

Salah satu keuntungan kunci dari mengelola kontrol sumber dengan Git adalah bahwa Anda dapat melompat di antara proyek yang berbeda hanya dengan beralih cabang.

> **Important**
>
> Anda tidak dapat mengganti cabang jika Anda memiliki file di cabang saat ini yang belum di-commit ke repositori Anda. Anda harus terlebih dahulu membersihkan direktori kerja Anda dengan meng-commit atau menyimpan pekerjaan Anda.

1. Memilih nama cabang saat ini di bagian bawah panel Git. Atau, pergi ke Panel Git dan memilih Checkout ke.
2. Memilih cabang dari daftar yang ditampilkan.

Setelah Anda beralih, direktori kerja repositori diperbarui dengan versi file yang baru di-commit untuk cabang yang dipilih.

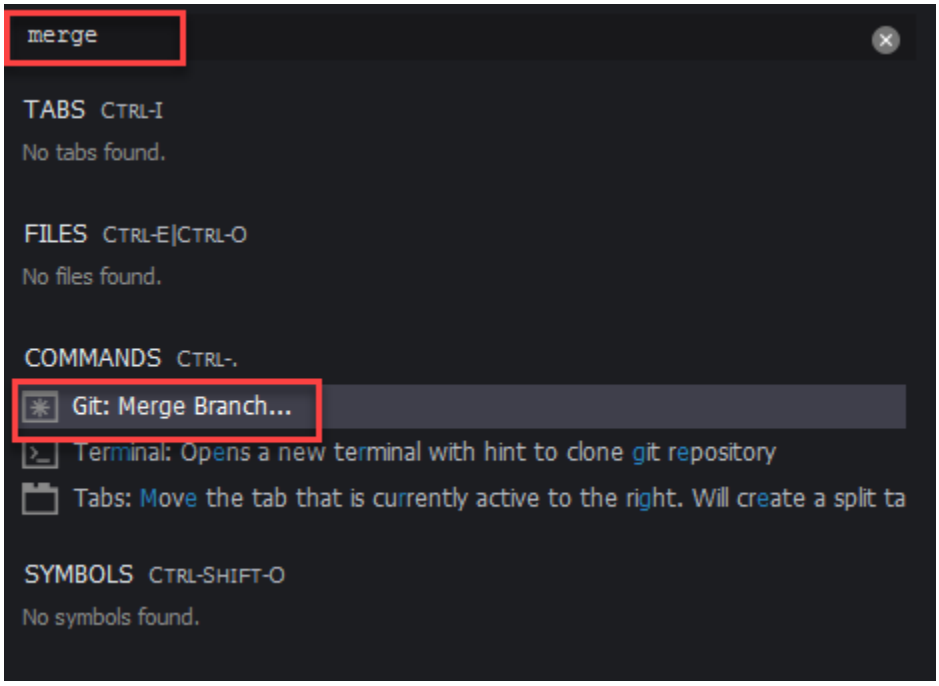
### Menggabungkan cabang

Setelah Anda selesai mengerjakan sebuah fitur di cabang diskrit, Anda biasanya ingin mengintegrasikan perubahan ke dalam proyek utama. Dengan Git, integrasi semacam ini difasilitasi dengan menggabungkan satu cabang (cabang fitur, misalnya) ke cabang lain (biasanya cabang utama atau default repositori).

1. Untuk memilih cabang tempat Anda akan menggabungkan cabang lain, pergi ke menu Panel Git dan memilih Checkout ke.

   Atau, memilih nama cabang saat ini di bagian bawah panel Git.
2. Dari daftar yang ditampilkan, memilih cabang untuk beralih.
3. Di kotak Cari untuk panel Git, mulai memasukkan kata "merge".

Saat Git: Merge Branch menampilkan di bawah daftar Perintah, memilih itu.

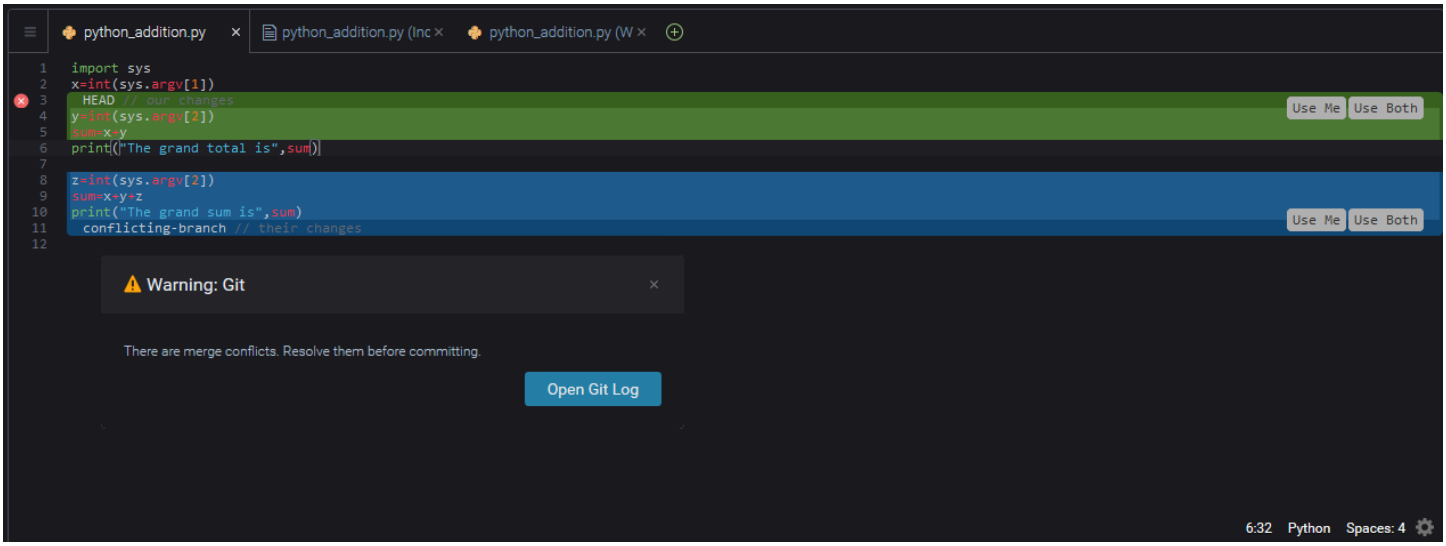



4. Dari daftar yang ditampilkan, memilih cabang untuk bergabung ke cabang target.

   Jika gabungan selesai tanpa konflik, antarmuka panel Git refresh untuk menunjukkan cabang target yang berisi perubahan gabungan.

Saat menggabungkan cabang, Anda mungkin mengalami konflik gabungan yang dihasilkan dari perubahan yang tidak kompatibel yang dibuat untuk konten yang sama. Jika hal ini terjadi, Anda diperingatkan bahwa Anda harus menyelesaikan konflik sebelum melakukan penggabungan.

Anda dapat menggunakan jendela editor kode IDE untuk mengidentifikasi konten yang bertentangan di dua cabang dan kemudian membuat perubahan untuk menyelesaikan perbedaan.



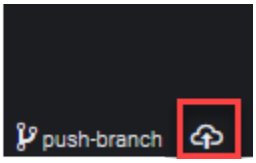

# Bekerja dengan repositori jauh

Repositori jarak jauh yang di-host di Internet atau jaringan memfasilitasi kolaborasi dengan mengizinkan anggota tim berbagi perubahan yang telah mereka lakukan terhadap tanggung jawab lokal mereka. Dengan menggunakan perintah Git yang unggah dan mengunduh data, Anda memastikan isi repositori "downstream" (lokal) disinkronkan dengan repositori "upstream" (jarak jauh).

Publikasikan sebuah cabang ke repositori jarak jauh

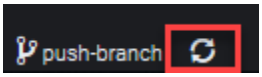
Setelah Anda membuat cabang untuk repositori lokal, itu privat untuk Anda dan tidak tersedia untuk kolaborator Anda sampai Anda mendorongnya "upstream" ke repositori jarak jauh.

1. Untuk mempublikasikan cabang saat ini, pergi ke menu Panel Git dan memilih Publikasikan Cabang. Atau, klik simbol cloud yang ada di sebelah nama cabang di bagian bawah panel Git.

2. Jika diperlukan, masukkan kredenal masuk Anda untuk mengakses repositori jarak jauh.

Jika cabang berhasil diterbitkan ke repositori jarak jauh, simbol sinkronisasi menampilkan di sebelah nama cabang di bagian bawah panel Git. Memilih untuk menyinkronkan isi repositori lokal dan jarak jauh.




## Dorong dan tarik konten antara repositori lokal dan jarak jauh

Ketika menggunakan Git untuk berkolaborasi pada proyek bersama, Anda biasanya mulai dengan menarik perubahan terbaru oleh anggota tim lain dari repositori jarak jauh ke repo lokal Anda. Dan setelah Anda commit perubahan repo lokal Anda, Anda mendorong mereka ke repositori jarak jauh sehingga mereka dapat diakses oleh seluruh tim. Tindakan ini dilakukan oleh perintah `git pull` dan `git push`.

> **Note**
>
> Anda harus memasukkan kredensi masuk saat mendorong dan menarik perubahan ke dan dari sebagian besar repositori yang dihosting (seperti yang ada di GitHub, misalnya).

Pull changes from remote

Menggunakan perintah `git pull` melalui antarmuka panel Git, Anda dapat memperbarui repositori lokal Anda dengan perubahan terbaru yang dilakukan pada cabang di repositori jarak jauh.

1. Di menu Panel Git, memilih Checkout ke.
2. Dalam daftar cabang, memilih cabang lokal yang ingin Anda tarik perubahannya.
3. Selanjutnya, pergi ke menu Panel Git dan memilih Tarik dari.
4. Pilih repositori jarak jauh dan kemudian cabang di repositori itu untuk menarik perubahan.

Setelah menarik, Anda dapat mengakses file yang diambil dari repo remote di direktori kerja repositori Anda. Setelah Anda memodifikasi file, Anda kemudian dapat mendorong perubahan Anda ke cabang jarak jauh.

Push changes to remote

Menggunakan perintah `git push` melalui antarmuka panel Git, Anda dapat memperbarui repositori jarak jauh Anda dengan perubahan terbaru di cabang tertentu di repositori lokal Anda.

1. Di menu Panel Git, memilih Checkout ke.
2. Dalam daftar cabang, memilih cabang lokal yang ingin Anda tarik perubahannya.

3. Selanjutnya, pergi ke menu Panel Git dan memilih Dorong ke.
4. Pilih repositori jarak jauh dan kemudian cabang di repositori itu untuk mendorong perubahan.

Setelah mendorong, anggota tim lain dapat mengakses perubahan Anda dengan menariknya ke salinan lokal repositori mereka sendiri.

## Menyimpan dan mengambil file

Dengan fitur menyimpan dari Git, Anda dapat beralih cabang tanpa terlebih dahulu harus commit file staged atau dimodifikasi. Fitur menyimpan menangkap status saat ini direktori kerja dan area persiapan dan menyimpannya untuk digunakan nanti. Fitur ini berguna setiap kali Anda masih mengerjakan konten yang belum selesai dan perlu mengganti cabang tanpa penundaan.

Pekerjaan menyimpan

1. Untuk menyimpan keadaan direktori kerja Anda saat ini, pergi ke menu Panel Git dan memilih salah satu opsi berikut:
   - Simpan: Semua file yang dimodifikasi atau di-stage dalam direktori kerja ditambahkan ke simpanan. File yang tidak dilacak tidak ditambahkan.
   - Simpan (termasuk Tidak Terlacak): Semua file dalam direktori kerja, termasuk yang belum dilacak, ditambahkan ke simpanan.
2. Memasukkan olahpesan opsional yang akan membantu Anda mengidentifikasi simpanan untuk pengambilan di masa mendatang.

Setelah menyimpan, antarmuka panel Git refresh untuk menampilkan direktori kerja yang telah dibersihkan.

Mengambil simpanan

1. Untuk mengambil simpanan dan mengaplikasikannya ke direktori kerja Anda, pergi ke menu Panel Git dan memilih salah satu opsi berikut:
   - Terapkan Simpanan: Menerapkan simpanan yang dipilih ke direktori kerja Anda dan menyimpan simpanan untuk digunakan nanti.
   - Pop Simpanan: Menerapkan simpanan yang dipilih untuk direktori kerja Anda dan menghapus simpanan dari tumpukan simpanan.

> **Note**
>
> Anda juga dapat memilih untuk menerapkan atau mengeluarkan stash terakhir yang ditambahkan ke tumpukan simpanan.

2. Pilih simpanan untuk diterapkan ke direktori kerja.

Antarmuka panel Git refresh untuk menampilkan direktori kerja Anda dengan simpanan diterapkan.

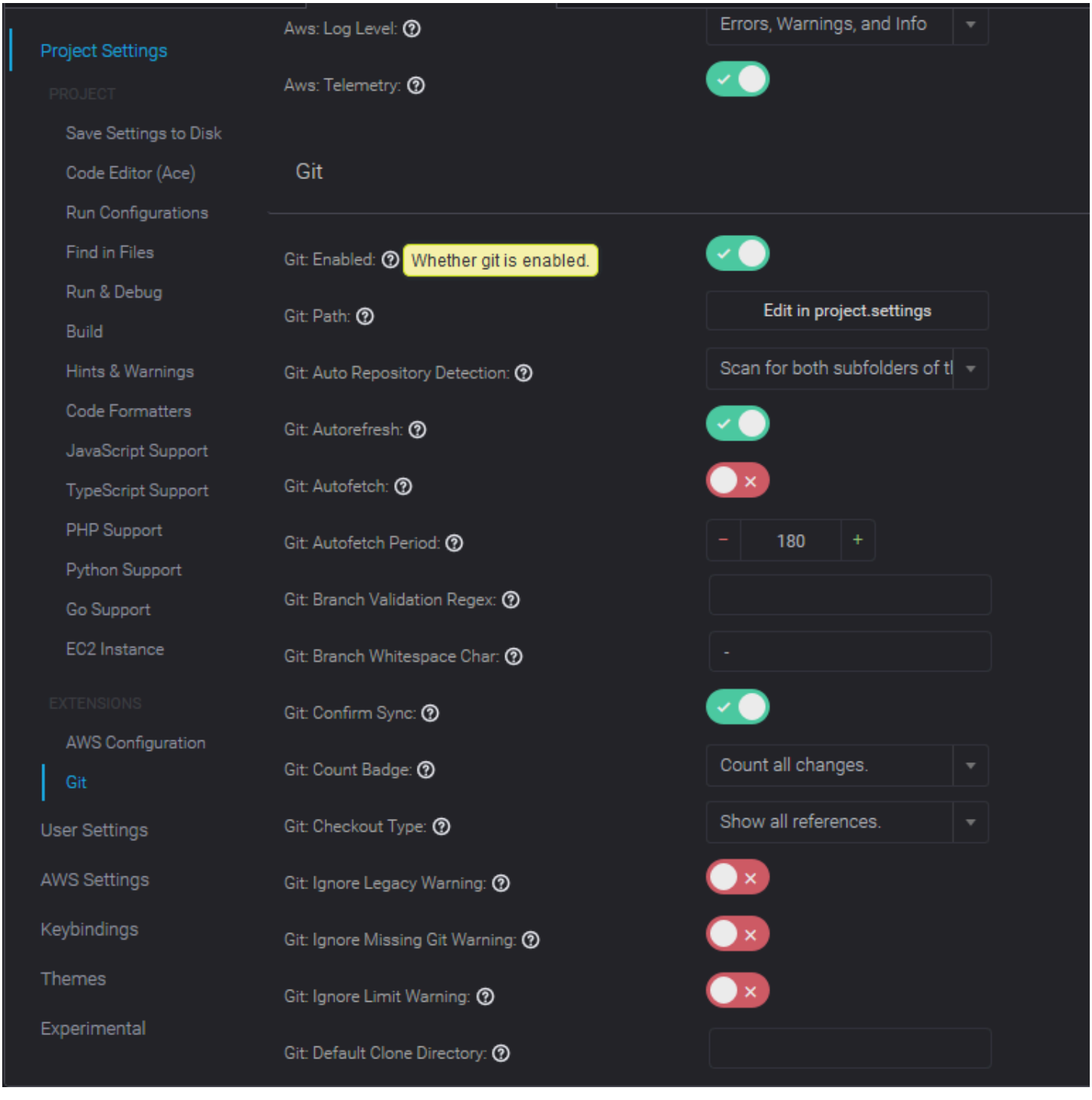
## Referensi: Perintah Git tersedia di panel Git

Menu panel Git untuk AWS Cloud9 menyediakan akses antarmuka pengguna yang nyaman ke perintah git inti dan lanjutan.

Perintah git tertentu—seperti yang digunakan untuk menggabungkan dan menghapus cabang, misalnya—hanya tersedia melalui bidang pencarian panel Git.

Anda juga dapat menyesuaikan bagaimana panel Git menjalankan perintah dan berinteraksi dengan repositori. Untuk mengubah pengaturan default, pertama pilih AWS Cloud9, Preferensi. Selanjutnya, di jendela Preferensi, di bawah Pengaturan Proyek, pilih Git.

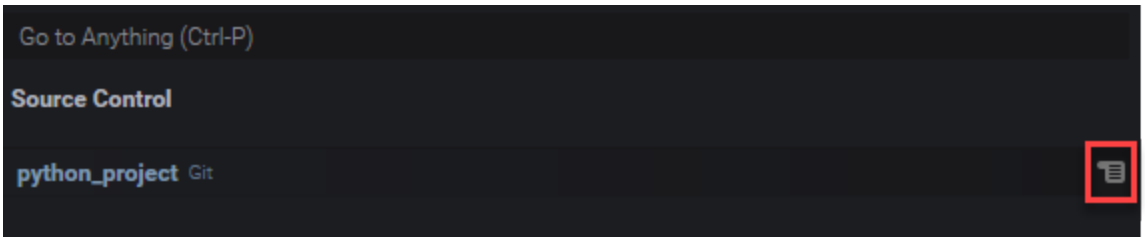
Jeda di atas ikon informasi untuk membaca deskripsi singkat tentang pengaturan.




> **Note**
>
> Anda dapat mengakses dokumentasi terperinci pada perintah Git yang terdaftar dari situs resmi Git: https://git-scm.com/doc.

# Referensi untuk perintah Git tersedia dari menu panel Git

Anda mengakses opsi pada menu Panel Git dengan memilih simbol di seberang nama repositori.




Menu panel Git

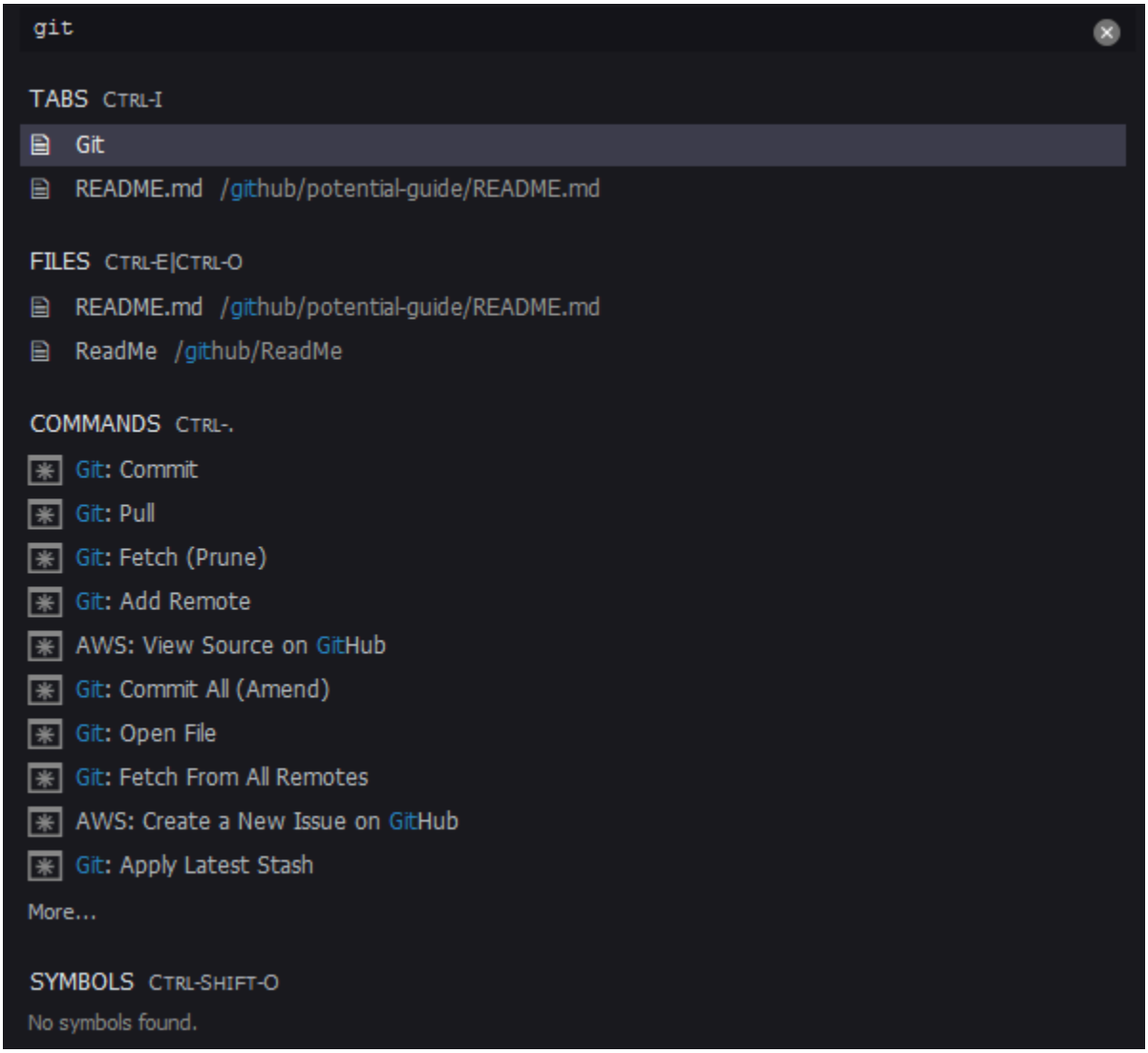
| Pilihan menu | Deskripsi |
|---|---|
| Berkomitmen | Mengkomit konten yang ditambahkan area persiapan ke direktori kerja dari repositori. Menambahkan pesan komit. |
| Segarkan | Menyegarkan GitPanel antarmuka untuk menunjukkan status direktori kerja dan area pementasan. |
| Tarik | Menarik perubahan terbaru dari repositori jarak jauh ke repositori lokal. |
| Tarik (Rebase) | Menerapkan kembali perubahan lokal Anda ke perubahan jarak jauh yang ditarik dari cabang jarak jauh. |
| Dorong dari... | Mendorong perubahan yang dikomit ke cabang di repositori lokal Anda ke cabang di repositori jarak jauh. |
| Dorong | Mendorong perubahan yang dikomit ke repositori lokal ke repositori jarak jauh. |
| Dorong ke... | Mendorong perubahan yang dikomit ke cabang di repositori lokal Anda ke cabang di repositori jarak jauh. |
| Sinkronisasi | Menyinkronkan konten repositori lokal dan jarak jauh dengan menjalankan perintah `git pull` diikuti dengan perintah `git push`. |

| Pilihan menu | Deskripsi |
| --- | --- |
| Checkout ke... | Beralih ke cabang yang ada atau membuat cabang dan beralih ke sana. |
| Publikasikan Cabang | Memublikasikan cabang pribadi yang dibuat pada repositori lokal dan membuatnya tersedia di repositori jarak jauh. |
| Komit Semua | Mengkomit file yang ditahapkan dan tidak ditahapkan ke repositori. (Sebuah perintah `git add -A` dijalankan untuk menambahkan file ke area persiapan sebelum perintah `git commit` dijalankan.) |
| Komit Semua (Ubah) | Memodifikasi pesan dari komit terakhir. (Menambahkan opsi `-amend` ketika menjalankan perintah `git commit`.) |
| Komit Semua (Ditandatangani) | Mengidentifikasi siapa yang melakukan komit di log Git. (Menambahkan opsi `-signed-off` ketika menjalankan perintah `git commit`.) |
| Komit Dipentaskan | Mengkomit hanya file yang ditahapkan ke repositori. |
| Komit Dipentaskan (Ubah) | Memodifikasi pesan dari komit terakhir. (Menambahkan opsi `-amend` ketika menjalankan perintah `git commit`.) |
| Komit Dipentaskan (Ditandat angani) | Mengidentifikasi siapa yang melakukan komit di log Git. (Menambahkan opsi `-signed-off` ketika menjalankan perintah `git commit`.) |
| Batalkan Komit Terakhir | Membatalkan komit sebelumnya. File dipindahkan kembali ke area persiapan. |
| Buang Semua Perubahan | Menghapus semua file dan folder dari area persiapan pada repositori. |
| Tahap Semua Perubahan | Menambahkan konten yang belum terlacak dan dimodifikasi ke area persiapan. |

| Pilihan menu | Deskripsi |
| --- | --- |
| Buka Panggung Semua Perubahan | Memindahkan semua file keluar dari area persiapan. File yang tidak ditahapkan tidak dapat dikomit ke repositori. |
| Terapkan Stash Terbaru | Menerapkan simpanan terakhir yang telah ditambahkan ke simpanan tumpukan ke direktori kerja. Simpanan tetap berada di tumpukan. |
| Terapkan Stash... | Menerapkan simpanan yang dipilih dari tumpukan simpanan ke direktori kerja. Simpanan tetap berada di tumpukan. |
| Pop Stash Terbaru | Menerapkan simpanan terakhir yang telah ditambahkan ke simpanan tumpukan ke direktori kerja. Simpanan kemudian dihapus dari tumpukan. |
| Simpanan Pop... | Menerapkan simpanan yang dipilih ke direktori kerja. Simpanan kemudian dihapus dari tumpukan. |
| Stash | Menambahkan file yang dimodifikasi dan ditahapkan di direktori kerja ke simpanan bernama. |
| Stash (termasuk Untracked) | Menambahkan semua file, termasuk file yang tidak terlacak, di direktori kerja ke simpanan bernama. |
| Tampilkan Keluaran Git | Menampilkan jendela yang menunjukkan perintah Git yang dijalankan ketika Anda berinteraksi dengan antarmuka panel Git. |

## Perintah Git tersedia dari bidang pencarian panel Git

Anda juga dapat mengakses beberapa perintah Git yang didukung yang tidak tersedia di menu panel Git dengan mengetikkan "git" di kotak pencarian:




Tabel berikut memberikan deskripsi perintah Git yang dipilih yang dapat Anda akses dengan cara ini.

Perintah Git yang dipilih

| Pilihan menu | Deskripsi |
| --- | --- |
| Git: Tambahkan Remote | Menambahkan koneksi ke repositori jarak jauh ke file konfigurasi Git Anda |
| Git: Hapus Cabang | Menghapus cabang tertentu. |
| Git: Ambil | Mengunduh konten dari cabang di repositori jarak jauh. Berbeda dengan `git pull`, perubahan jarak jauh tidak digabung ke repositori lokal. |

| Pilihan menu | Deskripsi |
| --- | --- |
| Git: Gabungkan Cabang | Mengintegrasikan perubahan yang dibuat di satu cabang ke cabang lain. Untuk informasi selengkapnya, lihat prosedur penggabungan cabang. |

# Bekerja dengan AWS Toolkit

Anda dapat menavigasi dan berinteraksi dengan Layanan AWS menggunakan AWS Toolkit melalui jendela AWS Explorer.

## Mengapa menggunakan AWS Toolkit?

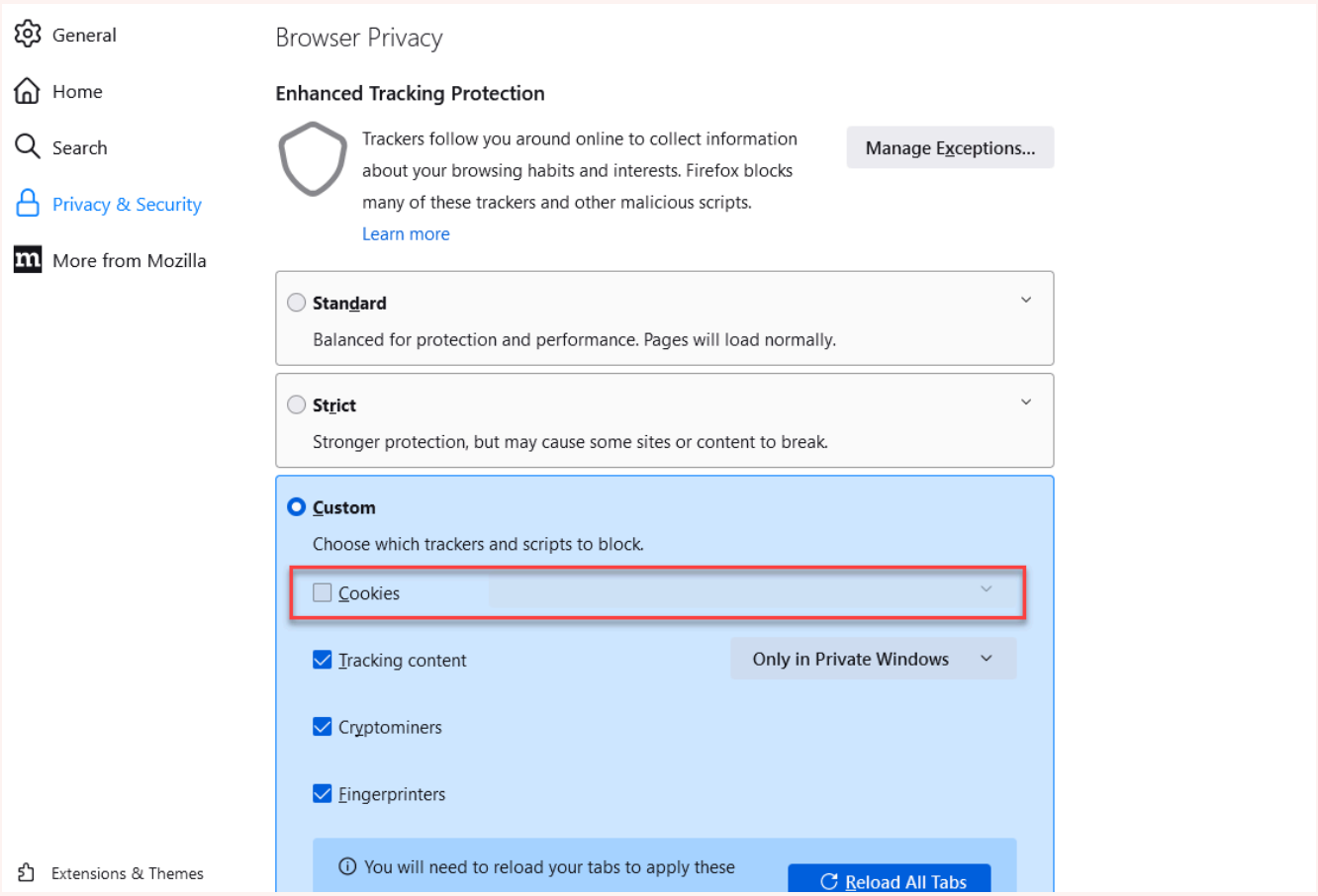
AWS Toolkit adalah ekstensi untuk lingkungan pengembangan AWS Cloud9 terintegrasi (IDE). Anda dapat mengakses dan bekerja dengan berbagai macam Layanan AWS melalui ekstensi ini. AWS Toolkit menggantikan fungsionalitas yang disediakan oleh plugin Lambda untuk. AWS Cloud9 Untuk informasi selengkapnya, lihat Menonaktifkan Toolkit AWS.

**Important**

AWS Dukungan Toolkit adalah fitur terintegrasi dari. AWS Cloud9 Saat ini, Anda tidak dapat menyesuaikan AWS Cloud9 IDE dengan ekstensi pihak ketiga.

**Warning**

Jika Anda menggunakan Mozilla Firefox sebagai browser pilihan Anda dengan AWS Cloud9 IDE, ada pengaturan cookie pihak ketiga yang mencegah AWS Cloud9 tampilan web dan AWS Toolkit berfungsi dengan benar di browser. Sebagai solusi untuk masalah ini, Anda harus memastikan bahwa Anda belum memblokir Cookie di bagian Privasi & Keamanan pengaturan browser Anda, seperti yang ditampilkan pada gambar di bawah ini.




Saat ini, berikut Layanan AWS dan sumber daya dapat diakses melalui ekstensi AWS Toolkit:

- AWS Pelari Aplikasi
- API Gateway
- AWS CloudFormation tumpukan
- CloudWatch Log
- AWS Lambda
- Sumber
- Ember dan objek Amazon S3
- AWS Serverless Application Model aplikasi
- Step Functions dan mesin negara
- Dokumen otomatisasi Systems Manager
- Bekerja dengan Amazon ECR di IDE AWS Cloud9

- AWS IoT
- ???
- Amazon EventBridge
- Bekerja dengan AWS Cloud Development Kit (AWS CDK)

# Mengaktifkan Toolkit AWS

Jika AWS Toolkit tidak tersedia di lingkungan Anda, Anda dapat mengaktifkannya di tab Preferensi.

Untuk mengaktifkan AWS Toolkit

1. Pilih AWS Cloud9, Preferensi pada bilah menu.
2. Pada tab Preferensi, di panel navigasi samping, pilih AWS Pengaturan.
3. Di panel AWS Resources, aktifkan AWS Toolkit sehingga menampilkan tanda centang pada latar belakang hijau.

   Saat Anda mengaktifkan AWS Toolkit, lingkungan pengembangan terintegrasi (IDE) akan diperbarui untuk menampilkan pengaturan Enable AWS  Toolkit yang diperbarui. Opsi AWS Toolkit di sisi IDE di bawah opsi Lingkungan juga muncul.

> **Important**
>
> Jika EC2 instans AWS Cloud9 lingkungan Anda tidak memiliki akses ke internet (yaitu, tidak ada lalu lintas keluar yang diizinkan), pesan mungkin akan ditampilkan setelah Anda mengaktifkan AWS Toolkit dan meluncurkan kembali IDE. Pesan ini menyatakan bahwa dependensi yang diperlukan oleh AWS Toolkit tidak dapat diunduh. Jika ini masalahnya, Anda juga tidak dapat menggunakan AWS Toolkit.
> Untuk memperbaiki masalah ini, buat VPC endpoint untuk Amazon S3. Ini memberikan akses ke bucket Amazon S3 di bucket Wilayah AWS Anda yang berisi dependensi yang diperlukan untuk memperbarui IDE Anda.
> Untuk informasi selengkapnya, lihat Mengonfigurasi VPC endpoint untuk Amazon S3 guna mengunduh dependensi.

# Mengelola kredensi akses untuk Toolkit AWS

AWS Toolkit berinteraksi dengan berbagai macam. Layanan AWS Untuk mengelola kontrol akses, pastikan bahwa entitas IAM untuk layanan AWS Toolkit Anda memiliki izin yang diperlukan untuk berbagai layanan ini. Sebagai permulaan yang cepat, gunakan kredenal sementara yang AWS dikelola untuk mendapatkan izin yang diperlukan. Kredensi terkelola ini berfungsi dengan memberikan akses EC2 lingkungan Anda Layanan AWS atas nama AWS entitas, seperti pengguna IAM.

Namun, jika Anda telah meluncurkan EC2 instance lingkungan pengembangan Anda ke subnet pribadi, kredensi sementara AWS terkelola tidak tersedia untuk Anda. Jadi, sebagai alternatif, Anda dapat mengizinkan AWS Toolkit untuk mengakses Anda Layanan AWS dengan membuat set kredenal Anda sendiri secara manual. Set ini disebut profil. Profil menampilkan kredensi jangka panjang yang disebut kunci akses. Anda bisa mendapatkan kunci akses ini dari konsol IAM.

Buat profil untuk memberikan kredensi akses untuk Alat kit AWS

1. Untuk mendapatkan kunci akses Anda (terdiri dari ID kunci akses dan kunci akses rahasia), buka konsol IAM di https://console.aws.amazon.com/iam.
2. Pilih Pengguna dari bilah navigasi lalu pilih AWS nama pengguna Anda (bukan kotak centang).
3. Pilih tab Kredensial keamanan, lalu pilih Buat access key.

   > **Note**
   >
   > Jika Anda sudah memilik access key tetapi Anda tidak dapat mengakses kunci rahasia Anda, nonaktifkan kunci lama dan buat yang baru.

4. Di kotak dialog yang menunjukkan access key ID dan secret access key Anda, pilih Unduh file .csv untuk menyimpan informasi ini di lokasi yang aman.
5. Setelah Anda mengunduh kunci akses Anda, luncurkan AWS Cloud9 lingkungan dan mulai sesi terminal dengan memilih Window, Terminal Baru.
6. Di jendela terminal, jalankan perintah berikut.

   ```
   aws configure --profile toolkituser
   ```

   Dalam kasus ini, `toolkituser` adalah nama profil yang digunakan, tetapi Anda dapat memilih sendiri.

7. Pada baris perintah, masukkan `AWS Access Key ID` dan `AWS Secret Access Key` yang sebelumnya Anda unduh dari konsol IAM.

   - Untuk`Default region name`, tentukan Wilayah AWS (misalnya,`us-east-1`).
   - Untuk`Default output format`, tentukan format file (misalnya,`json`).

   > **Note**
   >
   > Untuk informasi tentang opsi untuk mengonfigurasi profil, lihat Dasar-dasar konfigurasi di Panduan AWS Command Line Interface Pengguna.

8. Setelah Anda membuat profil Anda, luncurkan AWS Toolkit, buka menu AWS Toolkit, dan pilih Connect to. AWS
9. Untuk bidang Pilih profil AWS kredensi, pilih profil yang baru saja Anda buat di terminal (misalnya,`profile:toolkituser`).

Jika profil yang dipilih berisi kredenal akses yang valid, panel AWS Explorer akan menyegarkan untuk menampilkan Layanan AWS yang sekarang dapat Anda akses.

## Menggunakan peran IAM untuk memberikan izin ke aplikasi pada instance EC2

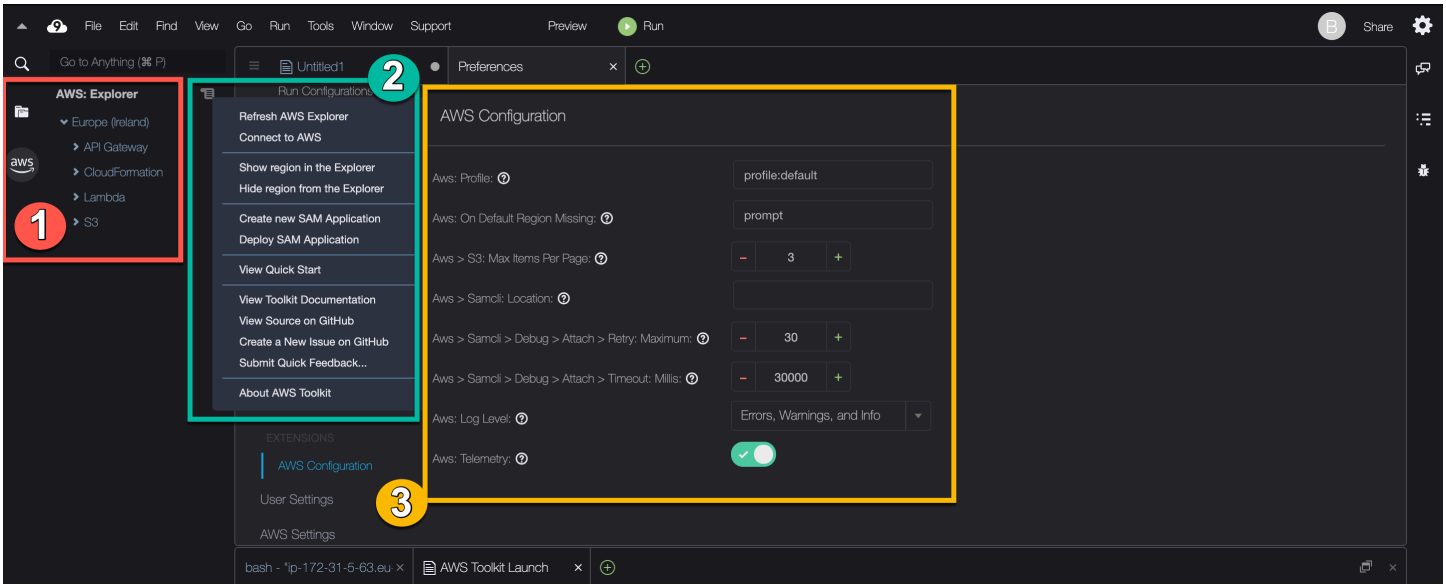
Anda juga dapat menggunakan peran IAM untuk mengelola kredensional sementara untuk aplikasi yang berjalan pada sebuah instance. EC2 Peran ini menyediakan izin sementara yang dapat digunakan aplikasi saat mereka melakukan panggilan ke AWS sumber daya lain. Saat meluncurkan EC2 instance, Anda menentukan peran IAM untuk dikaitkan dengan instance. Aplikasi yang berjalan pada instance kemudian dapat menggunakan kredenal sementara yang disediakan peran saat membuat permintaan API terhadap. Layanan AWS

Setelah Anda membuat peran, tetapkan peran ini dan izin terkait ke instance dengan membuat profil instance. Profil instans dilampirkan ke instans dan dapat memberikan kredensial sementara peran ke aplikasi yang berjalan pada instans.

Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon di Panduan Pengguna IAM.

# Mengidentifikasi AWS komponen Toolkit

Tangkapan layar berikut menunjukkan tiga komponen UI utama dari AWS Toolkit.




1. AWS Jendela Explorer: Digunakan untuk berinteraksi dengan Layanan AWS yang dapat diakses melalui Toolkit. Anda dapat beralih antara menampilkan dan menyembunyikan AWS Explorer menggunakan AWS opsi di sisi kiri lingkungan pengembangan terintegrasi (IDE). Untuk selengkapnya tentang menggunakan komponen antarmuka ini dan mengakses Layanan AWS untuk yang berbeda Wilayah AWS, lihatMenggunakan AWS Explorer untuk bekerja dengan layanan dan sumber daya di beberapa Wilayah.
2. Menu Toolkit: Digunakan untuk mengelola koneksi ke AWS, menyesuaikan tampilan jendela AWS Explorer, membuat dan menyebarkan aplikasi tanpa server, bekerja dengan GitHub repositori, dan mengakses dokumentasi. Untuk informasi selengkapnya, lihat Mengakses dan menggunakan menu AWS Toolkit.
3. AWS Panel konfigurasi: Digunakan untuk menyesuaikan perilaku yang berinteraksi dengan Anda menggunakan Toolkit. Layanan AWS Untuk informasi selengkapnya, lihat Memodifikasi pengaturan AWS Toolkit menggunakan panel Konfigurasi AWS.

# Menonaktifkan Toolkit AWS

Anda dapat menonaktifkan AWS Toolkit di tab Preferensi.

Untuk menonaktifkan AWS Toolkit

1. Pilih AWS Cloud9, Preferensi pada bilah menu.
2. Pada tab Preferensi, di panel navigasi samping, pilih AWS Pengaturan.
3. Di panel AWS Sumber Daya, matikan AWS AWS Toolkit.

   Saat Anda menonaktifkan AWS Toolkit, lingkungan pengembangan terintegrasi (IDE) akan menyegarkan untuk menghapus opsi AWS Toolkit di sisi IDE di bawah opsi Lingkungan.

# AWS Topik Toolkit

- Menavigasi dan mengkonfigurasi Toolkit AWS
- Menggunakan AWS App Runner dengan AWS Toolkit
- Bekerja dengan API Gateway menggunakan AWS Toolkit
- Bekerja dengan AWS CloudFormation tumpukan menggunakan Toolkit AWS
- Bekerja dengan AWS Lambda fungsi menggunakan AWS Toolkit
- Bekerja dengan sumber daya
- Bekerja dengan Amazon S3 menggunakan Toolkit AWS
- Bekerja dengan AWS SAM menggunakan AWS Toolkit
- Bekerja dengan Amazon CodeCatalyst
- ???

# Menavigasi dan mengkonfigurasi Toolkit AWS

Anda dapat mengakses sumber daya dan memodifikasi pengaturan melalui elemen antarmuka AWS Toolkit berikut:

- AWS Jendela Explorer: Akses Layanan AWS dari yang berbeda Wilayah AWS.
- AWS Menu Toolkit: Membuat dan menyebarkan aplikasi tanpa server, menampilkan atau menyembunyikan, mengakses bantuan pengguna Wilayah AWS, dan berinteraksi dengan repositori Git.
- AWS Panel konfigurasi: Ubah pengaturan yang memengaruhi cara Anda berinteraksi Layanan AWS di AWS Toolkit.

# Menggunakan AWS Explorer untuk bekerja dengan layanan dan sumber daya di beberapa Wilayah

Dengan jendela AWS Explorer, Anda dapat memilih Layanan AWS dan bekerja dengan sumber daya tertentu yang terkait dengan layanan itu. Di AWS Explorer, pilih node nama layanan (misalnya, API Gateway atau Lambda). Kemudian, pilih sumber daya tertentu yang terkait dengan layanan tersebut (misalnya, REST API atau fungsi Lambda). Saat Anda memilih sumber daya tertentu, menu menampilkan opsi interaksi yang tersedia seperti mengunggah atau mengunduh, memanggil, atau menyalin.

Pertimbangkan contoh berikut. Jika Akun AWS kredensional Anda dapat mengakses fungsi Lambda, perluas node Lambda yang terdaftar untuk sebuah Wilayah AWS, lalu pilih fungsi Lambda tertentu yang akan dipanggil atau diunggah sebagai kode ke IDE. AWS Cloud9 Anda juga dapat membuka menu konteks (klik kanan) untuk judul node untuk mulai membuat aplikasi yang menggunakan. AWS Serverless Application Model

> **Note**
>
> Jika Anda tidak dapat melihat opsi untuk melihat jendela AWS Explorer di lingkungan pengembangan terintegrasi (IDE), verifikasi bahwa Anda mengaktifkan AWS Toolkit. Kemudian, setelah Anda memverifikasi itu diaktifkan, coba lagi. Untuk informasi selengkapnya, lihat Mengaktifkan Toolkit AWS.

Jendela AWS Explorer juga dapat menampilkan layanan yang dihosting dalam beberapa Wilayah AWS.

Untuk mengakses Layanan AWS dari Wilayah yang dipilih

1. Di jendela Explorer AWS , pilih menu Alat kit, Tampilkan wilayah di Explorer.
2. Dari Pilih wilayah untuk ditampilkan dalam daftar AWS Explorer, pilih Wilayah AWS.

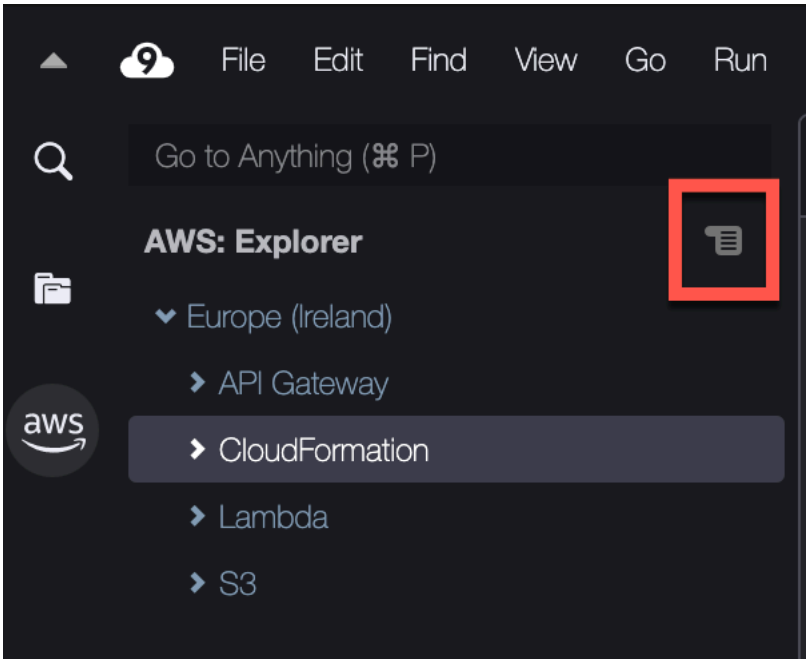
   Wilayah yang dipilih ditambahkan ke jendela AWS Explorer. Untuk mengakses layanan dan sumber daya yang tersedia, pilih panah (>) di depan nama Wilayah .

> **Note**
>
> Anda juga dapat menyembunyikan yang dipilih Wilayah AWS di jendela AWS Explorer menggunakan opsi berikut:
>
> - Buka menu konteks (klik kanan) untuk Wilayah dan pilih Sembunyikan wilayah dari Explorer.
> - Di menu AWS Toolkit, pilih Sembunyikan wilayah dari Explorer dan pilih Wilayah untuk disembunyikan.

## Mengakses dan menggunakan menu AWS Toolkit

Kit alat AWS  ini menyediakan akses untuk opsi untuk membuat dan menerapkan aplikasi nirserver. Anda dapat menggunakan menu ini untuk mengelola koneksi, memperbarui jendela AWS: Explorer, mengakses dokumentasi, dan berinteraksi dengan GitHub repositori.

Untuk mengakses menu Kit alat, pilih ikon gulir di seberang : judul AWS: Explorer di jendela AWS Explorer.




Tabel berikut memberikan ikhtisar opsi yang tersedia di menu Toolkit.

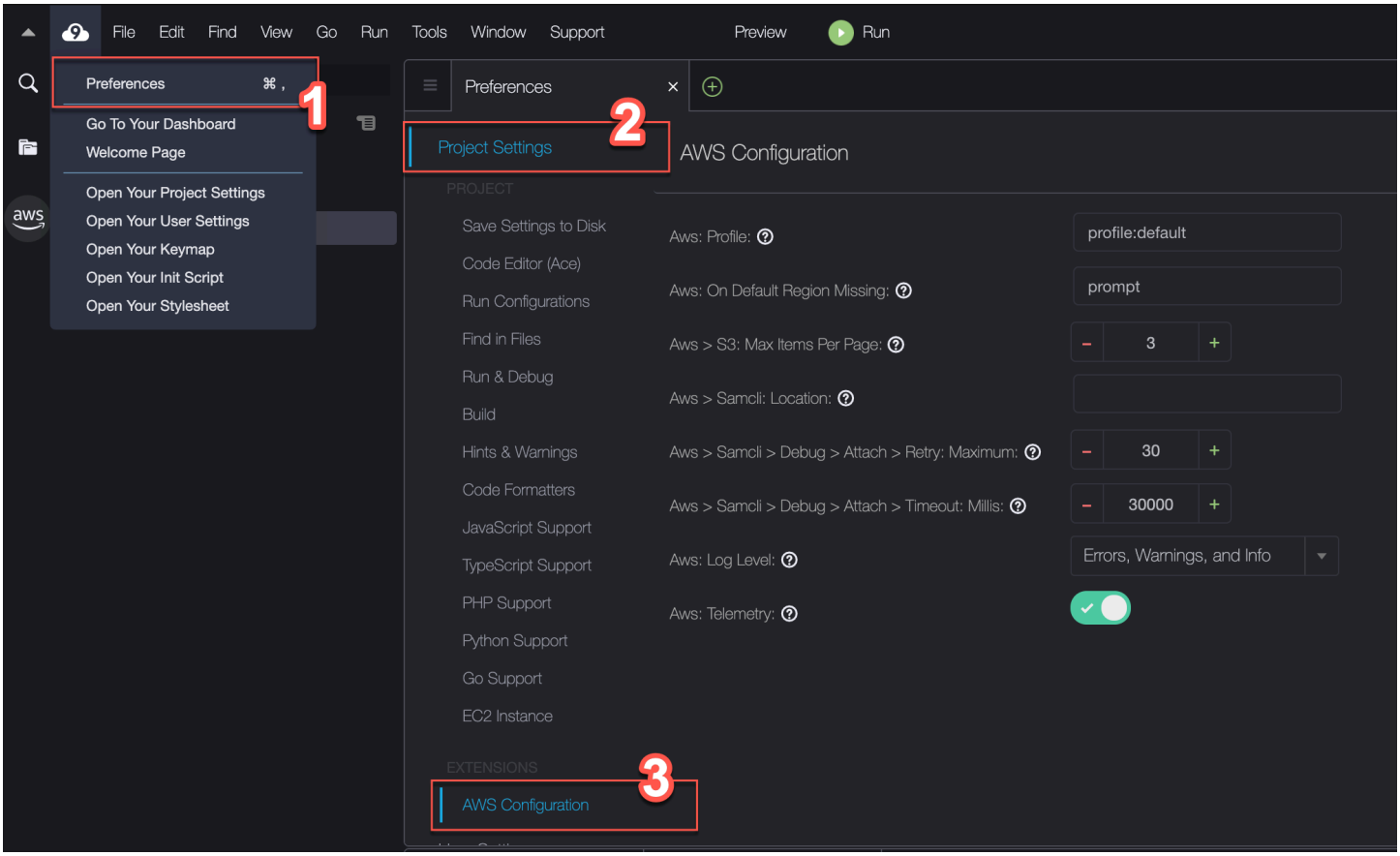
Opsi menu Toolkit

| Pilihan menu | Deskripsi |
| --- | --- |
| Segarkan AWS Penjelajah | Pilih opsi ini untuk menyegarkan AWS Explorer untuk menampilkan apa pun Layanan AWS yang telah dimodifikasi sejak Anda terakhir membuka jendela. |
| Connect ke AWS | Menghubungkan AWS Toolkit ke kredenal Akun AWS menggunakan yang disimpan dalam profil. Untuk informasi selengkapnya, lihat Mengelola kredensi akses untuk Toolkit AWS. |
| Tampilkan wilayah di Explorer | Menampilkan Wilayah AWS di jendela AWS Explorer. Untuk informasi selengkapnya, lihat Menggunakan AWS Explorer untuk bekerja dengan layanan dan sumber daya di beberapa Wilayah. |
| Sembunyikan wilayah dari Explorer | Menyembunyikan Wilayah AWS di jendela AWS Explorer. Untuk informasi selengkapnya, silakan lihat Menggunakan AWS Explorer untuk bekerja dengan layanan dan sumber daya di beberapa Wilayah |
| Buat Aplikasi SAM baru | Menghasilkan satu set file kode untuk aplikasi AWS tanpa server baru. Untuk informasi selengkapnya tentang cara membuat dan menerapkan aplikasi SAM, lihatBekerja dengan AWS SAM menggunakan AWS Toolkit. |
| Menyebarkan Aplikasi SAM | Menyebarkan aplikasi tanpa server ke. AWS Untuk informasi selengkapnya tentang cara membuat dan menerapkan aplikasi SAM, lihatBekerja dengan AWS SAM menggunakan AWS Toolkit. |
| Lihat Mulai Cepat | Buka panduan Quick Start. |

| Pilihan menu | Deskripsi |
| --- | --- |
| Lihat Dokumentasi Toolkit | Membuka panduan pengguna untuk AWS Toolkit. |
| Lihat Sumber di GitHub | Membuka GitHub repositori untuk Toolkit. AWS |
| Buat Edisi Baru di GitHub | Membuka halaman Masalah Baru AWS Toolkit di Github |
| Kirim Umpan Balik Cepat | Kirimkan umpan balik pribadi satu arah ke tim pengembangan AWS Toolkit. Untuk masalah yang memerlukan percakapan atau perbaikan bug, kirimkan masalah di Github dengan memilih opsi menu Buat Masalah Baru di Github. |
| Tentang AWS Toolkit | Menampilkan informasi tentang versi Toolkit yang berjalan dan sistem operasi Amazon yang dikonfigurasi untuknya. |

## Memodifikasi pengaturan AWS Toolkit menggunakan panel Konfigurasi AWS

Untuk mengakses panel Konfigurasi AWS , pilih AWS Cloud9, Preferensi. Selanjutnya, di jendela Preferensi, di bawah Pengaturan Proyek, pilih Konfigurasi AWS .




Tabel berikut memberikan ikhtisar opsi yang tersedia di panel AWS Konfigurasi.

| Pilihan menu | Deskripsi |
| --- | --- |
| AWS: Profil | Menetapkan nama profil kredensial untuk mendapatkan kredensial dari sana. |
| AWS: Pada Wilayah Default Hilang | Menunjukkan tindakan yang akan diambil jika default Wilayah AWS untuk profil kredensio nal yang dipilih tidak tersedia di jendela AWS Explorer. Anda dapat memilih dari tiga opsi:<br>• prompt (default): Anda ditanya apa yang ingin Anda lakukan.<br>• add: Region ditampilkan di jendela AWS Explorer. |

| Pilihan menu | Deskripsi |
| --- | --- |
| | • mengabaikan: Tidak ada tindakan yang diambil. |
| AWS > S3: Item Maks Per Halaman | Menentukan berapa banyak objek Amazon S3 atau folder yang ditampilkan pada satu waktu di jendela Explorer AWS . Ketika jumlah maksimum ditampilkan, Anda dapat memilih Tambah Muatan untuk menampilkan batch berikutnya.<br><br>Rentang nilai yang diterima untuk bidang ini adalah antara 3 dan 1000. Pengaturan ini hanya berlaku untuk jumlah objek atau folder yang ditampilkan pada satu waktu. Semua bucket yang telah Anda buat akan ditampilk an sekaligus. Secara default, Anda dapat membuat hingga 100 ember di masing-masing ember. Akun AWS |
| AWS > Samcli: Lokasi | Menunjukkan lokasi SAM CLI yang digunakan untuk membuat, membangun, memaketkan, dan men-deploy aplikasi nirserver. |
| AWS > Produk > Debug > Lampirkan> Coba lagi: Maksimum: | Menentukan berapa kali Kit alat mencoba untuk melampirkan debugger SAM CLI sebelum menyerah. Kuota default adalah 30 percobaan.<br><br>Ketika Anda secara lokal memanggil fungsi Lambda dalam mode debug dalam AWS SAMCLI, Anda kemudian dapat melampirkan debugger ke sana. |

| Pilihan menu | Deskripsi |
| --- | --- |
| AWS > Produk > Debug > Lampiran> Timeout: Millis: | Menentukan berapa lama Kit alat mencoba untuk melampirkan debugger SAM CLI sebelum menyerah. Batas waktu default adalah 30.000 milidetik (30 detik).<br><br>Ketika Anda secara lokal memanggil fungsi Lambda dalam mode debug dalam AWS SAMCLI, Anda kemudian dapat melampirkan debugger ke sana. |
| AWS : Tingkat Log: | Menetapkan kategori peristiwa alur kerja yang dicatat. Berikut ini adalah level yang tersedia:<br><br>• Kesalahan Hanya<br>• Kesalahan dan Peringatan<br>• Kesalahan, Peringatan, dan Info (opsi default)<br>• Kesalahan, Peringatan, dan Info, Verbose, dan Debug |
| AWS : Telemetri | Mengaktifkan atau menonaktifkan pengiriman data penggunaan ke AWS. Diaktifkan secara default |

# Bekerja dengan API Gateway menggunakan AWS Toolkit

Anda dapat menggunakan API Gateway untuk membuat RESTful APIs dan WebSocket APIs mengaktifkan aplikasi komunikasi dua arah secara real-time. Untuk informasi selengkapnya tentang cara membuat dan mengelola APIs dengan API Gateway, lihat Panduan Pengembang API Gateway.

Dengan AWS Toolkit, Anda dapat mengonfigurasi panggilan ke REST API dengan menentukan sumber daya REST, tipe metode, dan data yang diteruskan sebagai input.

# Memanggil REST APIs di API Gateway

> **Important**
>
> Memanggil metode API menggunakan AWS Toolkit dapat mengakibatkan perubahan pada sumber daya yang tidak dapat dibatalkan. Misalnya, jika Anda memanggil metode `POST`, sumber daya API diperbarui jika panggilan berhasil.

Anda dapat mengaktifkan API Gateway AWS dari AWS Toolkit.

Untuk memanggil REST API

1. Di jendela AWS Explorer, pilih node API Gateway untuk melihat daftar REST APIs yang tersedia saat ini Wilayah AWS.
2. Klik kanan REST API, dan kemudian pilih Panggilan pada AWS.

   > **Note**
   >
   > Anda dapat menggunakan menu konteks untuk menyalin URL, nama, dan Nama Sumber Daya Amazon (ARN) REST API.

   Jendela metode Invoke ditampilkan. Anda dapat mengonfigurasi panggilan ke API.
3. Untuk Pilih sumber daya, pilih sumber daya REST yang ingin Anda berinteraksi.
4. Untuk Pilih metode, pilih salah satu jenis metode berikut:

   - GET: Mendapatkan sumber daya dari layanan backend yang diakses melalui API.
   - OPTIONS: Meminta informasi tentang metode dan operasi yang didukung oleh API Gateway.
   - POST: Membuat sumber daya baru pada layanan backend yang diakses melalui API.
5. Untuk memberikan input ke panggilan metode API, Anda dapat menggunakan string kueri atau muatan berformat JSON:

   - String kueri: Masukkan string kueri menggunakan format:`parameter1=value1&parameter2=value2`. (Sebelum Anda menggunakan string kueri, buat templat pemetaan untuk mengubah permintaan web yang masuk sebelum dikirim ke bagian belakang integrasi.)

- Format JSON: Anda dapat menentukan muatan berformat JSON di bidang teks besar di jendela Metode pemanggilan.

  Misalnya, Anda dapat menambahkan sumber daya baru dengan metode `POST` yang berisi muatan berikut:

  ```
  {"type": "soda", "price" : 3.99}
  ```

6. Pilih tombol Panggil untuk memanggil sumber daya REST API.

   Respons REST API ditampilkan di tab Pemanggilan Jarak Jauh AWS . Isi respons berisi data sumber daya berformat JSON.

# Menggunakan AWS App Runner dengan AWS Toolkit

AWS App Runnermenyediakan cara cepat dan hemat biaya untuk menyebarkan dari kode sumber atau gambar kontainer langsung ke aplikasi web yang dapat diskalakan dan aman di. AWS Cloud Dengan menggunakannya, Anda tidak perlu mempelajari teknologi baru, memutuskan layanan komputasi mana yang akan digunakan, atau mengetahui cara menyediakan dan mengonfigurasi AWS sumber daya.

Anda dapat menggunakan AWS App Runner untuk membuat dan mengelola layanan berdasarkan gambar sumber atau kode sumber. Jika Anda menggunakan gambar sumber, Anda dapat memilih gambar kontainer publik atau pribadi yang disimpan dalam repositori gambar. App Runner mendukung penyedia repositori gambar berikut:

- Amazon Elastic Container Registry (Amazon ECR): Menyimpan gambar pribadi di file Anda. Akun AWS
- Amazon Elastic Container Registry Public (Amazon ECR Public): Menyimpan gambar yang dapat dibaca publik.

Jika Anda memilih opsi kode sumber, Anda dapat menerapkan dari repositori kode sumber yang dikelola oleh penyedia repositori yang didukung. Saat ini, App Runner mendukung GitHubsebagai penyedia repositori kode sumber.

## Prasyarat

Untuk berinteraksi dengan App Runner menggunakan AWS Toolkit memerlukan hal berikut:

- Sebuah Akun AWS
- Versi AWS Toolkit yang memiliki fitur AWS App Runner

Selain persyaratan inti tersebut, pastikan bahwa semua pengguna IAM yang relevan memiliki izin untuk berinteraksi dengan layanan App Runner. Pastikan juga untuk mendapatkan informasi spesifik tentang sumber layanan Anda seperti URI image container dan koneksi ke GitHub repositori. Anda memerlukan informasi ini saat membuat layanan App Runner.

Mengkonfigurasi izin IAM untuk App Runner

Untuk memberikan izin yang diperlukan untuk App Runner dengan cepat, lampirkan kebijakan AWS terkelola yang ada ke entitas yang relevan AWS Identity and Access Management (IAM). Secara khusus, Anda dapat melampirkan kebijakan ke pengguna atau grup. App Runner menyediakan dua kebijakan terkelola yang dapat Anda lampirkan ke pengguna IAM Anda:

- `AWSAppRunnerFullAccess`: Memungkinkan pengguna untuk melakukan semua tindakan App Runner.
- `AWSAppRunnerReadOnlyAccess`: Izinkan pengguna untuk membuat daftar dan melihat detail tentang sumber daya Pelari Aplikasi.

Jika Anda memilih repositori pribadi dari Amazon Elastic Container Registry (Amazon ECR) sebagai sumber layanan, Anda juga harus membuat peran akses berikut untuk layanan App Runner Anda:

- `AWSAppRunnerServicePolicyForECRAccess`: Memungkinkan App Runner mengakses gambar Amazon Elastic Container Registry (Amazon ECR) di akun Anda.

Anda dapat membuat peran ini secara otomatis saat mengonfigurasi instance layanan Anda dengan panel perintah AWS Toolkit.

> **Note**
>
> Peran AWSServiceRoleForAppRunnerterkait layanan memungkinkan AWS App Runner untuk menyelesaikan tugas-tugas berikut:
>
> - Dorong log ke grup CloudWatch log Amazon Logs.
> - Buat aturan Amazon CloudWatch Events untuk berlangganan push image Amazon Elastic Container Registry (Amazon ECR).

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda membuat AWS App Runner di dalam AWS Management Console atau dengan menggunakan operasi API yang dipanggil oleh AWS Toolkit, AWS App Runner buat peran terkait layanan ini untuk Anda.

Untuk informasi selengkapnya, lihat Identitas dan manajemen akses untuk Pelari Aplikasi di Panduan AWS App Runner Pengembang.

Memperoleh sumber layanan untuk App Runner

Anda dapat menggunakan AWS App Runner untuk menyebarkan layanan dari gambar sumber atau kode sumber.

Source image

Jika Anda menerapkan dari gambar sumber, dapatkan tautan ke repositori untuk gambar tersebut dari registri gambar pribadi atau publik AWS .

- Registri pribadi Amazon ECR: Salin URI untuk repositori pribadi yang menggunakan konsol Amazon ECR di repositori. https://console.aws.amazon.com/ecr/
- Registri publik Amazon ECR: Salin URI untuk repositori publik yang menggunakan Galeri Publik Amazon ECR di https://gallery.ecr.aws/.

> **Note**
>
> Anda juga dapat memperoleh URI untuk repositori Amazon ECR pribadi langsung dari AWS Explorer di Toolkit: AWS
>
> - Buka AWS Explorer dan perluas node ECR untuk melihat daftar repositori untuk itu. Wilayah AWS
> - Buka menu konteks (klik kanan) untuk repositori dan pilih Salin Repositori URI untuk menyalin tautan ke clipboard Anda.

Anda menentukan URI untuk repositori gambar saat mengonfigurasi instance layanan Anda dengan panel perintah AWS Toolkit.

Untuk informasi selengkapnya, lihat Layanan Pelari Aplikasi berdasarkan gambar sumber di Panduan AWS App Runner Pengembang.

Source code

Agar kode sumber Anda dapat digunakan ke AWS App Runner layanan, kode tersebut harus disimpan dalam repositori Git. Repositori Git ini harus dikelola oleh penyedia repositori yang didukung. App Runner mendukung satu penyedia repositori kode sumber:. GitHub

Untuk informasi tentang menyiapkan GitHub repositori, lihat dokumentasi Memulai. GitHub

Untuk menerapkan kode sumber Anda ke layanan App Runner dari GitHub repositori, App Runner membuat sambungan ke. GitHub Jika repositori Anda bersifat pribadi (artinya, repositori tidak dapat diakses publik GitHub), Anda harus memberikan detail koneksi kepada App Runner.

> **Important**
>
> Untuk membuat GitHub koneksi, Anda harus menggunakan konsol App Runner (https://console.aws.amazon.com/apprunner) untuk membuat koneksi yang ditautkan. GitHub AWS Anda dapat memilih koneksi yang tersedia di halaman GitHubkoneksi saat mengonfigurasi instance layanan Anda dengan panel perintah AWS Toolkit.
> Untuk informasi selengkapnya, lihat Mengelola koneksi App Runner di Panduan AWS App Runner Pengembang.

Instans layanan App Runner menyediakan runtime terkelola yang memungkinkan kode Anda dibuat dan dijalankan. AWS App Runner saat ini mendukung runtime berikut:

- Runtime terkelola Python
- Node.js mengelola runtime

Sebagai bagian dari konfigurasi layanan, Anda memberikan informasi tentang cara layanan App Runner membangun dan memulai layanan Anda. Anda dapat memasukkan informasi ini menggunakan Command Palette atau menentukan file konfigurasi App Runner yang diformat YAML. Nilai dalam file ini menginstruksikan App Runner cara membuat dan memulai layanan Anda, serta menyediakan konteks runtime. Ini termasuk pengaturan jaringan yang relevan dan variabel lingkungan. File konfigurasi diberi nama`apprunner.yaml`. Ini secara otomatis ditambahkan ke direktori root repositori aplikasi Anda.

# Harga

Anda dikenakan biaya untuk sumber daya komputasi dan memori yang digunakan aplikasi Anda. Selain itu, jika Anda mengotomatiskan penerapan, Anda juga membayar biaya bulanan yang ditetapkan untuk setiap aplikasi yang mencakup semua penerapan otomatis untuk bulan itu. Jika Anda memilih untuk menerapkan dari kode sumber, Anda membayar biaya pembuatan untuk waktu yang dibutuhkan App Runner untuk membuat container dari kode sumber Anda.

Untuk informasi selengkapnya, silakan lihat Harga AWS App Runner.

Topik

- Membuat layanan App Runner
- Mengelola layanan App Runner

# Membuat layanan App Runner

Anda dapat membuat layanan App Runner di AWS Toolkit dengan menggunakan Explorer.AWS Setelah Anda memilih untuk membuat layanan tertentu Wilayah AWS, panel perintah AWS Toolkit menjelaskan cara mengonfigurasi instance layanan tempat aplikasi Anda berjalan.

Sebelum Anda membuat layanan App Runner, pastikan Anda menyelesaikan prasyarat. Ini termasuk memberikan izin IAM yang relevan dan mengonfirmasi repositori sumber tertentu yang ingin Anda terapkan.

Untuk membuat layanan App Runner

1. Buka AWS Explorer, jika belum terbuka.
2. Klik kanan node App Runner dan pilih Create Service.

   Panel perintah AWS Toolkit ditampilkan.
3. Untuk Pilih jenis lokasi kode sumber, pilih ECR atau Repositori.

   Jika Anda memilih ECR, Anda menentukan gambar kontainer dalam repositori yang dikelola oleh Amazon Elastic Container Registry. Jika Anda memilih Repositori, Anda menentukan repositori kode sumber yang dikelola oleh penyedia repositori yang didukung. Saat ini, App Runner mendukung GitHubsebagai penyedia repositori kode sumber.

Menerapkan dari ECR

1. Untuk Pilih atau masukkan repositori gambar, pilih atau masukkan URL repositori gambar yang dikelola oleh registri pribadi Amazon ECR Anda atau Galeri Publik Amazon ECR.

   > **Note**
   >
   > Jika Anda menentukan repositori dari Galeri Publik Amazon ECR, pastikan penerapan otomatis dimatikan. App Runner tidak mendukung penerapan otomatis untuk gambar di repositori Publik ECR.
   > Penerapan otomatis dimatikan secara default. Ini ditunjukkan ketika ikon pada header panel perintah menampilkan garis diagonal melaluinya. Jika Anda memilih untuk mengaktifkan penerapan otomatis, sebuah pesan memberi tahu Anda bahwa opsi ini dapat menimbulkan biaya tambahan.

2. Jika langkah di panel perintah melaporkan bahwa Tidak ada tag yang ditemukan, kembalilah langkah untuk memilih repositori yang berisi gambar kontainer yang ditandai.
3. Untuk Port, masukkan port IP yang digunakan oleh layanan (misalnya, port`8000`).
4. (Opsional) Untuk Konfigurasi variabel lingkungan, tentukan file yang berisi variabel lingkungan yang digunakan untuk menyesuaikan perilaku dalam instance layanan Anda.
5. Jika Anda menggunakan registri pribadi Amazon ECR, Anda memerlukan AppRunnerECRAccessperan akses ECR Peran. Peran ini memungkinkan App Runner mengakses gambar Amazon Elastic Container Registry (Amazon ECR) di akun Anda. Pilih ikon “+” pada header panel perintah untuk membuat peran ini. Jika gambar Anda disimpan di Amazon ECR Public di mana gambar tersedia untuk umum, peran akses tidak diperlukan.
6. Untuk Nama layanan Anda, masukkan nama unik dan tekan Enter. Nama tidak dapat berisi spasi.
7. Untuk konfigurasi instans Pilih, pilih kombinasi unit CPU dan memori (keduanya dalam GB) untuk instance layanan Anda.

   Ketika layanan Anda sedang dibuat, statusnya berubah dari Creating to Running.
8. Setelah layanan Anda mulai berjalan, buka menu konteks (klik kanan) untuknya dan pilih Salin URL Layanan.
9. Untuk mengakses aplikasi yang Anda gunakan, rekatkan URL yang disalin ke bilah alamat browser web Anda.

Menerapkan dari repositori jarak jauh

1. Untuk Pilih koneksi, pilih koneksi yang terhubung GitHub ke AWS. Koneksi yang tersedia untuk dipilih tercantum di halaman GitHub koneksi di konsol App Runner.
2. Untuk Pilih GitHub repositori jarak jauh, pilih atau masukkan URL untuk repositori jarak jauh.

   Repositori jarak jauh yang sudah dikonfigurasi dengan manajemen kontrol AWS Cloud9 sumber tersedia untuk dipilih. Jika repositori tidak terdaftar, Anda juga dapat menempelkan tautan ke repositori.
3. Untuk Pilih cabang, pilih cabang Git mana dari kode sumber Anda yang ingin Anda gunakan.
4. Untuk Pilih sumber konfigurasi, tentukan bagaimana Anda ingin menentukan konfigurasi runtime Anda.

   Jika Anda memilih Gunakan file konfigurasi, instance layanan Anda dikonfigurasi oleh pengaturan yang ditentukan oleh file `apprunner.yaml` konfigurasi. File ini ada di direktori root repositori aplikasi Anda.

   Jika Anda memilih Konfigurasikan semua pengaturan di sini, gunakan panel perintah untuk menentukan hal berikut:

   - Runtime: Pilih Python 3 atau Nodejs 12.
   - Perintah build: Masukkan perintah untuk membangun aplikasi Anda di lingkungan runtime instance layanan Anda.
   - Mulai perintah: Masukkan perintah untuk memulai aplikasi Anda di lingkungan runtime instance layanan Anda.
5. Untuk Port, masukkan port IP yang digunakan layanan (misalnya, port`8000`).
6. (Opsional) Untuk Konfigurasi variabel lingkungan, tentukan file yang berisi variabel lingkungan untuk menyesuaikan perilaku dalam instance layanan Anda.
7. Untuk Nama layanan Anda, masukkan nama unik dan tekan Enter. Nama tidak dapat berisi spasi.
8. Untuk konfigurasi instans Pilih, pilih kombinasi unit CPU dan memori dalam GB untuk instance layanan Anda.

   Saat layanan Anda sedang dibuat, statusnya berubah dari Creating to Running.
9. Setelah layanan Anda mulai berjalan, buka menu konteks (klik kanan) untuknya dan pilih Salin URL Layanan.

10. Untuk mengakses aplikasi yang Anda gunakan, rekatkan URL yang disalin ke bilah alamat browser web Anda.

> **Note**
>
> Jika upaya Anda untuk membuat layanan App Runner gagal, layanan akan menampilkan status Buat gagal di AWS Explorer. Untuk informasi pemecahan masalah, lihat Saat pembuatan layanan gagal di Panduan Pengembang App Runner.

## Mengelola layanan App Runner

Setelah membuat layanan App Runner, Anda dapat mengelolanya dengan menggunakan panel AWS Explorer untuk melakukan aktivitas berikut:

- Menjeda dan melanjutkan layanan App Runner
- Menyebarkan layanan App Runner
- Melihat aliran log untuk App Runner
- Menghapus layanan App Runner

### Menjeda dan melanjutkan layanan App Runner

Jika Anda perlu menonaktifkan aplikasi web sementara dan menghentikan kode agar tidak berjalan, Anda dapat menjeda layanan AWS App Runner. App Runner mengurangi kapasitas komputasi untuk layanan menjadi nol. Ketika Anda siap untuk menjalankan aplikasi Anda lagi, lanjutkan layanan App Runner Anda. App Runner menyediakan kapasitas komputasi baru, menyebarkan aplikasi Anda ke dalamnya, dan menjalankan aplikasi.

> **Important**
>
> Anda ditagih untuk App Runner hanya saat sedang berjalan. Oleh karena itu, Anda dapat menjeda dan melanjutkan aplikasi Anda sesuai kebutuhan untuk mengelola biaya. Ini sangat membantu dalam skenario pengembangan dan pengujian.

Untuk menjeda layanan App Runner

1. Buka AWS Explorer, jika belum terbuka.
2. Perluas App Runner untuk melihat daftar layanan.
3. Klik kanan layanan Anda dan pilih Jeda.
4. Di kotak dialog yang ditampilkan, pilih Konfirmasi.

   Saat layanan berhenti, status layanan berubah dari Running ke Pusing dan kemudian ke Pause.

Untuk melanjutkan layanan App Runner

1. Buka AWS Explorer, jika belum terbuka.
2. Perluas App Runner untuk melihat daftar layanan.
3. Klik kanan layanan Anda dan pilih Lanjutkan.

   Saat layanan dilanjutkan, status layanan berubah dari Melanjutkan ke Berjalan.

## Menyebarkan layanan App Runner

Jika Anda memilih opsi penerapan manual untuk layanan Anda, Anda perlu secara eksplisit memulai setiap penerapan ke layanan Anda.

1. Buka AWS Explorer, jika belum terbuka.
2. Perluas App Runner untuk melihat daftar layanan.
3. Klik kanan layanan Anda dan pilih Mulai Deployment.
4. Saat aplikasi Anda sedang di-deploy, status layanan berubah dari Deploying ke Running.
5. Untuk mengonfirmasi bahwa aplikasi Anda berhasil digunakan, klik kanan layanan yang sama dan pilih Salin URL Layanan.
6. Untuk mengakses aplikasi web yang Anda gunakan, tempel URL yang disalin ke bilah alamat browser web Anda.

## Melihat aliran log untuk App Runner

Gunakan CloudWatch Log untuk memantau, menyimpan, dan mengakses aliran log Anda untuk layanan seperti App Runner. Pengaliran log adalah urutan log acara yang berbagi sumber yang sama.

1. Perluas App Runner untuk melihat daftar instance layanan.
2. Perluas instance layanan tertentu untuk melihat daftar grup log. (Grup log adalah grup aliran log yang berbagi pengaturan retensi, pemantauan, dan kontrol akses yang sama.)
3. Klik kanan grup log dan pilih Lihat Aliran Log.
4. Dari panel perintah, pilih aliran log dari grup.

   AWS Cloud9 IDE menampilkan daftar peristiwa log yang membentuk aliran. Anda dapat memilih untuk memuat acara lama atau yang lebih baru ke editor.

### Menghapus layanan App Runner

> **Important**
>
> Jika Anda menghapus layanan App Runner, layanan tersebut akan dihapus secara permanen dan data yang disimpan akan dihapus. Jika Anda perlu membuat ulang layanan, App Runner perlu mengambil sumber Anda lagi dan membangunnya jika itu adalah repositori kode. Aplikasi web Anda mendapatkan domain App Runner baru.

1. Buka AWS Explorer, jika belum terbuka.
2. Perluas App Runner untuk melihat daftar layanan.
3. Klik kanan layanan dan pilih Hapus Layanan.
4. Di panel perintah AWS Toolkit, masukkan hapus lalu tekan Enter untuk mengonfirmasi.

   Layanan yang dihapus menampilkan status Menghapus, dan kemudian layanan menghilang dari daftar.

## Bekerja dengan AWS CloudFormation tumpukan menggunakan Toolkit AWS

AWS Toolkit menyediakan dukungan untuk AWS CloudFormationtumpukan. Menggunakan AWS Toolkit, Anda dapat menghapus AWS CloudFormation tumpukan.

## Menghapus tumpukan AWS CloudFormation

Anda dapat menggunakan AWS Toolkit untuk melihat dan menghapus AWS CloudFormation tumpukan.

### Prasyarat

- Pastikan bahwa kredensi yang Anda gunakan di AWS Cloud9 lingkungan mencakup akses baca/tulis yang sesuai ke layanan. AWS CloudFormation Jika di AWS Explorer, di bawah CloudFormation, Anda melihat pesan yang mirip dengan “Error loading CloudFormation resources,” periksa izin yang dilampirkan pada kredensi tersebut. Perubahan yang Anda buat pada izin membutuhkan waktu beberapa menit untuk memengaruhi AWS Explorer.

Untuk menghapus AWS CloudFormation tumpukan

1. Di AWS Explorer, buka menu konteks (klik kanan) AWS CloudFormation tumpukan yang ingin Anda hapus.
2. Pilih Hapus CloudFormation Tumpukan.
3. Pada pesan yang muncul, pilih Ya untuk mengonfirmasi penghapusan.

Setelah tumpukan dihapus, itu tidak lagi tercantum di AWS Explorer.

# Bekerja dengan CloudWatch Log menggunakan AWS Toolkit

Anda dapat menggunakan Amazon CloudWatch Logs untuk memusatkan log dari semua sistem dan aplikasi Anda dan Layanan AWS yang Anda gunakan, dalam satu layanan yang sangat skalabel. Anda kemudian dapat dengan mudah melihat log, mencari adanya kode atau pola kesalahan tertentu, memfilter berdasarkan bidang tertentu, atau mengarsipkannya dengan aman untuk analisis pada masa mendatang. Untuk informasi selengkapnya, lihat Apa itu Amazon CloudWatch Logs? di Panduan CloudWatch Pengguna Amazon.

Topik berikut menjelaskan cara menggunakan AWS Toolkit untuk bekerja dengan CloudWatch Log di AWS akun:

Topik

- Melihat grup CloudWatch log dan aliran log menggunakan Toolkit AWS
- Bekerja dengan peristiwa CloudWatch log di aliran log

# Melihat grup CloudWatch log dan aliran log menggunakan Toolkit AWS

Pengaliran log adalah urutan log acara yang berbagi sumber yang sama. Setiap sumber log yang terpisah ke dalam CloudWatch Log membentuk aliran log terpisah.

Grup log adalah grup pengaliran log yang berbagi pengaturan retensi, pemantauan, dan kontrol akses yang sama. Anda dapat menentukan grup log dan menentukan pengaliran untuk dimasukkan ke dalam setiap grup. Tidak ada batasan jumlah aliran log yang dapat dimiliki oleh satu grup log.

Untuk informasi selengkapnya, lihat Bekerja dengan Grup Log dan Aliran Log di Panduan CloudWatch Pengguna Amazon.

Topik


- Melihat grup log dan aliran log dengan node CloudWatch Log


## Melihat grup log dan aliran log dengan node CloudWatch Log

1. Buka AWS Explorer, jika belum terbuka.
2. Klik node CloudWatch Log untuk memperluas daftar grup log.

   Grup log untuk saat ini Wilayah AWS ditampilkan di bawah node CloudWatch Log.
3. Untuk melihat aliran log dalam grup log tertentu, buka menu konteks (klik kanan) untuk nama grup log, lalu pilih Lihat Aliran Log.
4. Konten grup log ditampilkan di bawah judul Pilih pengaliran log.

   Anda dapat memilih aliran tertentu dari daftar atau memfilter aliran dengan memasukkan teks di bidang.

   Setelah Anda memilih aliran, peristiwa dalam aliran tersebut akan ditampilkan di jendela Aliran Log IDE. Untuk informasi tentang berinteraksi dengan peristiwa log di setiap aliran, lihat Bekerja dengan peristiwa CloudWatch log.

# Bekerja dengan peristiwa CloudWatch log di aliran log

Setelah Anda membuka jendela Aliran Log, Anda dapat mengakses peristiwa log di setiap aliran. Peristiwa log adalah catatan aktivitas yang direkam oleh aplikasi atau sumber daya yang dipantau.

Topik

- Melihat dan menyalin informasi aliran log
- Menyimpan konten dari editor aliran log ke file lokal

## Melihat dan menyalin informasi aliran log

Ketika Anda membuka aliran log, jendela Aliran Log menampilkan urutan aliran peristiwa log tersebut.

1. Untuk menemukan aliran log untuk dilihat, buka jendela Log Stream. Untuk informasi selengkapnya, lihat Melihat grup CloudWatch log dan aliran log.

   Setiap baris yang mencantumkan suatu peristiwa diberi stempel waktu untuk menunjukkan kapan peristiwa itu dicatat.
2. Anda dapat melihat dan menyalin informasi tentang peristiwa aliran menggunakan opsi berikut:

   - Lihat peristiwa berdasarkan waktu: Menampilkan peristiwa log terbaru dan yang lebih lama dengan memilih Muat peristiwa yang lebih baru atau Muat peristiwa lama.

     > **Note**
     >
     > Editor Log Stream awalnya memuat batch dari 10.000 baris peristiwa log terbaru atau 1 MB data log, mana yang lebih kecil. Jika Anda memilih Muat peristiwa yang lebih baru, editor menampilkan peristiwa yang dicatat setelah batch terakhir dimuat. Jika Anda memilih Muat peristiwa yang lebih lama, editor menampilkan batch peristiwa yang terjadi sebelum peristiwa yang saat ini ditampilkan.

   - Salin peristiwa log: Pilih acara yang akan disalin, lalu buka menu konteks (klik kanan) dan pilih Salin dari menu.
   - Salin nama aliran log: Buka menu konteks (klik kanan) untuk tab jendela Aliran Log dan pilih Salin Nama Aliran Log.

## Menyimpan konten dari editor aliran log ke file lokal

Anda dapat mengunduh konten editor aliran CloudWatch log ke `log` file di mesin lokal Anda.

> **Note**
>
> Anda dapat menggunakan opsi ini untuk menyimpan hanya untuk mengajukan peristiwa log yang saat ini ditampilkan di editor aliran log. Misalnya, misalkan ukuran total aliran log adalah

5MB dan hanya 2MB yang dimuat di editor. File Anda yang disimpan juga hanya berisi 2MB data log. Untuk menampilkan lebih banyak data yang akan disimpan, pilih Muat peristiwa yang lebih baru atau Muat peristiwa yang lebih lama dalam editor.

1. Untuk menemukan aliran log yang akan disalin, buka jendela Aliran Log (lihat Melihat grup CloudWatch log dan aliran log).
2. Buka menu konteks (klik kanan) untuk tab jendela Aliran Log dan pilih Simpan Konten Log Saat Ini ke File
3. Gunakan kotak dialog untuk memilih atau membuat folder unduhan untuk file log, dan pilih Simpan.

# Bekerja dengan AWS Lambda fungsi menggunakan AWS Toolkit

AWS Toolkit mendukung AWS Lambdafungsi. AWS Toolkit menggantikan fungsionalitas yang sebelumnya disediakan oleh plug-in Lambda. AWS Cloud9 Menggunakan AWS Toolkit, Anda dapat membuat kode untuk fungsi Lambda yang merupakan bagian dari aplikasi tanpa server. Selain itu, Anda dapat memanggil fungsi Lambda baik secara lokal atau pada AWS.

Lambda adalah layanan komputasi yang dikelola sepenuhnya yang menjalankan kode Anda sebagai respons terhadap peristiwa yang dihasilkan oleh kode kustom atau dari berbagai. Layanan AWS Mereka termasuk Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB, Amazon Kinesis, Amazon Simple Notification Service (Amazon SNS), dan Amazon Cognito.

**Important**

Jika Anda ingin membangun aplikasi Lambda yang menggunakan sumber daya yang disediakan oleh Model Aplikasi Tanpa Server (SAM), lihat. Bekerja dengan AWS SAM menggunakan AWS Toolkit

Topik

- Memanggil fungsi Lambda jarak jauh
- Mengunduh, mengunggah, dan menghapus fungsi Lambda

# Memanggil fungsi Lambda jarak jauh

Menggunakan AWS Toolkit Anda dapat berinteraksi dengan AWS Lambdafungsi dalam berbagai cara.

Untuk informasi tentang Lambda, lihat AWS Lambda Panduan Developer .

> **Note**
>
> Misalkan Anda telah membuat fungsi Lambda dengan menggunakan AWS Management Console atau dengan cara lain. Anda dapat memanggilnya dari AWS Toolkit. Untuk membuat fungsi baru dengan AWS Toolkit yang dapat Anda gunakan AWS Lambda, Anda harus terlebih dahulu membuat aplikasi tanpa server.

## Prasyarat

- Pastikan bahwa kredenal yang Anda konfigurasikan termasuk akses baca/tulis yang sesuai ke layanan. AWS Lambda Jika di AWS Explorer, di bawah Lambda, Anda melihat pesan yang mirip dengan “Kesalahan memuat sumber daya Lambda,” periksa izin yang dilampirkan pada kredensil tersebut. Perubahan yang Anda buat pada izin membutuhkan waktu beberapa menit untuk memengaruhi AWS Explorer di AWS Toolkit.

## Memanggil fungsi Lambda

> **Important**
>
> Memanggil metode API menggunakan AWS Toolkit dapat mengakibatkan perubahan pada sumber daya yang tidak dapat dibatalkan. Misalnya, jika Anda memanggil metode `POST`, sumber daya API diperbarui jika panggilan berhasil.

Anda dapat menjalankan fungsi Lambda saat menggunakan AWS AWS Toolkit.

1. Di AWS Explorer, pilih nama fungsi Lambda yang ingin Anda panggil, lalu buka menu konteksnya.
2. Pilih Invoke on AWS.

3. Di jendela Fungsi pemanggilan yang terbuka, pilih opsi untuk muatan yang dibutuhkan fungsi Lambda Anda. (Muatan adalah JSON yang ingin Anda berikan ke fungsi Lambda Anda sebagai input.) Anda dapat memilih Browse untuk memilih file yang akan digunakan sebagai payload atau menggunakan bidang dropdown untuk memilih template untuk payload. Dalam hal ini, fungsi Lambda mungkin muncul sebagai string sebagai input, seperti yang ditunjukkan dalam kotak teks.

Pilih Panggil untuk memanggil Lambda dan meneruskan muatan.

Anda melihat output dari fungsi Lambda di tab. AWS Lambda

## Mengunduh, mengunggah, dan menghapus fungsi Lambda

AWS Toolkit menyediakan opsi untuk mengimpor dan mengunggah fungsi Lambda di IDE. AWS Cloud9

### Mengunduh fungsi Lambda

Dengan mengunduh fungsi Lambda, Anda juga mengunduh file proyek yang menggambarkan fungsi dari AWS Cloud dan bekerja dengannya di AWS Cloud9 IDE.

Untuk mengunduh fungsi Lambda

1. Di AWS Explorer, di bawah simpul Lambda, buka menu konteks (klik kanan) untuk fungsi tersebut, dan pilih Unduh.
2. Saat diminta untuk Pilih folder ruang kerja untuk proyek baru Anda, Anda dapat melakukan salah satu hal berikut:

   - Pilih folder yang disarankan untuk membuat subfolder dengan nama yang sama dengan proyek Lambda Anda.
   - Pilih Pilih folder lain untuk membuka kotak dialog guna menelusuri dan memilih folder induk yang berbeda untuk subfolder proyek Anda.

   IDE membuka jendela editor baru.

## Mengkonfigurasi fungsi Lambda yang diunduh untuk menjalankan dan men-debug

Untuk menjalankan dan men-debug fungsi Lambda yang diunduh sebagai aplikasi tanpa server, Anda memerlukan konfigurasi peluncuran untuk ditentukan dalam file Anda. `launch.json` Fungsi Lambda yang dibuat di dalam AWS Management Console mungkin tidak disertakan dalam konfigurasi peluncuran. Jadi, Anda mungkin perlu menambahkannya secara manual.

Untuk menambahkan fungsi Lambda Anda untuk meluncurkan konfigurasi

1. Setelah Anda mengunduh fungsi Lambda, buka jendela Lingkungan untuk melihat folder dan file-nya.
2. Selanjutnya, periksa apakah fungsi Lambda Anda disertakan dalam file. `/home/ec2-user/.c9/launch.json` Jika tidak ada, lakukan hal berikut untuk menambahkan CodeLens tautan ke kode fungsi Anda:

   1. Buka file kode sumber yang mendefinisikan fungsi Lambda (misalnya, `.js` file `.py` atau). Kemudian, periksa apakah ada CodeLens tautan yang dapat Anda gunakan untuk menambahkan fungsi lambda Anda ke `launch.json` file. A CodeLens muncul di atas fungsi dan menyertakan `Add Debug Config` tautan.
   2. Pilih Go (ikon kaca pembesar) di sebelah kiri IDE, dan masukkan “sam hint” untuk menampilkan perintah. `AWS: Toggle SAM hints in source files` Pilih perintah untuk menjalankannya.
   3. Tutup file kode sumber Lambda Anda dan kemudian buka kembali.
   4. Jika CodeLens tersedia dalam kode sumber setelah Anda membuka kembali file, pilih `Add Debug Config` untuk menambahkan konfigurasi peluncuran.
3. Jika Anda tidak dapat menambahkan CodeLens bahkan setelah mengaktifkan opsi petunjuk SAM, lakukan hal berikut untuk menambahkan konfigurasi peluncuran:

   1. Pilih Go (ikon kaca pembesar) di sebelah kiri IDE, dan ketik “config” untuk menampilkan perintah. `AWS: SAM Debug Configuration Editor` Pilih perintah untuk menjalankannya.
   2. Editor Konfigurasi Debug SAM ditampilkan. Anda dapat menggunakan editor ini untuk menentukan properti konfigurasi peluncuran. Untuk informasi, lihat langkah untuk configuring launch properties masukMenggunakan templat SAM untuk menjalankan dan men-debug aplikasi nirserver.

> **Note**
>
> Jika fungsi Lambda Anda tidak memiliki aplikasi `template.yaml` untuk SAM, Anda harus menambahkannya. Untuk informasi selengkapnya, lihat Membuat template AWS SAM Anda.

3. Setelah Anda selesai memasukkan informasi konfigurasi yang diperlukan di editor, konfigurasi peluncuran Anda ditambahkan ke file launch.json.

Setelah Anda menetapkan konfigurasi peluncuran untuk fungsi Lambda Anda, Anda dapat menjalankannya dengan melakukan hal berikut:

1. Di bagian atas IDE, pilih panah di samping Auto dan pilih konfigurasi peluncuran yang relevan.
2. Selanjutnya, pilih Jalankan.

## Mengunggah fungsi Lambda

Anda dapat memperbarui fungsi Lambda yang ada dengan kode lokal. Memperbarui kode dengan cara ini tidak menggunakan AWS Serverless Application Model CLI untuk penerapan dan tidak membuat tumpukan. AWS CloudFormation Dengan cara ini, Anda dapat mengunggah fungsi Lambda dengan runtime apa pun yang didukung oleh Lambda.

Ada beberapa opsi antarmuka untuk mengunggah fungsi Lambda menggunakan AWS Toolkit.

Unggah dari jendela Lingkungan atau panel Perintah

1. Di jendela Lingkungan untuk file proyek Anda, pilih menu konteks (klik kanan) untuk aplikasi untuk Lambda yang ingin Anda unggah dan pilih Unggah Lambda. `template.yaml`

   Atau, tekan Ctrl+P untuk membuka panel Go to Anything dan masukkan “lambda” untuk mengakses perintah Unggah Lambda.AWS  Kemudian, pilih untuk memulai proses unggah.
2. Selanjutnya, pilih Wilayah AWS yang ingin Anda unggah.
3. Sekarang pilih opsi untuk mengunggah fungsi Lambda Anda:

   Unggah arsip.zip

   1. Pilih Arsip ZIP dari menu.

2. Pilih file.zip dari sistem AWS Cloud9 file Anda dan pilih Buka.

Unggah direktori apa adanya

1. Pilih Direktori dari menu.
2. Pilih direktori dari sistem AWS Cloud9 file Anda dan pilih Buka.

4. Tentukan penangan fungsi Lambda yang memproses peristiwa. Ketika fungsi Anda dipanggil, Lambda menjalankan metode handler ini.

> **Note**
>
> Saat memilih fungsi Lambda Anda, Anda dapat memilih dari daftar yang ditampilkan. Jika Anda tidak tahu fungsi mana yang harus dipilih, Anda dapat memasukkan Amazon Resource Number (ARN) dari fungsi Lambda yang tersedia di Toolkit.

Dialog menampilkan menanyakan apakah Anda ingin kode ini dipublikasikan sebagai versi terbaru dari fungsi Lambda. Pilih Ya untuk mengonfirmasi publikasi.

> **Note**
>
> Anda juga dapat mengunggah aplikasi Lambda dengan membuka menu konteks (klik kanan) untuk folder induk pada folder dan memilih Unggah Lambda. Folder induk dipilih secara otomatis untuk diunggah.

Unggah dari AWS Explorer

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk nama fungsi Lambda yang ingin Anda impor.
2. Pilih Unggah Lambda.
3. Pilih dari tiga opsi untuk mengunggah fungsi Lambda Anda.

Unggah arsip.zip premade

1. Pilih Arsip ZIP dari menu.
2. Pilih file.zip dari sistem AWS Cloud9 file Anda dan pilih Buka.

3. Konfirmasikan unggahan dengan dialog modal. Ini mengunggah file .zip dan segera memperbarui Lambda setelah deployment.

Unggah direktori apa adanya

1. Pilih Direktori dari menu.
2. Pilih direktori dari sistem AWS Cloud9 file Anda dan pilih Buka.
3. Pilih Tidak saat diminta untuk membangun direktori.
4. Konfirmasikan unggahan dengan dialog modal. Ini mengunggah direktori apa adanya dan segera memperbarui Lambda setelah deployment.

Membangun dan mengunggah direktori

1. Pilih Direktori dari menu.
2. Pilih direktori dari sistem AWS Cloud9 file Anda dan pilih Buka.
3. Pilih Ya saat diminta untuk membangun direktori.
4. Konfirmasikan unggahan dengan dialog modal. Ini membangun kode di direktori menggunakan perintah AWS SAM `sam build` CLI dan segera memperbarui Lambda setelah penerapan.

## Menerapkan fungsi Lambda untuk akses jarak jauh

Anda dapat membuat fungsi lokal Anda tersedia dari jarak jauh dengan menerapkannya sebagai aplikasi SAM tanpa server.

Untuk menyebarkan fungsi Lambda sebagai aplikasi SAM

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk simpul Lambda, dan pilih Deploy SAM Application.
2. Di panel perintah, pilih template YAMB yang mendefinisikan fungsi Anda sebagai aplikasi tanpa server.
3. Selanjutnya, pilih bucket Amazon S3 untuk penerapan Lambda. Anda juga dapat memilih untuk membuat bucket untuk penerapan.

4. Sekarang masukkan nama AWS CloudFormation tumpukan yang Anda gunakan. Jika Anda menentukan tumpukan yang ada, perintah akan memperbarui tumpukan. Jika Anda menentukan tumpukan baru, perintah tersebut membuatnya.

   Setelah Anda memasukkan nama tumpukan, fungsi Lambda Anda mulai digunakan sebagai aplikasi SAM. Setelah penerapan berhasil, aplikasi SAM Lambda tersedia dari jarak jauh. Dengan begitu, Anda dapat mengunduh atau memanggilnya dari lingkungan AWS Cloud9 pengembangan lainnya.

Jika Anda ingin membuat fungsi Lambda dari awal, kami sarankan mengikuti langkah-langkahnya. Membuat aplikasi nirserver dengan AWS Toolkit

### Menghapus fungsi Lambda

Anda juga dapat menghapus fungsi Lambda menggunakan menu konteks (klik kanan) yang sama.

> **Warning**
>
> Jangan gunakan prosedur ini untuk menghapus fungsi Lambda yang terkait dengannya. AWS CloudFormation Misalnya, jangan hapus fungsi Lambda yang dibuat saat membuat aplikasi tanpa server sebelumnya dalam panduan ini. Fungsi-fungsi ini harus dihapus melalui tumpukan AWS CloudFormation .

1. Di AWS Explorer, pilih nama fungsi Lambda yang ingin Anda hapus, lalu buka konteksnya (menu kanan).
2. Pilih Hapus.
3. Pada pesan yang muncul, pilih Ya untuk mengonfirmasi penghapusan.

Setelah fungsi tersebut dihapus, fungsi tersebut tidak lagi tercantum dalam tampilan AWS Explorer.

## Bekerja dengan sumber daya

Selain mengakses Layanan AWS yang terdaftar secara default di AWS Explorer, Anda dapat pergi ke Sumber Daya dan memilih dari ratusan sumber daya untuk ditambahkan ke antarmuka. Di AWS, sumber daya adalah entitas yang dapat Anda gunakan. Beberapa sumber daya yang ditambahkan

termasuk Amazon AppFlow, Amazon Kinesis Data AWS Streams, peran IAM, Amazon CloudFront VPC, dan distribusi Amazon.

Untuk melihat sumber daya yang tersedia, buka Sumber daya dan perluas jenis sumber daya untuk mencantumkan sumber daya yang tersedia untuk jenis itu. Misalnya, jika Anda memilih jenis `AWS::Lambda::Function` sumber daya, Anda dapat mengakses sumber daya yang menentukan fungsi yang berbeda, propertinya, dan atributnya.

Setelah menambahkan jenis sumber daya ke Sumber Daya, Anda dapat berinteraksi dengannya dan sumber dayanya dengan cara berikut:

- Lihat daftar sumber daya yang ada yang tersedia saat ini Wilayah AWS untuk jenis sumber daya ini.
- Melihat versi read-only JSON file yang menggambarkan sumber daya.
- Salin pengenal sumber daya untuk sumber daya.
- Lihat AWS dokumentasi yang menjelaskan tujuan dari jenis sumber daya dan skema (dalam JSON and YAML format) untuk memodelkan sumber daya.

## Izin IAM untuk mengakses sumber daya

Anda memerlukan AWS Identity and Access Management izin khusus untuk mengakses sumber daya yang terkait dengannya Layanan AWS. Misalnya, entitas IAM, seperti pengguna atau peran, memerlukan izin Lambda untuk mengakses sumber daya. `AWS::Lambda::Function`

Selain izin untuk sumber daya layanan, entitas IAM memerlukan izin untuk mengizinkan AWS Toolkit memanggil operasi AWS Cloud Control API. Operasi Cloud Control API memungkinkan pengguna atau peran IAM untuk mengakses dan memperbarui sumber daya jarak jauh.

Anda dapat dengan cepat memberikan izin dengan melampirkan kebijakan AWS terkelola PowerUserAccess, ke entitas IAM yang memanggil operasi API ini menggunakan antarmuka Toolkit. Kebijakan terkelola ini memberikan berbagai izin untuk melakukan tugas pengembangan aplikasi, termasuk memanggil operasi API.

Untuk izin khusus yang menentukan operasi API yang diizinkan pada sumber daya jarak jauh, lihat Panduan Pengguna AWS Cloud Control API.

## Berinteraksi dengan sumber daya yang ada

1. Di AWS Explorer, pilih Resources.

Daftar jenis sumber daya ditampilkan di bawah node Resources.

2. Ada dokumentasi yang menjelaskan sintaks yang mendefinisikan template untuk jenis sumber daya. Untuk mengakses dokumentasi ini, buka menu konteks (klik kanan) untuk jenis sumber daya tersebut dan pilih Lihat Dokumentasi.

> **Note**
>
> Anda mungkin diminta untuk mematikan pemblokir popup browser Anda sehingga Anda dapat mengakses halaman dokumentasi.

3. Untuk melihat sumber daya yang sudah ada untuk jenis sumber daya, perluas entri untuk jenis itu.

   Daftar sumber daya yang tersedia ditampilkan di bawah jenis sumber dayanya.

4. Untuk berinteraksi dengan sumber daya tertentu, buka menu konteks (klik kanan) untuk namanya dan pilih salah satu opsi berikut:

   - Copy Identifier: Salin pengenal untuk sumber daya tertentu ke clipboard. Misalnya, sumber daya `AWS::DynamoDB::Table` dapat diidentifikasi menggunakan `TableName` properti.
   - Pratinjau: Lihat versi read-only JSON-format template yang menjelaskan sumber daya.

# Bekerja dengan Amazon S3 menggunakan Toolkit AWS

Topik berikut menjelaskan cara menggunakan AWS Toolkit untuk bekerja dengan bucket dan objek Amazon S3 di file. Akun AWS

Topik

- Bekerja dengan bucket Amazon S3
- Cara menggunakan objek Amazon S3

## Bekerja dengan bucket Amazon S3

Setiap objek yang Anda simpan di Amazon S3 berada di dalam bucket. Anda dapat menggunakan bucket untuk mengelompokkan objek terkait dengan cara yang sama dengan Anda menggunakan direktori untuk file kelompok dalam sistem file.

Topik


- Membuat sebuah bucket Amazon S3
- Menambahkan folder ke bucket Amazon S3
- Menghapus sebuah bucket Amazon S3
- Mengonfigurasi tampilan item Amazon S3


## Membuat sebuah bucket Amazon S3

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk node S3, lalu pilih Create Bucket.
2. Di bidang Nama Bucket, masukkan nama yang valid untuk bucket. Tekan Enter untuk mengonfirmasi.

   Bucket baru ditampilkan di bawah simpul S3.

   > **Catatan**
   >
   > Karena bucket S3 Anda dapat digunakan sebagai URL yang diakses publik, nama bucket yang Anda pilih harus unik secara global. Jika beberapa akun lain telah membuat bucket dengan nama yang Anda pilih, Anda harus menggunakan nama lain.
   > Jika Anda tidak dapat membuat bucket, Anda dapat memeriksa Log AWS Toolkit di tab Output. Misalnya, jika Anda menggunakan nama bucket yang sudah digunakan, terjadi `BucketAlreadyExists` kesalahan. Untuk informasi selengkapnya, lihat Pembatasan dan batasan Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

   Setelah bucket dibuat, Anda dapat menyalin nama dan Amazon Resource Name (ARN) bucket ke clipboard. Buka menu konteks (klik kanan) untuk entri bucket dan pilih opsi yang relevan dari menu.

## Menambahkan folder ke bucket Amazon S3

Anda mengatur isi bucket dengan mengelompokkan objek dalam folder. Anda juga dapat membuat folder di dalam folder lain.

1. Di AWS Explorer, pilih simpul S3 untuk melihat daftar bucket.
2. Buka menu konteks (klik kanan) untuk ember atau folder, lalu pilih Buat Folder.

3. Masukkan Nama Folder, lalu tekan Enter.

   Folder baru sekarang ditampilkan di bawah bucket dan folder yang dipilih di jendela AWS Explorer.

## Menghapus sebuah bucket Amazon S3

Saat Anda menghapus ember, Anda juga menghapus folder dan objek yang dikandungnya. Sebelum bucket dihapus, Anda diminta untuk mengonfirmasi bahwa Anda ingin melakukan ini.

> **Note**
>
> Untuk menghapus hanya sebuah folder, bukan seluruh bucket, gunakan AWS Management Console.

1. Di AWS Explorer, pilih simpul S3 untuk memperluas daftar bucket.
2. Buka menu konteks untuk bucket yang akan dihapus, lalu pilih Hapus.
3. Masukkan nama bucket untuk mengonfirmasi bahwa Anda ingin menghapusnya, lalu tekan Enter.

   > **Note**
   >
   > Jika ember berisi objek, ember dikosongkan sebelum Anda menghapusnya. Hal ini dapat memakan waktu jika perlu untuk menghapus setiap versi dari ribuan objek. Pemberitahuan ditampilkan setelah proses penghapusan selesai.

## Mengonfigurasi tampilan item Amazon S3

Jika Anda bekerja dengan sejumlah besar objek atau folder Amazon S3, akan sangat membantu untuk menentukan berapa banyak yang ditampilkan pada satu waktu. Ketika jumlah maksimum ditampilkan, Anda dapat memilih Muat Lebih Banyak untuk menampilkan batch berikutnya.

1. Pada bilah menu, pilih AWS Cloud9, Preferensi.
2. Di jendela Preferensi, perluas Pengaturan Proyek, dan buka bagian EKSTENSI untuk memilih Konfigurasi AWS .

3. Di panel Konfigurasi AWS , buka pengaturan AWS > S3: Item Maks Per Halaman.
4. Sebelum memilih untuk memuat lebih banyak, ubah nilai default ke jumlah item S3 yang ingin Anda tampilkan.

> **Note**
>
> Rentang nilai yang diterima adalah antara 3 dan 1000. Pengaturan ini hanya berlaku untuk jumlah objek atau folder yang ditampilkan pada satu waktu. Semua ember yang Anda buat ditampilkan sekaligus. Secara default, Anda dapat membuat hingga 100 bucket di setiap Akun AWS Anda.

## Cara menggunakan objek Amazon S3

Objek adalah entitas dasar yang disimpan di Amazon S3. Objek terdiri atas data objek dan metadata.

Topik

- Mengunggah file ke bucket Amazon S3
- Mengunduh objek Amazon S3
- Menghapus objek Amazon S3
- Membuat URL yang telah ditetapkan sebelumnya untuk objek Amazon S3

### Mengunggah file ke bucket Amazon S3

Anda dapat menggunakan antarmuka Toolkit atau perintah untuk mengunggah file ke bucket

Kedua metode memungkinkan Anda untuk mengunggah file dari lingkungan pengguna dan menyimpannya sebagai objek S3 di AWS Cloud. Anda dapat mengunggah file ke bucket atau ke folder yang mengatur isi bucket tersebut.

Mengunggah file ke bucket S3 menggunakan antarmuka

1. Di AWS Explorer, pilih simpul S3 untuk melihat daftar bucket.
2. Buka menu konteks (klik kanan) untuk bucket atau folder di bucket tersebut, lalu pilih Unggah File.

> **Catatan**
>
> Jika Anda membuka menu konteks (klik kanan) objek S3, Anda dapat memilih Unggah ke Induk. Hal ini memungkinkan Anda untuk menambahkan file ke folder atau bucket yang berisi file yang dipilih.

3. Dengan menggunakan pengelola file lingkungan Anda, pilih file, dan kemudian pilih Unggah.

   File yang dipilih diunggah sebagai objek S3 ke bucket atau folder. Setiap entri objek menjelaskan ukuran objek yang disimpan dan berapa lama objek tersebut diunggah. Anda dapat menjeda daftar objek untuk melihat jalur, ukuran, dan waktu terakhir diubah.

Mengunggah file saat ini ke bucket S3 menggunakan perintah

1. Untuk memilih file yang akan diunggah, pilih tab file.
2. Tekan Ctrl+P untuk menampilkan panel Perintah.
3. Untuk Pergi Ke Apa Pun, mulailah memasukkan frasa `upload file` untuk menampilkan perintah `AWS: Upload File`. Pilih perintah saat muncul.
4. Untuk Langkah 1: Pilih file yang akan diunggah, Anda dapat memilih file yang telah Anda pilih atau menelusuri file lain.
5. Untuk Langkah 2: Pilih bucket S3 untuk tujuan unggahan, pilih bucket dari daftar.

   File yang dipilih diunggah sebagai objek S3 ke bucket atau folder. Setiap entri objek menjelaskan ukuran objek yang disimpan dan berapa lama objek tersebut diunggah. Anda dapat menjeda daftar objek untuk melihat jalur, ukuran, dan waktu terakhir diubah.

## Mengunduh objek Amazon S3

Anda dapat mengunduh objek di bucket Amazon S3 dari AWS Cloud ke folder di lingkungan Anda AWS Cloud9 .

1. Di AWS Explorer, pilih simpul S3 untuk melihat daftar bucket.
2. Di bucket atau folder dalam bucket, buka menu konteks (klik kanan) untuk objek, lalu pilih Unduh Sebagai.
3. Dengan menggunakan pengelola file lingkungan Anda, pilih folder tujuan, masukkan nama file, lalu pilih Unduh.

Setelah file diunduh, Anda dapat membukanya di AWS Cloud9.

### Menghapus objek Amazon S3

Anda dapat menghapus objek secara permanen jika objek tersebut berada dalam bucket yang tidak berversi. Namun untuk bucket yang mendukung versioning, permintaan penghapusan tidak menghapus objek tersebut secara permanen. Sebaliknya, Amazon S3 menyisipkan delete marker dalam bucket. Untuk informasi selengkapnya, lihat Menghapus versi objek di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

1. Di AWS Explorer, pilih simpul S3 untuk melihat daftar bucket.
2. Di bucket atau folder dalam bucket, buka menu konteks (klik kanan) untuk objek, lalu pilih Hapus.
3. Pilih Hapus untuk mengonfirmasi penghapusan.

### Membuat URL yang telah ditetapkan sebelumnya untuk objek Amazon S3

Dengan URL yang telah ditetapkan sebelumnya, pemilik objek dapat berbagi objek Amazon S3 pribadi dengan orang lain dengan memberikan izin terbatas waktu untuk mengunduh objek. Untuk informasi selengkapnya, lihat Berbagi objek dengan URL yang telah ditetapkan sebelumnya di Panduan Pengguna Amazon S3.

1. Di AWS Explorer, pilih simpul S3 untuk melihat daftar bucket.
2. Di bucket atau folder dalam bucket, klik kanan objek, lalu pilih Hasilkan URL Presigned.
3. Di panel perintah AWS Toolkit, masukkan jumlah menit yang URL dapat digunakan untuk mengakses objek. Tekan Enter untuk mengonfirmasi.

   Status di bagian bawah IDE mengonfirmasi bahwa URL yang telah ditetapkan sebelumnya untuk objek telah disalin ke clipboard Anda.

## Bekerja dengan AWS SAM menggunakan AWS Toolkit

AWS Toolkit menyediakan dukungan untuk aplikasi tanpa server. Menggunakan AWS Toolkit, Anda dapat membuat aplikasi tanpa server yang berisi AWS Lambdafungsi, dan kemudian menyebarkan aplikasi ke tumpukan. AWS CloudFormation

# Membuat aplikasi nirserver

Contoh ini menunjukkan cara menggunakan AWS Toolkit untuk membuat aplikasi tanpa server. Untuk informasi tentang cara menjalankan dan men-debug aplikasi tanpa server, lihat. Menjalankan dan men-debug aplikasi nirserver

Prasyarat yang diperlukan untuk membuat aplikasi tanpa server termasuk CLI dan CLI AWS SAM .AWS  Ini termasuk dengan AWS Cloud9. Jika AWS SAM CLI tidak diinstal, atau jika sudah usang, Anda mungkin perlu menjalankan instalasi atau upgrade. Untuk petunjuk tentang cara menginstal AWS SAM CLI, lihat Menginstal CLI dan untuk petunjuk tentang cara memutakhirkan AWS SAM CLIAWS SAM , lihat Memutakhirkan CLI. AWS SAM

## Membuat aplikasi nirserver dengan AWS Toolkit

Contoh ini menunjukkan cara membuat aplikasi tanpa server dengan AWS Toolkit dengan menggunakan ().AWS Serverless Application ModelAWS SAM

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk simpul Lambda, lalu pilih Buat Aplikasi SAM Lambda.

   > **Note**
   >
   > Atau, Anda dapat memilih ikon menu di seberang judul AWS: Explorer, dan memilih Create Lambda SAM Application.

2. Pilih runtime untuk aplikasi SAM Anda. Untuk contoh ini, pilih nodejs12.x.

   > **Note**
   >
   > Jika Anda memilih salah satu runtime dengan “(Gambar),” aplikasi Anda adalah jenis `Image` paket. Jika Anda memilih salah satu runtime tanpa “(Gambar),” aplikasi Anda adalah `Zip` jenisnya. Untuk informasi selengkapnya tentang perbedaan antara jenis paket `Image` dan `Zip`, lihat paket deployment Lambda dalam Panduan Developer AWS Lambda .

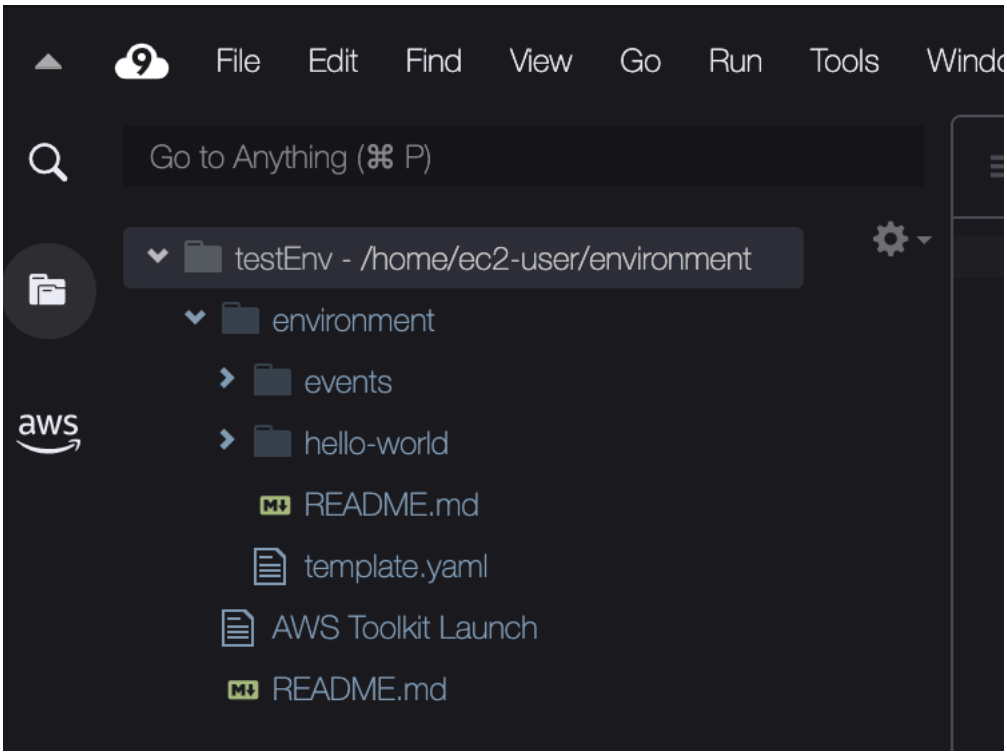
3. Pilih salah satu templat berikut untuk aplikasi tanpa server Anda:

   - AWS SAM Hello World: Template dasar dengan fungsi Lambda yang mengembalikan pesan klasik “Hello World”.

- AWS Step Functions Sample App: Contoh aplikasi yang menjalankan alur kerja perdagangan saham. Fungsi langkah mengatur interaksi fungsi Lambda yang terlibat.

4. Pilih lokasi untuk proyek baru Anda. Jika tersedia, Anda dapat memilih folder ruang kerja yang ada. Jika tidak, telusuri folder yang berbeda. Jika Anda memilih Pilih folder yang berbeda, kotak dialog akan ditampilkan di mana Anda dapat memilih lokasi folder.
5. Masukkan nama untuk aplikasi baru Anda. Untuk contoh ini, gunakan `my-sam-app-nodejs`. Setelah Anda menekan Enter, AWS Toolkit membutuhkan beberapa saat untuk membuat proyek.

Ketika proyek dibuat, Anda dapat melihat file aplikasi Anda di jendela Lingkungan. Temukan itu tercantum di jendela Explorer.




## Menjalankan dan men-debug aplikasi nirserver

Anda dapat menggunakan AWS Toolkit untuk mengonfigurasi cara men-debug aplikasi tanpa server dan menjalankannya secara lokal di lingkungan pengembangan Anda. Anda dapat men-debug aplikasi tanpa server yang ditentukan oleh template AWS Serverless Application Model ()AWS SAM. Template ini menggunakan sintaks YAMB sederhana untuk mendeskripsikan sumber daya seperti fungsi, database APIs, dan pemetaan sumber peristiwa yang membentuk aplikasi tanpa server.

Untuk melihat lebih dekat pada AWS SAM template, lihat anatomi AWS SAM template di Panduan AWS Serverless Application Model Pengembang.

Atau, Anda dapat dengan cepat men-debug aplikasi nirserver yang belum dikomit ke templat SAM.

Anda mulai mengonfigurasi perilaku debug dengan menggunakan tindakan sebaris untuk mengidentifikasi fungsi yang memenuhi syarat AWS Lambda . Untuk menggunakan infrastruktur yang ditentukan oleh templat SAM, gunakan tindakan sebaris dalam file berformat YAML yang relevan. Untuk menguji fungsi secara langsung tanpa templat, gunakan tautan sadar konteks untuk penangan Lambda di file aplikasi.

> **Note**
>
> Dalam contoh ini, kita men-debug aplikasi yang menggunakan JavaScript. Tetapi Anda dapat menggunakan fitur debugging yang tersedia di AWS Toolkit dengan bahasa dan runtime berikut:
>
> - JavaScript - Node.js 10. x, 12. x, 14. x
> - Python — 3.7, 3.8, 3.9, 3.10 (Python 2.7 dan 3.6 aplikasi tanpa server dapat dijalankan tetapi tidak di-debug oleh Toolkit.) AWS
>
> Pilihan bahasa Anda juga memengaruhi bagaimana tautan sadar konteks menunjukkan penangan Lambda yang memenuhi syarat. Untuk informasi selengkapnya, lihat Menjalankan dan men-debug fungsi nirserver langsung dari kode.

## Menggunakan templat SAM untuk menjalankan dan men-debug aplikasi nirserver

Untuk aplikasi yang dijalankan dan di-debug menggunakan templat SAM, file berformat YAML menjelaskan perilaku aplikasi dan sumber daya yang digunakannya. Jika Anda membuat aplikasi tanpa server menggunakan AWS Toolkit, file bernama akan dibuat `template.yaml` secara otomatis untuk proyek Anda.

Dalam prosedur ini, gunakan contoh aplikasi yang dibuat diMembuat aplikasi nirserver.

Untuk menggunakan templat SAM guna menjalankan dan men-debug aplikasi nirserver

1. Untuk melihat file aplikasi Anda yang membentuk aplikasi nirserver Anda, buka jendela Lingkungan.

2. Dari folder aplikasi (misalnya, my-sample-app), buka `template.yaml` file.
3. Untuk`template.yaml`, pilih Edit Konfigurasi Peluncuran.

   Editor baru menampilkan file `launch.json` yang menyediakan konfigurasi debug dengan atribut default.

4. Edit atau konfirmasi nilai untuk properti konfigurasi berikut:

   - `"name"`— Masukkan nama yang ramah pembaca untuk muncul di bidang tarik-turun Konfigurasi di tampilan Jalankan.
   - `"target"`— Pastikan nilainya`"template"`. Dengan begitu, template SAM adalah titik masuk untuk sesi debug.
   - `"templatePath"` – Masukkan jalur relatif atau absolut untuk file `template.yaml`.
   - `"logicalId"`— Pastikan bahwa nama cocok dengan yang ditentukan di bagian Sumber daya dari template SAM. Dalam hal ini, ini adalah `HelloWorldFunction` dari tipe `AWS::Serverless::Function`.

   Untuk informasi selengkapnya tentang entri ini dan entri lainnya dalam file `launch.json`, lihat Opsi konfigurasi untuk men-debug aplikasi nirserver.

5. Jika Anda puas dengan konfigurasi debug Anda, simpan `launch.json`. Kemudian, pilih tombol “putar” hijau di sebelah RUN untuk memulai debugging.

   > **Note**
   >
   > Jika aplikasi SAM Anda gagal dijalankan, periksa jendela Output untuk melihat apakah kesalahan tersebut disebabkan oleh citra Docker yang tidak membangun. Anda mungkin perlu mengosongkan ruang disk di lingkungan Anda.
   > Untuk informasi selengkapnya, lihat Kesalahan menjalankan aplikasi SAM secara lokal di AWS Toolkit karena AWS Cloud9 lingkungan tidak memiliki cukup ruang disk.

   Saat sesi debugging dimulai, panel DEBUG CONSOLE menampilkan output debugging dan menampilkan nilai apa pun yang dikembalikan oleh fungsi Lambda. Saat men-debug aplikasi SAM, AWS Toolkit dipilih sebagai saluran Output di panel Output.

> **Note**
>
> Untuk pengguna Windows, jika Anda melihat kesalahan pemasangan Docker selama proses ini, Anda mungkin perlu menyegarkan kredensil untuk drive bersama Anda di Pengaturan Docker. Kesalahan pemasangan Docker terlihat mirip dengan yang berikut ini.
>
> ```
> Fetching lambci/lambda:nodejs10.x Docker container image......
> 2019-07-12 13:36:58 Mounting C:\Users\<username>\AppData\Local\Temp\ ...
>  as /var/task:ro,delegated inside runtime container
> Traceback (most recent call last):
> ...requests.exceptions.HTTPError: 500 Server Error: Internal Server
>  Error ...
> ```

## Menjalankan dan men-debug fungsi nirserver langsung dari kode

Saat menguji AWS SAM aplikasi, Anda dapat memilih untuk menjalankan dan men-debug hanya fungsi Lambda. Kecualikan sumber daya lain yang ditentukan oleh template SAM. Pendekatan ini melibatkan penggunaan tindakan inline untuk mengidentifikasi penangan fungsi Lambda dalam kode sumber yang dapat langsung dipanggil.

Penangan Lambda yang dideteksi oleh tautan sadar konteks bergantung pada bahasa dan runtime yang Anda gunakan untuk aplikasi Anda.

| Bahasa/runtime | Kondisi untuk fungsi Lambda untuk diidentifikasi oleh tautan sadar konteks |
| --- | --- |
| JavaScript (Node.js 10.x, 12.x, dan 14.x) | Fungsi tersebut memiliki fitur berikut:<br>• Ini adalah fungsi yang diekspor dengan hingga tiga parameter.<br>• Ini memiliki file `package.json` dalam folder induknya di dalam folder ruang kerja. |
| Python (3.7, 3.8, 3.9, dan 3.10) | Fungsi tersebut memiliki fitur berikut:<br>• Ini adalah fungsi tingkat atas. |

| Bahasa/runtime | Kondisi untuk fungsi Lambda untuk diidentifikasi oleh tautan sadar konteks |
| --- | --- |
| | • Ini memiliki file `requirements.txt` dalam folder induknya di dalam folder ruang kerja. |

Untuk menjalankan dan men-debug aplikasi nirserver langsung dari kode aplikasi

1. Untuk melihat file aplikasi tanpa server Anda, navigasikan ke folder aplikasi dengan memilih ikon folder di sebelah editor.
2. Dari folder aplikasi (misalnya, my-sample-app), perluas folder fungsi (dalam contoh ini, hello-world) dan buka file. `app.js`
3. Dalam tindakan sebaris yang mengidentifikasi fungsi penangan Lambda yang memenuhi syarat, pilih `Add Debug Configuration`. Jika opsi tambahkan konfigurasi debug tidak muncul, Anda harus mengaktifkan lensa kode. Untuk mengaktifkan lensa kode, lihatthe section called “Mengaktifkan lensa kode AWS Toolkit”.
4. Pilih runtime tempat aplikasi SAM Anda berjalan.
5. Di editor untuk file `launch.json`, edit atau konfirmasi nilai untuk properti konfigurasi berikut:

   - `"name"` – Masukkan nama yang mudah dibaca.
   - `"target"` – Pastikan bahwa nilainya adalah `"code"` agar penangan fungsi Lambda dipanggil secara langsung.
   - `"lambdaHandler"` – Masukkan nama metode dalam kode Anda yang dipanggil Lambda untuk memanggil fungsi Anda. Misalnya, untuk aplikasi di JavaScript, defaultnya adalah`app.lambdaHandler`.
   - `"projectRoot"` – Masukkan jalur ke file aplikasi yang berisi fungsi Lambda.
   - `"runtime"`— Masukkan atau konfirmasikan runtime yang valid untuk lingkungan eksekusi Lambda (misalnya`"nodejs.12x"`,).
   - `"payload"` – Pilih salah satu opsi berikut untuk menentukan muatan peristiwa yang ingin Anda berikan ke fungsi Lambda Anda sebagai input:
     - `"json"`: Pasangan nilai kunci berformat JSON yang menentukan muatan peristiwa.
     - `"path"`: Jalur ke file yang digunakan sebagai muatan peristiwa.
6.

Jika Anda puas dengan konfigurasi debug, pilih panah play hijau di sebelah JALANKAN untuk memulai debugging.

Saat sesi debugging dimulai, panel DEBUG CONSOLE menampilkan output debugging dan menampilkan nilai apa pun yang dikembalikan oleh fungsi Lambda. Saat men-debug aplikasi SAM, AWS Toolkit dipilih sebagai saluran Output di panel Output.

> **Note**
>
> Jika Anda melihat Docker disebutkan dalam pesan kesalahan, lihat catatan ini.

## Menjalankan dan men-debug sumber daya Amazon API Gateway lokal

Anda dapat menjalankan atau men-debug sumber daya lokal AWS SAM API Gateway yang ditentukan dalam`template.yaml`. Lakukan dengan menjalankan konfigurasi AWS Cloud9 peluncuran `type=aws-sam` dengan file`invokeTarget.target=api`.

> **Note**
>
> API Gateway mendukung dua jenis APIs. Mereka adalah REST dan HTTP APIs. Namun, fitur API Gateway dengan AWS Toolkit hanya mendukung REST APIs. Terkadang HTTP APIs disebut “API Gateway V2 APIs.”

Untuk menjalankan dan men-debug sumber daya API Gateway lokal

1. Pilih salah satu pendekatan berikut untuk membuat konfigurasi peluncuran untuk sumber daya AWS SAM API Gateway:

   - Opsi 1: Kunjungi kode sumber handler (khususnya, file.js, .cs, atau .py) di AWS SAM proyek Anda, arahkan kursor ke handler Lambda, dan pilih Tambahkan Konfigurasi Debug Jika opsi tambahkan konfigurasi debug tidak muncul, aktifkan lensa kode. Untuk mengaktifkan lensa kode, lihatthe section called “Mengaktifkan lensa kode AWS Toolkit”.). Kemudian, dalam menu, pilih item bertanda API Event.
   - Opsi 2: Edit `launch.json` dan buat konfigurasi peluncuran baru menggunakan sintaks berikut.

```
{
    "type": "aws-sam",
    "request": "direct-invoke",
    "name": "myConfig",
    "invokeTarget": {
        "target": "api",
        "templatePath": "n12/template.yaml",
        "logicalId": "HelloWorldFunction"
    },
    "api": {
        "path": "/hello",
        "httpMethod": "post",
        "payload": {
            "json": {}
        }
    },
    "sam": {},
    "aws": {}
}
```

2. Di menu tarik-turun di sebelah tombol Run, pilih konfigurasi peluncuran (dinamai `myConfig` dalam contoh sebelumnya).
3. (Opsional) Tambahkan titik henti ke kode proyek Lambda Anda.
4. Pilih tombol jalankan di sebelah tombol “play” hijau.
5. Di panel output, lihat hasilnya.

### Konfigurasi

Ketika Anda menggunakan `invokeTarget.target` nilai properti `api`, Toolkit mengubah validasi dan perilaku konfigurasi peluncuran untuk mendukung bidang `api`.

```
{
    "type": "aws-sam",
    "request": "direct-invoke",
    "name": "myConfig",
    "invokeTarget": {
        "target": "api",
        "templatePath": "n12/template.yaml",
        "logicalId": "HelloWorldFunction"
```

```
    },
    "api": {
        "path": "/hello",
        "httpMethod": "post",
        "payload": {
            "json": {}
        },
        "querystring": "abc=def&qrs=tuv",
        "headers": {
            "cookie": "name=value; name2=value2; name3=value3"
        }
    },
    "sam": {},
    "aws": {}
}
```

Ganti nilai-nilai dalam contoh sebagai berikut:

invokeTarget.logicalId

Sumber daya API.

path

Jalur API yang diminta konfigurasi peluncuran (misalnya,`"path": "/hello"`).

Harus merupakan jalur API yang valid yang diselesaikan dari `template.yaml` yang ditentukan oleh`invokeTarget.templatePath`.

httpMethod

Gunakan salah satu kata kerja berikut: “delete,” “get,” “head,” “options,” “patch,” “post,” dan “put.”

payload

Payload JSON (badan HTTP) untuk mengirim permintaan dengan struktur dan aturan yang sama dengan bidang lambda.payload.

`payload.path`menunjuk ke file yang berisi payload JSON.

`payload.json` menentukan muatan JSON sebaris.

headers

Peta opsional pasangan nama-nilai. Gunakan untuk menentukan header HTTP untuk disertakan dalam permintaan.

```
"headers": {
     "accept-encoding": "deflate, gzip;q=1.0, *;q=0.5",
     "accept-language": "fr-CH, fr;q=0.9, en;q=0.8, de;q=0.7, *;q=0.5",
     "cookie": "name=value; name2=value2; name3=value3",
     "user-agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36",
}
```

querystring

(Opsional) Gunakan string ini untuk mengatur permintaan (misalnya,`"querystring": "abc=def&ghi=jkl"`). `querystring`

aws

Bagaimana informasi AWS koneksi disediakan. Untuk informasi selengkapnya, lihat tabel properti koneksi AWS (**`aws`**) di Opsi konfigurasi untuk men-debug aplikasi nirserver.

sam

Bagaimana AWS SAM CLI membangun aplikasi. Untuk informasi selengkapnya, lihat properti AWS SAM CLI (**`sam`**) di Opsi konfigurasi untuk men-debug aplikasi nirserver.

## Menyinkronkan aplikasi tanpa server

Contoh ini menunjukkan cara menyinkronkan aplikasi tanpa server yang dibuat di topik sebelumnya (Membuat aplikasi nirserver) untuk AWS menggunakan. AWS Toolkit for Visual Studio Code

### Prasyarat

- Pastikan untuk memilih nama bucket Amazon S3 yang unik secara global.
- Pastikan kredenal yang Anda konfigurasikan menyertakan akses baca/tulis yang sesuai ke layanan berikut: Amazon S3,,, AWS CloudFormation dan Amazon API Gateway. AWS Lambda

- Untuk aplikasi dengan tipe penerapan`Image`, pastikan Anda memiliki nama bucket Amazon S3 yang unik secara global dan URI repositori Amazon ECR yang akan digunakan untuk penerapan.

## Menyinkronkan aplikasi tanpa server

1. Di jendela AWS Explorer, buka menu konteks (klik kanan) untuk simpul Lambda dan pilih Sync SAM Application.
2. Pilih Wilayah AWS untuk menyebarkan ke.
3. Pilih file `template.yaml` yang akan digunakan untuk deployment.
4. Masukkan nama bucket Amazon S3 yang dapat digunakan penerapan ini. Bucket harus berada di Wilayah tempat Anda menerapkan.

   > **Warning**
   >
   > Nama bucket Amazon S3 harus unik secara global di semua nama bucket yang ada di Amazon S3. Tambahkan pengenal unik ke nama yang diberikan dalam contoh berikut atau pilih nama lain.

5. Jika aplikasi nirserver Anda menyertakan fungsi dengan tipe paket `Image`, masukkan nama repositori Amazon ECR yang dapat digunakan oleh deployment ini. Repositori harus berada di Wilayah tempat Anda men-deploy.
6. Masukkan nama untuk tumpukan yang di-deploy, baik nama tumpukan baru atau nama tumpukan yang sudah ada.
7. Verifikasi keberhasilan deployment pada tab AWS Toolkit di Konsol.

   Jika terjadi kesalahan, pesan muncul di kanan bawah.

   Jika hal ini terjadi, periksa teks di tab AWS Toolkit untuk detailnya. Berikut ini adalah contoh detail kesalahan.

```
Error with child process: Unable to upload artifact HelloWorldFunction referenced
 by CodeUri parameter of HelloWorldFunction resource.
S3 Bucket does not exist. Execute the command to create a new bucket
aws s3 mb s3://pbart-my-sam-app-bucket
An error occurred while deploying a SAM Application. Check the logs for more
 information by running the "View AWS Toolkit Logs" command from the Command
 Palette.
```

Dalam contoh ini, kesalahan terjadi karena bucket Amazon S3 tidak ada.

Ketika penerapan selesai, Anda akan melihat aplikasi Anda yang terdaftar di AWS Explorer. Untuk mempelajari cara memanggil fungsi Lambda yang dibuat sebagai bagian dari aplikasi, lihat Memanggil fungsi Lambda jarak jauh.

## Menghapus aplikasi nirserver dari AWS Cloud

Menghapus aplikasi tanpa server melibatkan penghapusan AWS CloudFormation tumpukan yang sebelumnya Anda gunakan ke Cloud. AWS Perhatikan bahwa prosedur ini tidak menghapus direktori aplikasi Anda dari host lokal Anda.

1. Buka AWS Explorer.
2. Di jendela AWS Explorer, perluas Wilayah yang berisi aplikasi yang di-deploy yang ingin Anda hapus, lalu perluas AWS CloudFormation.
3. Buka menu konteks (klik kanan) untuk nama AWS CloudFormation tumpukan yang sesuai dengan aplikasi tanpa server yang ingin Anda hapus. Kemudian, pilih Delete CloudFormation Stack.
4. Untuk mengonfirmasi bahwa Anda ingin menghapus tumpukan yang dipilih, pilih Hapus.

Jika penghapusan tumpukan berhasil, AWS Toolkit menghapus nama tumpukan dari daftar di Explorer. AWS CloudFormation AWS

## Mengaktifkan lensa kode AWS Toolkit

Langkah ini menunjukkan bagaimana Anda dapat mengaktifkan lensa kode AWS Toolkit.

1. Pada bilah menu, pilih AWS Cloud9, dan kemudian Preferensi.
2. Pada tab Preferensi, di bilah sisi, pilih AWS Toolkit.
3. Untuk mengaktifkan lensa kode, pilih Aktifkan Lensa Kode.

## Opsi konfigurasi untuk men-debug aplikasi nirserver

Dengan tindakan sebaris, Anda dapat dengan mudah menemukan dan menentukan properti untuk menjalankan fungsi Lambda secara langsung atau dengan templat SAM. Anda juga dapat menentukan properti untuk `"lambda"` (bagaimana fungsi berjalan), `"sam"` (bagaimana AWS SAM CLI membangun aplikasi), dan `"aws"` (bagaimana informasi AWS koneksi disediakan).

AWS SAM: Pemanggilan penangan Lambda langsung/Pemanggilan Lambda berbasis Template

| Properti | Deskripsi |
| --- | --- |
| `type` | Menentukan ekstensi mana yang mengelola konfigurasi peluncuran. Selalu atur `aws-sam` untuk menggunakan AWS SAM CLI untuk membangun dan men-debug secara lokal. |
| `name` | Menentukan nama yang mudah dibaca untuk muncul di daftar Konfigurasi peluncuran debug. |
| `request` | Menentukan jenis konfigurasi yang akan dilakukan oleh ekstensi yang ditunjuk (`aws-sam`). Selalu atur ke `direct-invoke` untuk memulai fungsi Lambda. |
| `invokeTarget` | Menentukan titik masuk untuk memanggil sumber daya.<br><br>Untuk memanggil fungsi Lambda secara langsung, tetapkan nilai untuk bidang `invokeTarget` berikut:<br><br>• `target` – Atur ke `code`.<br>• `lambdaHandler` – Nama dari penangan fungsi Lambda untuk dipanggil.<br>• `projectRoot` – Jalur untuk file aplikasi yang berisi penangan Lambda.<br><br>Untuk memanggil sumber daya Lambda dengan templat SAM, atur nilai untuk bidang `invokeTarget` berikut:<br><br>• `target` – Atur ke `template`.<br>• `templatePath` – Jalur ke file templat SAM.<br>• `logicalId` – Nama sumber daya dari `AWS::Lambda::Function` atau `AWS::Serverless::Function` untuk dipanggil. Anda dapat menemukan nama sumber daya di templat SAM berformat YAML. |

Properti Lambda (**`"lambda"`**)

| Properti | Deskripsi |
| --- | --- |
| `environmentVariables` | Melewati parameter operasional ke fungsi Anda. Misalnya, jika Anda menulis ke bucket Amazon S3, konfigurasikan nama bucket sebagai variabel lingkungan. Jangan membuat kode keras nama bucket yang Anda tulis. |
| `payload` | Menyediakan dua opsi untuk muatan peristiwa yang Anda berikan ke fungsi Lambda Anda sebagai input.<br>• `"json"`: Pasangan nilai kunci berformat JSON yang menentukan muatan peristiwa.<br>• `"path"`: Jalur ke file yang digunakan sebagai muatan peristiwa. |
| `memoryMB` | Menentukan megabyte memori yang disediakan untuk menjalankan sebuah fungsi Lambda yang dipanggil. |
| `runtime` | Menentukan runtime yang digunakan oleh fungsi Lambda. Untuk informasi selengkapnya, lihat runtime AWS Lambda. |
| `timeoutSec` | Menetapkan waktu yang diizinkan, dalam detik, sebelum waktu sesi debug habis. |

Ekstensi AWS Toolkit menggunakan AWS SAM CLI untuk membangun dan men-debug aplikasi tanpa server secara lokal. Anda dapat mengonfigurasi perilaku perintah AWS SAM CLI menggunakan properti `"sam"` konfigurasi dalam file. `launch.json`

AWS SAM Properti CLI () **`"sam"`**

| Properti | Deskripsi | Nilai default |
| --- | --- | --- |
| `buildArguments` | Mengonfigurasi cara perintah `sam build` membangun kode sumber Lambda Anda. Untuk melihat opsi pembangunan, lihat membangun sam dalam | Empty string |

| Properti | Deskripsi | Nilai default |
| --- | --- | --- |
| | Panduan Developer AWS Serverless Application Model . | |
| `containerBuild` | Menunjukkan apakah akan membangun fungsi Anda di dalam wadah Docker AWS Lambda-like. | `false` |
| `dockerNetwork` | Menentukan nama atau ID dari jaringan Docker yang harus dihubungkan ke kontainer Lambda Docker, bersama dengan jaringan bridge default. Jika tidak ditentukan, kontainer Lambda hanya terhubung ke jaringan Docker bridge default. | Empty string |
| `localArguments` | Argumen pemanggilan lokal tambahan. | Empty string |
| `skipNewImageCheck` | Menentukan apakah perintah harus melewatkan menarik ke bawah citra Docker terbaru untuk runtime Lambda. | `false` |
| `template` | Menyesuaikan templat SAM Anda dengan menggunakan parameter untuk masukan nilai pelanggan ke dalamnya. Untuk informasi selengkap nya, lihat Parameter dalam Panduan Pengguna AWS CloudFormation . | `"parameters":{}` |

Properti koneksi AWS (**`"aws"`**)

| Properti | Deskripsi | Nilai default |
| --- | --- | --- |
| `credentials` | Memilih profil tertentu (misalnya,`profile:default` ) dari file kredensi Anda untuk mendapatkan AWS kredensi. | AWS Kredensial yang disediakan oleh file AWS konfigurasi bersama yang ada atau file kredensi bersama AWS . |
| `Region` | Menetapkan AWS Wilayah layanan (misalnya, us-east-1). | AWS Wilayah default yang terkait dengan profil kredensial aktif. |

# Bekerja dengan AWS Step Functions menggunakan AWS Toolkit

AWS Toolkit menyediakan dukungan untuk AWS Step Functions. Step Functions memungkinkan Anda membuat mesin status yang menentukan alur kerja untuk AWS Lambda fungsi dan AWS layanan lain yang mendukung aplikasi penting bisnis.

Anda dapat menggunakan AWS Toolkit untuk melakukan hal berikut dengan Step Functions:

- Buat dan publikasikan mesin negara, yang merupakan alur kerja yang terdiri dari langkah-langkah individual.
- Unduh file yang mendefinisikan alur kerja mesin status.
- Jalankan alur kerja mesin status dengan masukan yang telah Anda masukkan atau pilih.

Topik


- Prasyarat
- Membuat dan mempublikasikan mesin negara
- Jalankan mesin status di AWS Toolkit
- Unduh file definisi mesin negara dan visualisasikan alur kerjanya

## Prasyarat

Step Functions dapat menjalankan kode dan mengakses AWS sumber daya (seperti menjalankan fungsi Lambda). Untuk menjaga keamanan, Anda harus memberikan akses Step Functions ke sumber daya tersebut dengan menggunakan IAM role.

Dengan AWS Toolkit, Anda dapat memanfaatkan peran IAM yang dihasilkan secara otomatis yang valid untuk AWS Wilayah tempat Anda membuat mesin status. Untuk membuat peran IAM Anda sendiri untuk mesin status, lihat Cara AWS Step Functions Kerja dengan IAM di Panduan AWS Step Functions Pengembang.

## Membuat dan mempublikasikan mesin negara

Saat Anda membuat mesin status dengan AWS Toolkit, Anda memilih template pemula yang mendefinisikan alur kerja untuk kasus bisnis. Anda kemudian dapat mengedit atau mengganti template yang sesuai dengan kebutuhan spesifik Anda. Untuk informasi selengkapnya tentang mendefinisikan mesin status dalam file yang mewakili strukturnya, lihat Bahasa Negara Amazon di Panduan AWS Step Functions Pengembang.

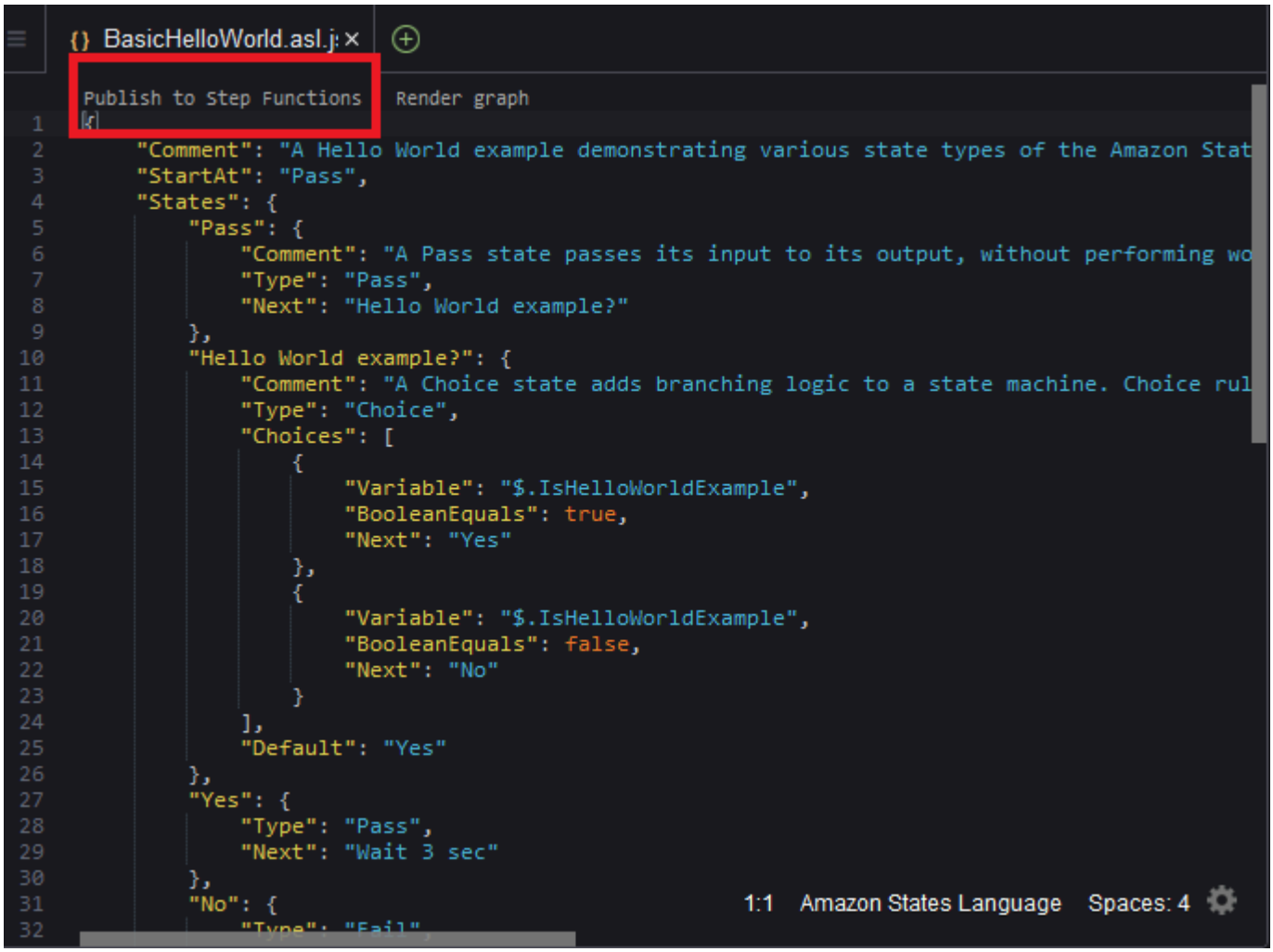
1. Di panel AWS Explorer, buka menu konteks (klik kanan) untuk Step Functions, lalu pilih Create a new Step Function state machine.
2. Di panel perintah, pilih template starter untuk alur kerja mesin negara Anda.
3. Selanjutnya, pilih format untuk file Amazon States Language (ASL) yang mendefinisikan mesin status Anda.

   Editor terbuka untuk menampilkan file ASL yang mendefinisikan alur kerja mesin negara.

   > **Note**
   >
   > Untuk informasi tentang mengedit file ASL untuk menyesuaikan alur kerja Anda, lihat Menyatakan Struktur Mesin.

4. Dalam file ASL, pilih Publish to Step Functions untuk menambahkan state machine Anda ke AWS Cloud.

> **Note**
>
> Anda juga dapat memilih Render grafik dalam file ASL untuk menampilkan representasi visual dari alur kerja mesin negara.




5. Di panel perintah, pilih AWS Region untuk meng-host fungsi langkah Anda.
6. Selanjutnya, Anda dapat memilih untuk membuat fungsi langkah baru atau memperbarui yang sudah ada.

Quick Create

Opsi ini memungkinkan Anda untuk membuat fungsi langkah baru dari file ASL menggunakan langkah- functions/latest/dg/concepts - standard-vs-express .html. Anda diminta untuk menentukan yang berikut:

- Peran IAM yang memungkinkan fungsi langkah Anda menjalankan kode dan mengakses AWS sumber daya. (Anda dapat memilih peran IAM yang dibuat secara otomatis yang valid untuk AWS Wilayah tempat Anda membuat mesin status.)
- Nama untuk fungsi baru Anda.

Anda dapat memeriksa apakah mesin status Anda berhasil dibuat dan mendapatkan ARN di tab keluaran AWS Toolkit.

Quick Update

Jika mesin status sudah ada di AWS Wilayah, Anda dapat memilih salah satu untuk diperbarui dengan file ASL saat ini.

Anda dapat memeriksa apakah mesin status Anda berhasil diperbarui dan mendapatkan ARN di tab keluaran AWS Toolkit.

Setelah Anda membuat mesin status, itu muncul di bawah Step Functions di panel AWS Explorer. Jika tidak segera muncul, pilih menu Toolkit, Refresh Explorer.

## Jalankan mesin status di AWS Toolkit

Anda dapat menggunakan AWS Toolkit untuk menjalankan mesin status jarak jauh. Mesin status yang sedang berjalan menerima teks JSON sebagai input dan meneruskan input itu ke status pertama dalam alur kerja. Status individu menerima JSON sebagai input dan biasanya lulus JSON sebagai output ke status berikutnya. Untuk informasi selengkapnya, lihat Pemrosesan Input dan Output di Step Functions.

1. Di panel AWS Explorer, pilih Step Functions. Kemudian buka menu konteks (klik kanan) untuk mesin status tertentu dan pilih Mulai Eksekusi.

2. Di panel Mulai Eksekusi, tambahkan input berformat JSON untuk alur kerja mesin status dengan memasukkan teks secara langsung di bidang di bawah ini atau mengunggah file dari perangkat lokal Anda.
3. Pilih Jalankan

   Tab keluaran AWS Toolkit menampilkan konfirmasi bahwa alur kerja telah dimulai dan ARN dari ID proses. Anda dapat menggunakan ID proses tersebut untuk memeriksa di AWS Step Functions konsol apakah alur kerja berhasil berjalan. Anda juga dapat melihat stempel waktu kapan alur kerja Anda dimulai dan berakhir.

## Unduh file definisi mesin negara dan visualisasikan alur kerjanya

Untuk mengunduh mesin status berarti Anda mengunduh file yang berisi teks JSON yang mewakili struktur mesin status itu. Anda kemudian dapat mengedit file ini untuk membuat mesin status baru atau memperbarui yang sudah ada. Untuk informasi selengkapnya, lihat Bahasa Negara Amazon di Panduan AWS Step Functions Pengembang.

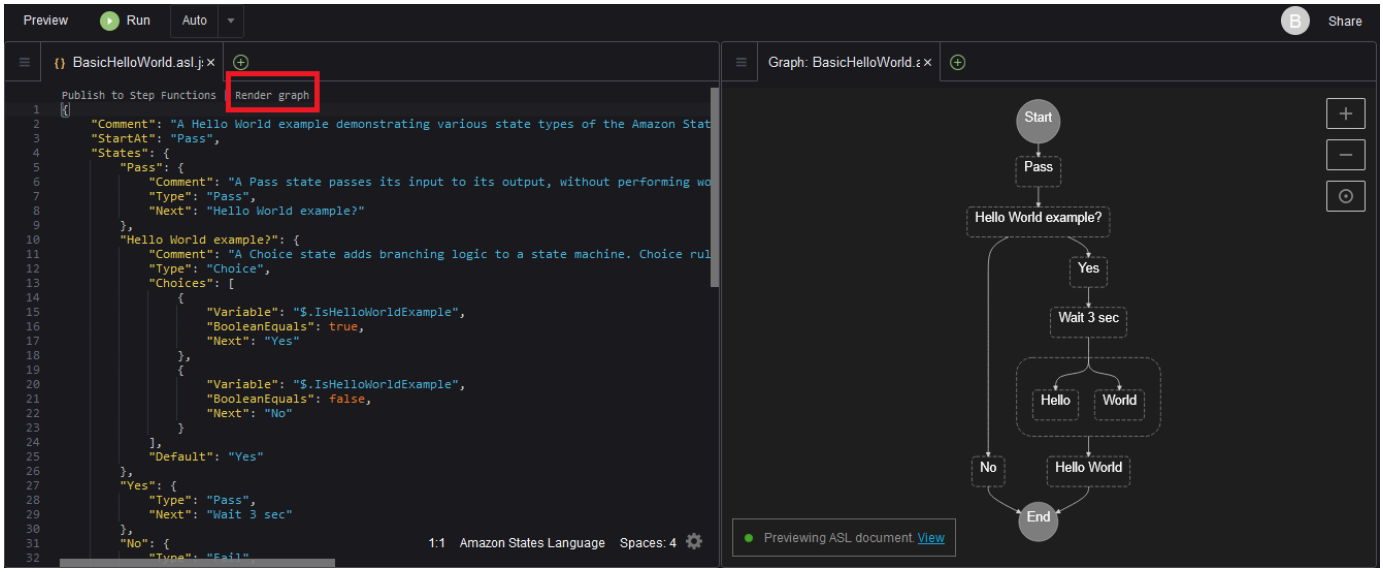
1. Di panel AWS Explorer, pilih Step Functions. Kemudian buka menu konteks (klik kanan) untuk mesin status tertentu dan pilih Download Definition.

   > **Note**
   >
   > Menu konteks juga menawarkan opsi untuk Salin Nama dan Salin ARN.

2. Dalam kotak dialog Simpan, pilih folder di lingkungan tempat Anda menyimpan file mesin status yang diunduh, lalu pilih Simpan.

   File berformat JSON yang mendefinisikan alur kerja mesin status Anda ditampilkan dalam editor.
3. Untuk menampilkan representasi visual alur kerja, pilih Render grafik.

   Sebuah jendela menampilkan diagram alur, yang menunjukkan urutan status dalam alur kerja mesin negara Anda.




# Bekerja dengan dokumen otomatisasi Systems Manager

Dengan AWS Systems Manager, Anda memiliki visibilitas dan kontrol infrastruktur Anda. AWS Systems Manager menyediakan antarmuka pengguna terpadu yang dapat Anda gunakan untuk melihat data operasional dari beberapa Layanan AWS dan mengotomatiskan tugas operasional di seluruh sumber daya Anda AWS .

Dokumen Systems Manager mendefinisikan tindakan yang dilakukan Systems Manager pada instans terkelola Anda. Dokumen otomatisasi adalah jenis dokumen Systems Manager yang digunakan untuk melakukan tugas pemeliharaan dan penyebaran umum. Ini termasuk membuat atau memperbarui Amazon Machine Image (AMI). Topik ini menguraikan cara membuat, mengedit, menerbitkan, dan menghapus dokumen otomatisasi dengan AWS Toolkit.

Topik


- Asumsi dan prasyarat
- Izin IAM untuk dokumen Otomasi Systems Manager
- Membuat dokumen otomatisasi Systems Manager baru
- Menerbitkan dokumen otomatisasi Systems Manager
- Mengedit dokumen otomatisasi Systems Manager yang ada
- Bekerja dengan versi
- Menghapus dokumen otomatisasi Systems Manager
- Menjalankan dokumen otomatisasi Systems Manager

- Memecahkan masalah dokumen otomatisasi Systems Manager di Toolkit AWS

## Asumsi dan prasyarat

Sebelum Anda mulai, pastikan Anda memenuhi persyaratan berikut:

- Anda sudah familiar dengan Systems Manager. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS Systems Manager .
- Anda sudah familiar dengan kasus penggunaan otomatisasi Systems Manager. Untuk informasi selengkapnya, lihat Otomasi AWS Systems Manager di Panduan AWS Systems Manager Pengguna.

## Izin IAM untuk dokumen Otomasi Systems Manager

Untuk membuat, mengedit, menerbitkan, dan menghapus dokumen otomatisasi Systems Manager, Anda harus memiliki profil kredensial yang berisi izin AWS Identity and Access Management (IAM) yang diperlukan. Dokumen kebijakan berikut mendefinisikan izin IAM yang diperlukan yang dapat digunakan dalam kebijakan utama.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions",
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion",
                "ssm:DeleteDocument"
            ],
            "Resource": "*"
        }
    ]
}
```

Untuk informasi tentang cara memperbarui kebijakan IAM, lihat Membuat kebijakan IAM di Panduan Pengguna IAM.

## Membuat dokumen otomatisasi Systems Manager baru

Anda dapat membuat dokumen otomatisasi di `JSON` atau `YAML` menggunakan AWS Toolkit. Saat Anda membuat dokumen otomatisasi, dokumen tersebut disajikan dalam file tanpa judul. Anda dapat memberi nama file Anda dan menyimpannya. Namun, file tersebut tidak diunggah AWS sampai Anda mempublikasikannya.

Untuk membuat dokumen otomatisasi baru

1. Pilih ikon pencarian di panel navigasi kiri atau tekan Ctrl+P untuk membuka panel Pencarian.
2. Di panel Pencarian, mulailah memasukkan istilah “manajer sistem” dan pilih perintah AWS: Buat Dokumen Manajer Sistem baru Secara Lokal saat ditampilkan.
3. Pilih salah satu template pemula untuk contoh “Hello World”.
4. Pilih salah satu `JSON` atau `YAML` sebagai format untuk dokumen Anda.

   Editor menampilkan dokumen otomatisasi baru Anda.

> **Note**
>
> Saat pertama kali membuat dokumen otomatisasi lokal, dokumen tersebut tidak muncul secara otomatis AWS. Sebelum Anda dapat menjalankannya, Anda harus mempublikasikannya ke AWS.

## Menerbitkan dokumen otomatisasi Systems Manager

Setelah Anda membuat atau mengedit dokumen otomatisasi Anda di AWS Toolkit, Anda dapat mempublikasikannya ke AWS.

Untuk mempublikasikan dokumen otomatisasi Anda

1. Buka dokumen otomatisasi yang ingin Anda publikasikan menggunakan prosedur yang diuraikan. Mengedit dokumen otomatisasi Systems Manager yang ada
2. Pilih ikon pencarian di panel navigasi kiri atau tekan Ctrl+P untuk membuka panel Pencarian.

3. Di panel Pencarian, mulailah memasukkan istilah “manajer sistem” dan pilih perintah AWS: Publikasikan Dokumen Systems Manager baru saat ditampilkan.
4. Untuk Langkah 1 dari 3, pilih Wilayah AWS tempat Anda ingin mempublikasikan dokumen.
5. Untuk Langkah 2 dari 3, pilih Buat Cepat untuk membuat dokumen otomatisasi. Atau, pilih Pembaruan Cepat untuk memperbarui dokumen otomatisasi yang ada di Wilayah tersebut.

   > **Note**
   >
   > Anda hanya dapat memperbarui dokumen otomatisasi yang Anda miliki. Jika Anda memilih Pembaruan Cepat dan Anda tidak memiliki dokumen apa pun di Wilayah itu, pesan akan memberi tahu Anda untuk menerbitkan dokumen sebelum memperbaruinya.

6. Untuk Langkah 3 dari 3, tergantung pada pilihan Anda di langkah sebelumnya, masukkan nama dokumen otomatisasi baru atau pilih dokumen yang ada untuk diperbarui.

   > **Note**
   >
   > Saat Anda mempublikasikan pembaruan ke dokumen otomatisasi yang ada di AWS, versi baru ditambahkan ke dokumen. Jika dokumen memiliki beberapa versi, Anda dapat mengatur yang default.

## Mengedit dokumen otomatisasi Systems Manager yang ada

Anda menggunakan AWS Explorer untuk menemukan dokumen otomatisasi Systems Manager yang ada. Ketika Anda membuka dokumen yang ada, itu muncul sebagai file tanpa judul di editor AWS Cloud9 . Ada tiga jenis dokumen otomatisasi yang Anda unduh:

- Dimiliki oleh Amazon: Dokumen SSM pra-konfigurasi yang dapat digunakan dengan menentukan parameter saat runtime.
- Dimiliki oleh saya: Dokumen yang telah saya buat dan terbitkan AWS.
- Berbagi dengan saya: Dokumen yang telah dibagikan pemilik dengan Anda, berdasarkan Akun AWS ID Anda.

Satu-satunya jenis dokumen yang dapat Anda perbarui AWS adalah yang dimiliki oleh saya. Anda juga dapat mengunduh dokumen otomatisasi yang dibagikan atau dimiliki oleh Amazon, dan mengeditnya AWS Cloud9. Namun, ketika Anda mempublikasikannya AWS, Anda harus

menggunakan membuat dokumen baru atau memperbarui dokumen yang sudah ada yang Anda miliki. Anda tidak dapat membuat versi baru dokumen yang memiliki pemilik lain atau dimiliki oleh Amazon.

Untuk informasi selengkapnya, lihat AWS Systems Manager dokumen di Panduan AWS Systems Manager Pengguna.

1. Di AWS Explorer, untuk Systems Manager, pilih kategori dokumen SSM yang ingin Anda unduh: Dimiliki oleh Amazon, Dimiliki oleh saya, atau Dibagikan dengan saya.
2. Untuk dokumen tertentu, buka menu konteks (klik kanan) dan pilih Unduh sebagai YAMAL atau Unduh sebagai JSON.

   Dokumen SSM yang diformat ditampilkan di tab editor baru.

Setelah selesai mengedit, Anda dapat menggunakan perintah AWS: Publikasikan Dokumen Systems Manager baru untuk membuat dokumen baru di AWS Cloud atau memperbarui dokumen yang sudah ada yang Anda miliki.

## Bekerja dengan versi

Dokumen otomatisasi Systems Manager menggunakan versi untuk manajemen perubahan. Dengan AWS Toolkit, Anda dapat mengatur versi default dokumen, yang merupakan versi yang digunakan saat Anda menjalankan dokumen.

Untuk menyetel versi default

- Di AWS Explorer, navigasikan ke dokumen yang ingin Anda atur versi default, buka menu konteks (klik kanan) untuk dokumen, dan pilih Setel versi default.

> **Note**
>
> Jika dokumen yang dipilih hanya memiliki satu versi, Anda tidak dapat mengubah default.

## Menghapus dokumen otomatisasi Systems Manager

Anda dapat menghapus dokumen otomatisasi yang Anda miliki di AWS Toolkit. Menghapus dokumen Otomasi menghapus dokumen dan semua versi dokumen.

**Important**

- Menghapus adalah tindakan destruktif yang tidak dapat dibatalkan.
- Menghapus dokumen otomatisasi yang telah dimulai tidak menghapus AWS sumber daya yang dibuat atau dimodifikasi saat dijalankan.
- Menghapus hanya diizinkan jika Anda memiliki dokumen.

Untuk menghapus dokumen otomatisasi

1. Di panel AWS Explorer, untuk Systems Manager, perluas Owned by Me untuk mencantumkan dokumen Anda.
2. Buka menu konteks (klik kanan) untuk dokumen yang ingin Anda hapus, dan pilih Hapus dokumen.
3. Di kotak dialog peringatan yang ditampilkan, pilih Hapus untuk mengonfirmasi.

## Menjalankan dokumen otomatisasi Systems Manager

Setelah dokumen otomatisasi Anda diterbitkan ke AWS, Anda dapat menjalankannya untuk melakukan tugas atas nama Anda di Anda Akun AWS. Untuk menjalankan dokumen Otomasi Anda AWS Management Console, Anda menggunakan APIs, Systems Manager AWS CLI, atau Alat AWS untuk PowerShell. Untuk petunjuk tentang cara menjalankan dokumen otomatisasi, lihat Menjalankan otomatisasi sederhana di Panduan AWS Systems Manager Pengguna.

Atau, jika Anda ingin menggunakan salah satu AWS SDKs dengan Systems Manager APIs untuk menjalankan dokumen Otomasi Anda, lihat referensi AWS SDK.

**Important**

Menjalankan dokumen otomatisasi dapat membuat sumber daya baru AWS dan dapat menimbulkan biaya penagihan. Kami sangat menyarankan agar Anda memahami apa yang akan dibuat dokumen otomatisasi Anda di akun Anda sebelum menjalankannya.

## Memecahkan masalah dokumen otomatisasi Systems Manager di Toolkit AWS

Saya menyimpan dokumen otomatisasi saya di AWS Toolkit, tetapi saya tidak melihatnya di. AWS Management Console

Menyimpan dokumen otomatisasi di AWS Toolkit tidak mempublikasikan dokumen otomatisasi ke AWS. Untuk informasi selengkapnya tentang memublikasikan dokumen Otomasi Anda, lihatMenerbitkan dokumen otomatisasi Systems Manager.

Penerbitan dokumen otomatisasi saya gagal dengan kesalahan izin.

Pastikan profil AWS kredensil Anda memiliki izin yang diperlukan untuk mempublikasikan dokumen Otomasi. Untuk contoh kebijakan izin, lihatIzin IAM untuk dokumen Otomasi Systems Manager.

Saya menerbitkan dokumen otomatisasi saya ke AWS, tetapi saya tidak melihatnya di panel AWS Explorer.

Pastikan Anda telah menerbitkan dokumen ke AWS Wilayah yang sama yang Anda jelajahi di panel AWS Explorer.

Saya telah menghapus dokumen otomatisasi saya, tetapi saya masih ditagih untuk sumber daya yang dibuatnya.

Menghapus dokumen otomatisasi tidak menghapus sumber daya yang dibuat atau dimodifikasi. Anda dapat mengidentifikasi AWS sumber daya yang telah dibuat dari Konsol Manajemen AWS Penagihan, menjelajahi tagihan, dan memilih sumber daya yang akan dihapus dari sana.

# Bekerja dengan Amazon ECR di IDE AWS Cloud9

Amazon Elastic Container Registry (Amazon ECR) adalah layanan registri kontainer terkelola AWS yang aman dan dapat diskalakan. Beberapa fungsi layanan Amazon ECR dapat diakses dari AWS Toolkit Explorer:

- Membuat repositori.
- Membuat AWS App Runner layanan untuk repositori atau gambar yang ditandai.
- Mengakses tag gambar dan repositori URIs atau. ARNs
- Menghapus tag gambar dan repositori.

Anda juga dapat mengakses berbagai fungsi Amazon ECR melalui AWS Cloud9 konsol dengan menginstal AWS CLI dan platform lainnya.

Untuk informasi selengkapnya tentang Amazon ECR, lihat Apa itu Amazon ECR? di Panduan Pengguna Registri Amazon Elastic Container.

## Prasyarat

Berikut ini sudah diinstal sebelumnya di EC2 lingkungan AWS Cloud9 IDE untuk AWS Cloud9 Amazon. Mereka diminta untuk mengakses layanan Amazon ECR dari AWS Cloud9 IDE.

### Kredensial IAM

Peran IAM yang Anda buat dan gunakan untuk otentikasi di konsol. AWS Untuk informasi lebih lanjut tentang IAM, lihat Panduan Pengguna AWS Identity and Access Management.

### Konfigurasi bucket

Docker sudah diinstal sebelumnya di AWS Cloud9 IDE untuk lingkungan AWS Cloud9 Amazon EC2 . Untuk informasi selengkapnya tentang Docker, lihat Menginstal Docker Engine.

### AWS Konfigurasi CLI versi 2

AWS CLI versi 2 sudah diinstal sebelumnya di IDE AWS Cloud9 untuk lingkungan Amazon AWS Cloud9 . EC2 Untuk informasi selengkapnya tentang AWS CLI versi 2, lihat Menginstal, memperbarui, dan menghapus CLI AWS versi 2.

Topik

- Bekerja dengan layanan Amazon ECR di AWS Cloud9

## Bekerja dengan layanan Amazon ECR di AWS Cloud9

Anda dapat mengakses layanan Amazon Elastic Container Registry (Amazon ECR) langsung dari Explorer AWS di IDE. AWS Cloud9 Anda dapat menggunakan Amazon ECR untuk mendorong gambar program ke repositori Amazon ECR. Untuk memulai, ikuti langkah-langkah ini:

1. Buat Dockerfile yang berisi informasi yang diperlukan untuk membangun gambar.
2. Buat gambar dari Dockerfile itu dan beri tag gambar untuk diproses.

3. Buat repositori yang ada di dalam instans Amazon ECR Anda.
4. Dorong gambar yang ditandai ke repositori Anda.

Bagian-bagian


- Prasyarat
- 1. Membuat Dockerfile
- 2. Membangun gambar Anda dari Dockerfile Anda
- 3. Membuat repositori baru
- 4. Mendorong, menarik, dan menghapus gambar


## Prasyarat

Sebelum Anda dapat menggunakan fitur Amazon ECR dari AWS Toolkit untuk AWS Cloud9, pastikan Anda memenuhi prasyarat ini terlebih dahulu. Prasyarat ini sudah diinstal sebelumnya di IDE AWS Cloud9 untuk EC2 lingkungan Amazon dan diperlukan untuk AWS Cloud9 mengakses Amazon ECR.

## 1. Membuat Dockerfile

Docker menggunakan file yang disebut Dockerfile untuk menentukan gambar yang dapat didorong dan disimpan di repositori jarak jauh. Sebelum Anda dapat mengunggah gambar ke repositori ECR, buat Dockerfile dan kemudian buat gambar dari Dockerfile itu.

Membuat Dockerfile

1. Untuk menavigasi ke direktori tempat Anda ingin menyimpan Dockerfile Anda, pilih opsi Toggle Tree di bilah navigasi kiri dalam IDE Anda. AWS Cloud9
2. Buat file baru bernama Dockerfile.

> **Note**
>
> AWS Cloud9 IDE mungkin meminta Anda untuk memilih jenis file atau ekstensi file. Jika ini terjadi, pilih plaintext. AWS Cloud9 IDE memiliki ekstensi “dockerfile”. Namun, kami tidak menyarankan Anda menggunakannya. Ini karena ekstensi dapat menyebabkan konflik dengan versi tertentu dari Docker atau aplikasi terkait lainnya.

Mengedit Dockerfile Anda menggunakan IDE AWS Cloud9

Jika Dockerfile Anda memiliki ekstensi file, buka menu konteks (klik kanan) untuk file tersebut dan hapus ekstensi file. Sebuah Dockerfile dengan ekstensi dapat menyebabkan konflik dengan versi tertentu dari Docker atau aplikasi terkait lainnya.

Setelah ekstensi file dihapus dari Dockerfile Anda:

1. Buka Dockerfile kosong langsung di AWS Cloud9 IDE.
2. Salin isi contoh berikut ke Dockerfile Anda.

   Example Templat gambar Dockerfile

   ```
   FROM ubuntu:22.04

   # Install dependencies
   RUN apt-get update && \
    apt-get -y install apache2

   # Install apache and write hello world message
   RUN echo 'Hello World!' > /var/www/html/index.html

   # Configure apache
   RUN echo '. /etc/apache2/envvars' > /root/run_apache.sh && \
    echo 'mkdir -p /var/run/apache2' >> /root/run_apache.sh && \
    echo 'mkdir -p /var/lock/apache2' >> /root/run_apache.sh && \
    echo '/usr/sbin/apache2 -D FOREGROUND' >> /root/run_apache.sh && \
    chmod 755 /root/run_apache.sh

   EXPOSE 80

   CMD /root/run_apache.sh
   ```

   Ini adalah Dockerfile yang menggunakan gambar Ubuntu 22.04. Petunjuk RUN memperbarui cache paket. Instal paket perangkat lunak untuk server web, lalu tulis “Hello World!” konten ke root dokumen server web. Instruksi EXPOSE mengekspos port 80 pada wadah, dan instruksi CMD memulai server web.
3. Simpan Dockerfile Anda.

## 2. Membangun gambar Anda dari Dockerfile Anda

Dockerfile yang Anda buat berisi informasi yang diperlukan untuk membangun gambar untuk suatu program. Sebelum Anda dapat mendorong gambar itu ke instans Amazon ECR Anda, buat dulu gambarnya.

Membangun gambar dari Dockerfile Anda

1. Untuk menavigasi ke direktori yang berisi Dockerfile Anda, gunakan CLI Docker atau CLI yang terintegrasi dengan instance Docker Anda.
2. Untuk membuat gambar yang ditentukan di Dockerfile Anda, jalankan perintah build Docker dari direktori yang sama dengan Dockerfile.

   ```
   docker build -t hello-world .
   ```

3. Untuk memverifikasi bahwa gambar telah dibuat dengan benar, jalankan perintah gambar Docker.

   ```
   docker images --filter reference=hello-world
   ```

   Example

   Outputnya adalah sebagai berikut.

   ```
   REPOSITORY          TAG                 IMAGE ID            CREATED
    SIZE
   hello-world         latest              e9ffedc8c286        4 minutes ago
    241MB
   ```

4. Untuk menjalankan gambar yang baru dibangun berdasarkan Ubuntu 22.04, gunakan perintah echo.

> **Note**
>
> Langkah ini tidak diperlukan untuk membuat atau mendorong gambar Anda. Namun, Anda dapat melihat bagaimana gambar program bekerja saat dijalankan.

```
FROM ubuntu:22.04
CMD ["echo", "Hello from Docker in Cloud9"]
```

Kemudian, jalankan dan bangun dockerfile. Anda harus menjalankan perintah ini dari direktori yang sama dengan dockerfile.

```
docker build -t hello-world .
docker run --rm hello-world
```

Example

Outputnya adalah sebagai berikut.

```
Hello from Docker in Cloud9
```

Untuk informasi selengkapnya tentang perintah Docker run, lihat referensi Docker run di situs web Docker.

## 3. Membuat repositori baru

Untuk mengunggah gambar Anda ke instans Amazon ECR Anda, buat repositori baru tempat gambar tersebut dapat disimpan.

Membuat repositori Amazon ECR baru

1. Dari bilah navigasi AWS Cloud9 IDE, pilih ikon AWS Toolkit.
2. Perluas menu AWS Explorer.

3. Temukan default Wilayah AWS yang terkait dengan Anda Akun AWS. Kemudian, pilih untuk melihat daftar layanan yang melalui AWS Cloud9 IDE.
4. Buka menu konteks (klik kanan) untuk opsi ECR untuk memulai proses Create new repository. Kemudian, pilih Create Repository.
5. Untuk menyelesaikan proses, ikuti prompt.
6. Setelah proses selesai, Anda dapat mengakses repositori baru Anda dari bagian ECR pada menu Explorer. AWS

## 4. Mendorong, menarik, dan menghapus gambar

Setelah Anda membuat gambar dari Dockerfile dan membuat repositori, Anda dapat mendorong gambar Anda ke repositori Amazon ECR Anda. Selain itu, menggunakan AWS Explorer dengan Docker dan AWS CLI, Anda dapat melakukan hal berikut:

- Tarik gambar dari repositori Anda.
- Hapus gambar yang disimpan di repositori Anda.
- Hapus repositori Anda.

Mengautentikasi Docker dengan registri default Anda

Otentikasi diperlukan untuk bertukar data antara instans Amazon ECR dan Docker. Untuk mengautentikasi Docker dengan registri Anda:

1. Buka terminal di dalam AWS Cloud9 IDE Anda.
2. Gunakan get-login-passwordmetode ini untuk mengautentikasi ke registri ECR pribadi Anda dan masukkan wilayah dan Akun AWS ID Anda.

```
aws ecr get-login-password \
    --region <region> \
| docker login \
    --username AWS \
    --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

> **Important**
>
> Dalam perintah sebelumnya, ganti **`region`** dan **`AWS_account_id`** dengan informasi yang spesifik untuk Anda. Akun AWS**`region`**Nilai yang valid adalah us-east-1.

Menandai dan mendorong gambar ke repositori Anda

Setelah Anda mengautentikasi Docker dengan instance Anda AWS, dorong gambar ke repositori Anda.

1. Gunakan perintah docker images untuk melihat gambar yang Anda simpan secara lokal dan mengidentifikasi gambar yang ingin Anda tag.

   ```
   docker images
   ```

   Example

   Outputnya adalah sebagai berikut.

   ```
   REPOSITORY          TAG                 IMAGE ID            CREATED
    SIZE
   hello-world         latest              e9ffedc8c286        4 minutes ago
    241MB
   ```

2. Tandai gambar Anda dengan perintah tag Docker.

   ```
   docker tag hello-world:latest AWS_account_id.dkr.ecr.region.amazonaws.com/hello-
   world:latest
   ```

3. Dorong gambar yang ditandai ke repositori Anda dengan perintah push Docker.

   > **Important**
   >
   > Pastikan bahwa nama repositori lokal Anda sama dengan repositori AWS Amazon EC2 Anda. Dalam contoh ini, kedua repositori harus dipanggil. `hello-world` Untuk informasi

selengkapnya tentang mendorong gambar dengan docker, lihat Mendorong gambar Docker.

```
docker push AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world:latest
```

Example

Outputnya adalah sebagai berikut.

```
The push refers to a repository [AWS_account_id.dkr.ecr.region.amazonaws.com/hello-
world] (len: 1)
e9ae3c220b23: Pushed
a6785352b25c: Pushed
0998bf8fb9e9: Pushed
0a85502c06c9: Pushed
latest: digest:
 sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b size: 6774
```

Setelah gambar yang ditandai berhasil diunggah ke repositori, segarkan AWS Toolkit dengan memilih Refresh Explorer dari tab Explorer. AWS Ini kemudian terlihat di menu AWS Explorer di AWS Cloud9 IDE.

Menarik gambar dari Amazon ECR

- Anda dapat menarik gambar ke instance lokal Anda dari perintah tag Docker.

```
docker pull AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world:latest
```

Example

Outputnya adalah sebagai berikut.

```
azonaws.com/hello-world:latest
latest: Pulling from hello-world
Digest: sha256:e02c521fd65eae4ef1acb746883df48de85d55fc85a4172a09a124b11b339f5e
```

```
Status: Image is up to date for 922327013870.dkr.ecr.us-west-2.amazonaws.com/hello-
world.latest
```

Menghapus gambar dari repositori Amazon ECR Anda

Ada dua metode untuk menghapus gambar dari AWS Cloud9 IDE. Metode pertama adalah menggunakan AWS Explorer.

1. Dari AWS Explorer, perluas menu ECR.
2. Perluas repositori yang ingin Anda hapus gambarnya.
3. Buka menu konteks (klik kanan) untuk tag gambar yang terkait dengan gambar yang ingin Anda hapus.
4. Untuk menghapus semua gambar yang disimpan yang terkait dengan tag itu, pilih Hapus Tag... .

Menghapus gambar menggunakan CLI AWS

- Anda juga dapat menghapus gambar dari repositori Anda dengan perintah AWS batch-delete-imageecr.

```
aws ecr batch-delete-image \
      --repository-name hello-world \
      --image-ids imageTag=latest
```

Example

Outputnya adalah sebagai berikut.

```
{
    "failures": [],
    "imageIds": [
        {
            "imageTag": "latest",
            "imageDigest":
 "sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b"
        }
    ]
```

```
}
```

### Menghapus repositori dari instans Amazon ECR Anda

Ada dua metode untuk menghapus repositori dari IDE. AWS Cloud9 Metode pertama adalah menggunakan AWS Explorer:

1. Dari AWS Explorer, perluas menu ECR.
2. Buka menu konteks (klik kanan) untuk repositori yang ingin Anda hapus.
3. Pilih Hapus Repositori... .

### Menghapus repositori Amazon ECR dari CLI AWS

- Anda dapat menghapus repositori dengan perintah AWS ecr delete-repository.

> **Note**
>
> Anda biasanya tidak dapat menghapus repositori tanpa terlebih dahulu menghapus gambar yang terkandung di dalamnya. Namun, jika Anda menambahkan flag --force, Anda dapat menghapus repositori dan semua gambarnya dalam satu langkah.

```
  aws ecr delete-repository \
--repository-name hello-world \
--force
```

Example

Outputnya adalah sebagai berikut.

```
--repository-name hello-world --force
{
    "repository": {
        "repositoryUri": "922327013870.dkr.ecr.us-west-2.amazonaws.com/hello-
world",
```

```
        "registryId": "922327013870",
        "imageTagMutability": "MUTABLE",
        "repositoryArn": "arn:aws:ecr:us-west-2:922327013870:repository/hello-
world",
        "repositoryName": "hello-world",
        "createdAt": 1664469874.0
    }
}
```

# Bekerja AWS IoT dengan AWS Cloud9 IDE

Dengan AWS IoT in AWS Cloud9 IDE, Anda dapat berinteraksi dengan AWS IoT layanan sambil meminimalkan gangguan pada alur kerja Anda. AWS Cloud9 Panduan ini mencakup bagaimana Anda dapat mulai menggunakan fitur AWS IoT layanan yang tersedia di AWS Cloud9 IDE. Untuk informasi selengkapnya, lihat Apa itu AWS IoT? dalam Panduan Pengguna AWS IoT .

## AWS IoT prasyarat

Untuk mulai menggunakan AWS IoT AWS Cloud9 IDE, pastikan AWS Cloud9 pengaturan Akun AWS dan pengaturan Anda memenuhi semua persyaratan. Untuk informasi tentang Akun AWS persyaratan dan izin AWS pengguna khusus untuk AWS IoT layanan, lihat Memulai dengan AWS IoT Inti di Panduan AWS IoT Pengembang.

## AWS IoT Hal-hal

AWS IoT menghubungkan perangkat ke Layanan AWS dan AWS sumber daya. Anda dapat menghubungkan perangkat Anda AWS IoT dengan menggunakan objek yang disebut benda. Sesuatu adalah representasi dari perangkat tertentu atau entitas logis. Ini bisa berupa perangkat fisik atau sensor (misalnya, bola lampu atau sakelar di dinding). Untuk informasi selengkapnya tentang AWS IoT berbagai hal, lihat Mengelola perangkat dengan AWS IoT di Panduan AWS IoT Pengembang.

### Mengelola AWS IoT hal-hal

AWS Cloud9 IDE memiliki beberapa fitur yang membuat manajemen barang Anda efisien. Untuk mengelola AWS IoT barang-barang Anda, ikuti langkah-langkah ini:

- Create a thing

- Attach a certificate to a thing
- Detach a certificate from a thing
- Delete a thing

Untuk menciptakan sesuatu

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Buka menu konteks (klik kanan) untuk hal itu dan pilih Create Thing.
3. Masukkan nama untuk benda itu di bidang Thing Name dan ikuti prompt.
4. Ketika langkah ini selesai, ikon benda diikuti dengan nama yang Anda tentukan terlihat di bagian Thing.

Untuk melampirkan sertifikat pada suatu hal

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di bawah subbagian Things, temukan hal di mana Anda melampirkan sertifikat.
3. Buka menu konteks (klik kanan) untuk hal itu dan pilih Lampirkan Sertifikat dari menu konteks, untuk membuka pemilih input dengan daftar sertifikat Anda.
4. Dari daftar, pilih ID sertifikat yang sesuai dengan sertifikat yang ingin Anda lampirkan ke barang Anda.
5. Setelah langkah ini selesai, sertifikat Anda dapat diakses di AWS Explorer sebagai item dari benda yang Anda lampirkan.

Untuk melepaskan sertifikat dari suatu hal

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di subbagian Things, temukan hal yang ingin Anda lepaskan sertifikatnya.
3. Buka menu konteks (klik kanan) untuk hal itu dan pilih Lampirkan Sertifikat.
4. Setelah langkah ini selesai, sertifikat terpisah tidak lagi ditampilkan di bawah benda di AWS Explorer. Namun, itu masih dapat diakses dari sub-bagian Sertifikat.

Untuk menghapus sesuatu

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di subbagian Things, cari hal yang ingin Anda hapus.
3. Buka menu konteks (klik kanan) untuk hal itu dan pilih Delete Thing.
4. Setelah langkah ini selesai, hal yang dihapus tidak lagi tersedia dari subbagian Things.

> **Note**
>
> Anda hanya dapat menghapus sesuatu yang tidak memiliki sertifikat yang dilampirkan padanya.

## AWS IoT sertifikat

Sertifikat adalah cara umum untuk membuat koneksi aman antara AWS IoT layanan dan perangkat Anda. Sertifikat X.509 adalah sertifikat digital yang menggunakan standar infrastruktur kunci publik X.509 untuk mengaitkan kunci publik dengan identitas yang terkandung dalam sertifikat. Untuk informasi selengkapnya tentang AWS IoT sertifikat, lihat Otentikasi (IoT) di Panduan AWS IoT Pengembang.

### Mengelola sertifikat

AWS Toolkit ini menawarkan berbagai cara bagi Anda untuk mengelola AWS IoT sertifikat Anda langsung dari AWS Explorer. Mereka diuraikan dalam langkah-langkah berikut:

- Create a certificate
- Change a certificate status
- Attach a policy to a certificate
- Delete a certificate

Untuk membuat AWS IoT sertifikat

Sertifikat X.509 digunakan untuk terhubung dengan instance Anda. AWS IoT

1. Dari AWS Explorer, perluas bagian layanan IoT, dan buka (klik kanan) Sertifikat.
2. Untuk membuka kotak dialog, pilih Buat Sertifikat dari menu konteks.

3. Untuk menyimpan key pair RSA dan sertifikat X.509, pilih direktori di sistem file lokal Anda.

> **Note**
>
> - Nama file default berisi ID sertifikat sebagai awalan.
> - Hanya sertifikat X.509 yang disimpan bersama Anda Akun AWS, melalui layanan. AWS IoT
> - Key pair RSA Anda hanya dapat diterbitkan sekali, menyimpannya ke lokasi yang aman di sistem file Anda saat Anda diminta.
> - Jika sertifikat atau key pair tidak dapat disimpan ke sistem file Anda, maka AWS Toolkit menghapus sertifikat dari file Anda. Akun AWS

Untuk mengubah status sertifikat

Status sertifikat individu ditampilkan di sebelah ID sertifikat di AWS Explorer dan dapat diatur ke aktif, tidak aktif, atau dicabut.

> **Note**
>
> - Sertifikat Anda memerlukan status aktif sebelum Anda dapat menggunakannya untuk menghubungkan perangkat ke AWS IoT layanan Anda.
> - Sertifikat yang tidak aktif dapat diaktifkan, apakah itu dinonaktifkan sebelumnya atau tidak aktif secara default.
> - Sertifikat yang telah dicabut tidak dapat diaktifkan kembali.

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di sub-bagian Sertifikat, cari sertifikat yang ingin Anda ubah.
3. Buka menu konteks (klik kanan) untuk sertifikat yang menampilkan opsi perubahan status yang tersedia untuk sertifikat tersebut.

- Jika sertifikat memiliki status tidak aktif, pilih aktifkan untuk mengubah status menjadi aktif.
- Jika sertifikat memiliki status aktif, pilih nonaktifkan untuk mengubah status menjadi tidak aktif.

- Jika sertifikat memiliki status aktif atau tidak aktif, pilih cabut untuk mengubah status yang dicabut.

> **Note**
>
> Masing-masing tindakan perubahan status ini tersedia jika Anda memilih sertifikat yang dilampirkan pada suatu benda saat ditampilkan di subbagian Things.

Untuk melampirkan kebijakan IoT ke sertifikat

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di sub-bagian Sertifikat, cari sertifikat yang ingin Anda ubah.
3. Buka menu konteks (klik kanan) untuk sertifikat dan pilih Lampirkan Kebijakan untuk membuka pemilih input dengan daftar kebijakan yang tersedia.
4. Pilih kebijakan yang ingin Anda lampirkan ke sertifikat.
5. Ketika langkah ini selesai, kebijakan yang Anda pilih akan ditambahkan ke sertifikat sebagai item sub-menu.

Untuk melepaskan kebijakan IoT dari sertifikat

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Di sub-bagian Sertifikat, cari sertifikat yang ingin Anda ubah.
3. Perluas sertifikat dan temukan kebijakan yang ingin Anda lepaskan.
4. Buka menu konteks (klik kanan) untuk kebijakan tersebut dan pilih Lepaskan dari menu konteks.
5. Ketika langkah ini selesai, kebijakan tidak lagi dapat diakses dari sertifikat Anda, itu tersedia dari subbagian Kebijakan.

Untuk menghapus sertifikat

1. Dari AWS Explorer, perluas judul layanan IoT.
2. Di sub-bagian Sertifikat, cari sertifikat yang ingin Anda hapus.
3. Buka menu konteks (klik kanan) untuk sertifikat dan pilih Hapus Sertifikat dari menu konteks.

> **Note**
>
> Anda tidak dapat menghapus sertifikat jika terlampir pada suatu benda atau memiliki status aktif. Anda dapat menghapus sertifikat yang memiliki kebijakan terlampir.

## AWS IoT kebijakan

AWS IoT Kebijakan inti didefinisikan melalui dokumen JSON. Masing-masing berisi setidaknya satu pernyataan kebijakan. Kebijakan menentukan bagaimana AWS IoT AWS, dan perangkat Anda dapat berinteraksi satu sama lain. Untuk informasi selengkapnya tentang cara membuat dokumen kebijakan, lihat Kebijakan IoT di Panduan PengembangAWS IoT .

> **Note**
>
> Kebijakan bernama diberi versi sehingga Anda dapat memutarnya kembali. Di AWS Explorer, kebijakan IoT Anda tercantum di bawah subbagian Kebijakan dalam layanan. AWS IoT Anda dapat melihat versi kebijakan dengan memperluas kebijakan. Versi default dilambangkan dengan tanda bintang (*).

### Mengelola kebijakan

AWS Cloud9 IDE menawarkan beberapa cara bagi Anda untuk mengelola kebijakan AWS IoT layanan Anda. Ini adalah cara Anda dapat mengelola atau memodifikasi kebijakan Anda langsung dari AWS Explorer di VS Code:

- Create a policy
- Upload a new policy version
- Edit a policy version
- Change the policy version defualt
- Change the policy version defualt

Untuk membuat AWS IoT kebijakan

> **Note**
>
> Anda dapat membuat kebijakan baru dari AWS Explorer. Namun, dokumen JSON yang mendefinisikan kebijakan harus sudah ada di sistem file Anda.

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Buka menu konteks (klik kanan) untuk subbagian Kebijakan dan untuk membuka kolom input Nama Kebijakan pilih Buat Kebijakan dari Dokumen.
3. Masukkan nama dan ikuti petunjuk untuk membuka dialog yang meminta Anda memilih dokumen JSON dari sistem file Anda.
4. Pilih file JSON yang berisi definisi kebijakan Anda, kebijakan tersedia di AWS explorer setelah ini selesai.

Untuk mengunggah versi AWS IoT kebijakan baru

Anda dapat membuat versi baru kebijakan dengan mengunggah dokumen JSON ke kebijakan.

> **Note**
>
> Dokumen JSON baru harus ada di sistem file Anda untuk membuat versi baru menggunakan AWS Explorer.

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Perluas subbagian Kebijakan untuk melihat AWS IoT kebijakan Anda.
3. Buka menu konteks (klik kanan) untuk kebijakan yang ingin Anda perbarui dan pilih Buat versi baru dari Dokumen.
4. Saat dialog terbuka, pilih file JSON yang berisi pembaruan definisi kebijakan Anda.

   Versi baru dapat diakses dari kebijakan Anda di AWS Explorer.

Untuk mengedit versi AWS IoT kebijakan

Anda dapat membuka dan mengedit dokumen kebijakan menggunakan AWS Cloud9. Setelah selesai mengedit dokumen, simpan ke sistem file Anda. Kemudian, unggah ke AWS IoT layanan Anda dari AWS Explorer.

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Perluas subbagian Kebijakan dan temukan kebijakan yang ingin Anda perbarui.
3. Untuk membuka Nama Kebijakan, pilih Buat Kebijakan dari Dokumen.
4. Perluas kebijakan yang ingin Anda perbarui, lalu buka menu konteks (klik kanan) untuk versi kebijakan yang ingin Anda edit.
5. Untuk membuka versi kebijakan AWS Cloud9, pilih Lihat dari menu konteks untuk membuka versi kebijakan.
6. Saat dokumen kebijakan dibuka, edit dan simpan perubahan.

> **Note**
>
> Pada titik ini, perubahan yang Anda buat pada kebijakan hanya disimpan ke sistem file lokal Anda. Untuk memperbarui versi dan melacaknya dengan AWS Explorer, ulangi langkah-langkahnyaUpload a new policy version.

Untuk memilih default versi kebijakan baru

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Perluas subbagian Kebijakan dan temukan kebijakan yang ingin Anda perbarui.
3. Perluas kebijakan yang ingin Anda perbarui, lalu buka menu konteks (klik kanan) untuk versi kebijakan yang ingin Anda tetapkan, lalu pilih Setel sebagai Default.

   Ketika ini selesai, versi default baru yang Anda pilih memiliki bintang di sebelahnya.

Untuk menghapus kebijakan

> **Note**
>
> Sebelum Anda dapat menghapus kebijakan atau versi kebijakan, pastikan bahwa kondisi berikut terpenuhi:
>
> - Anda tidak dapat menghapus kebijakan jika kebijakan tersebut dilampirkan pada sertifikat.
> - Anda tidak dapat menghapus kebijakan jika kebijakan tersebut memiliki versi non-default.
> - Anda hanya dapat menghapus versi default kebijakan jika versi default baru dipilih atau seluruh kebijakan dihapus.
> - Sebelum menghapus seluruh kebijakan, Anda harus menghapus semua versi non-default dari kebijakan yang sama.

1. Dari AWS Explorer, perluas bagian layanan IoT.
2. Perluas subbagian Kebijakan dan temukan kebijakan yang ingin Anda perbarui.
3. Perluas kebijakan yang ingin Anda perbarui, dan buka menu konteks (klik kanan) untuk versi kebijakan yang ingin dihapus, lalu pilih Hapus.
4. Ketika versi dihapus, itu tidak lagi terlihat dari AWS Explorer.
5. Jika hanya versi default kebijakan yang tersisa, buka menu konteks (klik kanan) untuk kebijakan induk dan pilih Hapus.

# Bekerja dengan Amazon Elastic Container Service

AWS Cloud9 IDE menyediakan beberapa dukungan untuk Amazon Elastic Container Service (Amazon ECS). Anda dapat menggunakan AWS Cloud9 IDE untuk mengelola sumber daya Amazon ECS. Misalnya, Anda dapat membuat definisi tugas.

Topik

- Amazon ECS Exec di AWS Toolkit untuk AWS Cloud9

# Amazon ECS Exec di AWS Toolkit untuk AWS Cloud9

Anda dapat mengeluarkan perintah tunggal dalam wadah Amazon Elastic Container Service (Amazon ECS) dengan AWS Toolkit for. AWS Cloud9 Anda dapat melakukan ini menggunakan fitur Amazon ECS Exec.

> **Important**
>
> Mengaktifkan dan Menonaktifkan Amazon ECS Exec mengubah status sumber daya ECS Anda di. Akun AWS Perubahan termasuk menghentikan dan memulai kembali layanan. Selain itu, mengubah status sumber daya saat Amazon ECS Exec diaktifkan dapat menyebabkan hasil yang tidak terduga. Untuk informasi selengkapnya tentang Amazon ECS, lihat Menggunakan Amazon ECS Exec untuk Debugging di Panduan Pengembang Amazon ECS.

## Prasyarat Amazon ECS Exec

Sebelum Anda dapat menggunakan fitur Amazon ECS Exec, ada persyaratan prasyarat tertentu yang harus Anda penuhi.

### Persyaratan Amazon ECS

Tergantung pada apakah tugas Anda di-host di Amazon EC2 atau AWS Fargate, dan Amazon ECS Exec memiliki persyaratan versi yang berbeda.

- Jika Anda menggunakan Amazon EC2, Anda harus menggunakan AMI Amazon ECS yang dioptimalkan yang dirilis setelah 20 Januari 2021, dengan versi agen 1.50.2 atau yang lebih baru. Untuk informasi selengkapnya, lihat Amazon ECS yang dioptimalkan AMIs di Panduan Pengembang Amazon ECS.
- Jika Anda menggunakan AWS Fargate, Anda harus menggunakan platform versi 1.4.0 atau yang lebih baru. Untuk informasi selengkapnya, lihat versi AWS Fargate platform di Panduan Pengembang Amazon ECS.

### AWS konfigurasi akun dan izin IAM

Untuk menggunakan fitur Amazon ECS Exec, Anda harus memiliki kluster Amazon ECS yang ada yang terkait dengan Anda. Akun AWS Amazon ECS Exec menggunakan Systems Manager untuk

membuat koneksi dengan container di cluster Anda. Izin Peran IAM Tugas ECSrequires khusus Amazon untuk berkomunikasi dengan layanan SSM.

Untuk informasi tentang peran dan kebijakan IAM yang khusus untuk Amazon ECS Exec, lihat izin IAM yang diperlukan untuk ECS Exec di Panduan Pengembang Amazon ECS.

## Bekerja dengan Amazon ECS Exec

Anda dapat mengaktifkan atau menonaktifkan Amazon ECS Exec langsung dari AWS Explorer di AWS Toolkit untuk. AWS Cloud9 Saat Anda mengaktifkan Amazon ECS Exec, pilih container dari menu Amazon ECS, dan jalankan perintah terhadapnya.

### Mengaktifkan Amazon ECS Exec

1. Dari AWS Explorer, cari dan perluas menu Amazon ECS.
2. Perluas cluster dengan layanan yang ingin Anda modifikasi.
3. Buka menu konteks untuk (klik kanan) layanan dan pilih Aktifkan Eksekusi Perintah.

> **Important**
>
> Langkah ini memulai penyebaran baru layanan Anda dan mungkin memakan waktu beberapa menit. Untuk informasi lebih lanjut, lihat catatan di awal bagian ini.

### Menonaktifkan Amazon ECS Exec

1. Dari AWS Explorer, cari dan perluas menu Amazon ECS.
2. Perluas cluster yang berisi layanan yang Anda inginkan.
3. Buka menu konteks untuk (klik kanan) layanan dan pilih Nonaktifkan Eksekusi Perintah.

> **Important**
>
> Langkah ini memulai penyebaran baru layanan Anda dan mungkin memakan waktu beberapa menit. Untuk informasi lebih lanjut, lihat catatan di awal bagian ini.

Menjalankan perintah terhadap Container

Untuk menjalankan perintah terhadap penampung menggunakan AWS Explorer, Amazon ECS Exec harus diaktifkan. Jika tidak diaktifkan, lihat Mengaktifkan Amazon ECS Exec prosedur di bagian ini.

1. Dari AWS Explorer, cari dan perluas menu Amazon ECS.
2. Perluas cluster yang layanan yang Anda inginkan.
3. Perluas layanan untuk membuat daftar kontainer terkait.
4. Buka menu konteks untuk (klik kanan) wadah dan pilih Run Command in Container.
5. Prompt terbuka dengan daftar Tugas yang sedang berjalan. Pilih Tugas ARN yang Anda inginkan.

> **Note**
>
> Jika hanya satu tugas yang berjalan, prompt tidak terbuka. Sebagai gantinya, tugas dipilih secara otomatis.

6. Saat diminta, masukkan perintah yang ingin Anda jalankan dan tekan Enter untuk melanjutkan.

# Bekerja dengan Amazon EventBridge

AWS Toolkit untuk AWS Cloud9 menyediakan dukungan untuk Amazon EventBridge. Menggunakan AWS Toolkit untuk AWS Cloud9, Anda dapat bekerja dengan aspek-aspek tertentu EventBridge, seperti skema.

Topik

- Bekerja dengan Amazon EventBridge Schemas

## Bekerja dengan Amazon EventBridge Schemas

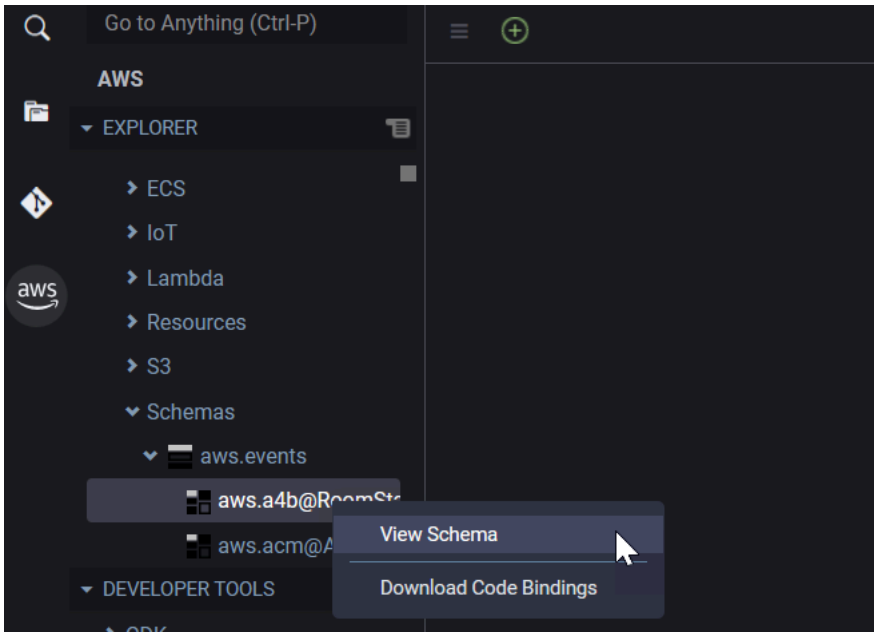
Anda dapat menggunakan AWS Toolkit AWS Cloud9 untuk melakukan berbagai operasi pada EventBridge skema Amazon.

## Prasyarat

EventBridge Skema yang ingin Anda kerjakan harus tersedia di Anda Akun AWS. Jika tidak tersedia, buat atau unggah skema. Untuk informasi selengkapnya, lihat EventBridge Skema Amazon di Panduan EventBridge Pengguna Amazon.

## Lihat Skema yang tersedia

1. Di AWS Explorer, perluas Skema.
2. Perluas nama registri yang berisi skema yang ingin Anda lihat. Misalnya, banyak skema yang AWS memasok ada di registri aws.events.
3. Untuk melihat skema di editor, buka menu konteks (klik kanan) untuk skema, lalu pilih Lihat Skema.




## Temukan Skema yang tersedia

Di AWS Explorer, lakukan satu atau beberapa hal berikut:

- Mulai masukkan judul skema yang ingin Anda temukan. AWS Explorer menyoroti judul skema yang berisi kecocokan. (Registri harus diperluas agar Anda dapat melihat judul yang disorot.)

- Buka menu konteks (klik kanan) untuk Skema, lalu pilih Skema Pencarian. Atau, perluas Skema, buka menu konteks (klik kanan) untuk registri yang berisi skema yang ingin Anda temukan,

lalu pilih Skema Pencarian di Registri. Di kotak dialog EventBridgeSchemas Search, mulailah memasukkan judul skema yang ingin Anda temukan. Kotak dialog menampilkan judul skema yang berisi kecocokan.

Untuk menampilkan skema di kotak dialog, pilih judul skema.

## Hasilkan kode untuk Skema yang tersedia

1. Di AWS Explorer, perluas Skema.
2. Perluas nama registri yang berisi Skema yang ingin Anda hasilkan kodenya.
3. Buka menu konteks (klik kanan) untuk judul Skema, lalu pilih Unduh ikatan kode.
4. Di halaman wizard yang dihasilkan, pilih yang berikut ini:

   - Versi skema
   - Bahasa pengikatan kode
   - Folder ruang kerja tempat Anda ingin menyimpan kode yang dihasilkan di mesin pengembangan lokal Anda

# Tutorial untuk AWS Cloud9

Apakah Anda baru AWS Cloud9? Ikuti tur IDE di Memulai: tutorial basic.

Bereksperimenlah dengan tutorial dan kode sampel ini untuk meningkatkan pengetahuan dan kepercayaan diri Anda menggunakan AWS Cloud9 berbagai bahasa dan AWS layanan pemrograman.

Topik


- AWS CLI dan tutorial aws-shell untuk AWS Cloud9
- AWS CodeCommit tutorial untuk AWS Cloud9
- Amazon DynamoDB tutorial untuk AWS Cloud9
- AWS CDK tutorial untuk AWS Cloud9
- Tutorial LAMP untuk AWS Cloud9
- WordPress tutorial untuk AWS Cloud9
- Tutorial Java untuk AWS Cloud9
- Tutorial C++ untuk AWS Cloud9
- Tutorial Python untuk AWS Cloud9
- Tutorial.NET untuk AWS Cloud9
- Tutorial Node.js untuk AWS Cloud9
- PHP tutorial untuk AWS Cloud9
- AWS SDK for Ruby di AWS Cloud9
- Pergi tutorial untuk AWS Cloud9
- TypeScript tutorial untuk AWS Cloud9
- Tutorial Docker untuk AWS Cloud9
- Tutorial Terkait


## AWS CLI dan tutorial aws-shell untuk AWS Cloud9

Tutorial berikut memungkinkan Anda untuk mengatur AWS Command Line Interface (AWS CLI), aws-shell, atau keduanya dalam lingkungan AWS Cloud9 pengembangan. The AWS CLI and the aws-

shell adalah alat terpadu yang menyediakan antarmuka yang konsisten untuk berinteraksi dengan semua bagian. AWS Anda dapat menggunakan AWS CLI alih-alih AWS Management Console untuk menjalankan perintah dengan cepat untuk berinteraksi AWS, dan beberapa perintah ini dapat dijalankan dengan AWS CLI atau menggunakan alternatif AWS CloudShell.

Untuk informasi selengkapnya tentang AWS CLI, lihat Panduan AWS Command Line Interface Pengguna. Untuk aws-shell, lihat sumber daya berikut:

- aws-shell di situs web GitHub
- aws-shell di situs web pip

Untuk daftar perintah yang dapat Anda jalankan dengan AWS CLI untuk berinteraksi AWS, lihat Referensi AWS CLI Perintah. Anda dapat menggunakan perintah yang sama dengan AWS CloudShell, kecuali bahwa Anda memulai perintah tanpa `aws` awalan.

Membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Instal AWS CLI, aws-shell, atau keduanya di lingkungan Anda
- Langkah 2: Mengatur pengelolaan kredensial di lingkungan Anda
- Langkah 3: Jalankan perintah dasar dengan AWS CLI atau aws-shell di lingkungan Anda
- Langkah 4: Membersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.

- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Instal AWS CLI, aws-shell, atau keduanya di lingkungan Anda

Pada langkah ini, Anda menggunakan AWS Cloud9 IDE untuk menginstal AWS CLI, aws-shell, atau keduanya di lingkungan Anda sehingga Anda dapat menjalankan perintah untuk berinteraksi. AWS

Jika Anda menggunakan lingkungan AWS Cloud9 EC2 pengembangan dan Anda hanya ingin menggunakannya AWS CLI, Anda dapat melompat ke depanLangkah 3: Jalankan perintah dasar dengan AWS CLI atau aws-shell di lingkungan Anda. Ini karena AWS CLI sudah diinstal di EC2 lingkungan, dan satu set kredensyal AWS akses sudah diatur di lingkungan. Untuk informasi selengkapnya, lihat AWS kredensi sementara yang dikelola.

Jika Anda tidak menggunakan EC2 lingkungan, lakukan hal berikut untuk menginstal AWS CLI:

1. Dengan lingkungan Anda terbuka, di IDE, periksa AWS CLI apakah sudah diinstal. Di terminal, jalankan perintah **`aws --version`** . (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika diinstal, nomor versi ditampilkan, dengan informasi seperti nomor versi Python dan nomor versi sistem operasi EC2 instans Amazon Anda atau server Anda sendiri. AWS CLI Jika AWS CLI sudah diinstal, lewati ke depanLangkah 2: Mengatur pengelolaan kredensial di lingkungan Anda.
2. Untuk menginstal AWS CLI, lihat Menginstal AWS Command Line Interface di Panduan AWS Command Line Interface Pengguna. Misalnya, untuk EC2 lingkungan yang menjalankan Amazon Linux, jalankan ketiga perintah ini, satu per satu, di terminal untuk menginstal AWS CLI.

   ```
   sudo yum -y update          # Install the latest system updates.
   sudo yum -y install aws-cli # Install the AWS CLI.
   aws --version               # Confirm the AWS CLI was installed.
   ```

   Untuk EC2 lingkungan yang menjalankan Ubuntu Server, jalankan ketiga perintah ini sebagai gantinya, satu per satu, di terminal untuk menginstal file AWS CLI.

   ```
   sudo apt update             # Install the latest system updates.
   sudo apt install -y awscli  # Install the AWS CLI.
   aws --version               # Confirm the AWS CLI was installed.
   ```

Jika Anda ingin memasang aws-shell, lakukan hal berikut:

1. Dengan lingkungan Anda terbuka, di IDE, periksa apakah aws-shell sudah terpasang. Di terminal, jalankan perintah **`aws-shell`** . (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika aws-shell terpasang, prompt `aws>` ditampilkan. Jika aws-shell terpasang, lewati ke Langkah 2: Mengatur pengelolaan kredensial di lingkungan Anda.
2. Untuk memasang aws-shell, Anda menggunakan pip. Untuk menggunakan pip, Anda harus memiliki Python yang sudah terpasang.

   Untuk memeriksa apakah Python sudah terpasang (dan untuk memasangnya jika diperlukan), ikuti instruksi di Langkah 1: Pasang Python dalam Sampel Python, lalu kembali ke topik ini.

   Untuk memeriksa apakah pip sudah terpasang, di terminal, jalankan perintah **`pip --version`** . Jika pip terpasang, nomor versi akan ditampilkan. Jika pip tidak terpasang, pasang dengan menjalankan ketiga perintah ini, satu per satu, di terminal.

   ```
   wget https://bootstrap.pypa.io/get-pip.py # Get the pip install file.
   sudo python get-pip.py                    # Install pip. (You might need to run
    'sudo python2 get-pip.py' or 'sudo python3 get-pip.py' instead, depending on how
    Python is installed.)
   rm get-pip.py                             # Delete the pip install file, as it is
    no longer needed.
   ```
3. Untuk menggunakan pip untuk memasang aws-shell, jalankan perintah berikut.

   ```
   sudo pip install aws-shell
   ```

## Langkah 2: Mengatur pengelolaan kredensial di lingkungan Anda

Setiap kali Anda menggunakan AWS CLI atau aws-shell untuk memanggil AWS layanan, Anda harus memberikan satu set kredensyal dengan panggilan tersebut. Kredensyal ini menentukan apakah AWS CLI atau aws-shell memiliki izin yang sesuai untuk melakukan panggilan itu. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Jika Anda menggunakan lingkungan AWS Cloud9 EC2 pengembangan, Anda dapat melompat ke depanLangkah 3: Jalankan perintah dasar dengan AWS CLI atau aws-shell di lingkungan Anda. Ini karena kredensyal sudah diatur di lingkungan EC2 . Untuk informasi selengkapnya, lihat AWS kredensi sementara yang dikelola.

Jika Anda tidak menggunakan EC2 lingkungan, Anda harus menyimpan kredensyal Anda secara manual di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

## Langkah 3: Jalankan perintah dasar dengan AWS CLI atau aws-shell di lingkungan Anda

Pada langkah ini, Anda menggunakan AWS CLI atau aws-shell di lingkungan Anda untuk membuat bucket di Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket.

1. Jika Anda ingin menggunakan aws-shell tetapi belum memulainya, mulai aws-shell dengan menjalankan perintah `aws-shell`. Prompt `aws>` ditampilkan.
2. Buat bucket. Jalankan **`aws s3 mb`**perintah dengan **`s3 mb`**perintah AWS CLI or dengan aws-shell, berikan nama bucket untuk dibuat. Dalam contoh ini, kami menggunakan bucket bernama`cloud9-123456789012-bucket`, di `123456789012` mana ID AWS akun Anda. Jika Anda menggunakan nama yang berbeda, gantilah sepanjang langkah ini.

```
aws s3 mb s3://cloud9-123456789012-bucket # For the AWS CLI.
s3 mb s3://cloud9-123456789012-bucket     # For the aws-shell.
```

> **Note**
>
> Nama bucket harus unik di semua AWS, bukan hanya AWS akun Anda. Nama bucket yang disarankan sebelumnya dapat membantu Anda menemukan nama bucket yang unik. Jika Anda mendapatkan pesan yang berisi kesalahan `BucketAlreadyExists`, Anda harus menjalankan perintah lagi dengan nama bucket yang berbeda.

3. Buat daftar bucket Anda yang tersedia. Jalankan **`aws s3 ls`**perintah dengan AWS CLI atau **`s3 ls`**perintah dengan aws-shell. Daftar bucket Anda yang tersedia akan ditampilkan.
4. Hapus bucket. Jalankan **`aws s3 rb`**perintah dengan AWS CLI atau **`s3 rb`**perintah dengan aws-shell, berikan nama bucket untuk dihapus.

```
aws s3 rb s3://cloud9-123456789012-bucket # For the AWS CLI.
s3 rb s3://cloud9-123456789012-bucket     # For the aws-shell.
```

Untuk mengonfirmasi apakah bucket telah dihapus, jalankan **`aws s3 ls`**perintah lagi dengan AWS CLI atau **`s3 ls`**perintah lagi dengan aws-shell. Nama bucket yang telah dihapus seharusnya tidak lagi muncul dalam daftar.

> **Note**
>
> Anda tidak perlu menghapus bucket jika Anda ingin terus menggunakannya. Untuk informasi selengkapnya, lihat Menambahkan Objek ke Bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Lihat juga perintah s3 di Referensi Perintah AWS CLI . (Ingat, jika Anda tidak menghapus bucket, hal itu dapat mengakibatkan tagihan berkelanjutan ke AWS akun Anda.)

Untuk terus bereksperimen dengan AWS CLI, lihat Bekerja dengan Amazon Web Services di Panduan AWS Command Line Interface Pengguna serta Referensi AWS CLI Perintah. Untuk melanjutkan eksperimen dengan aws-shell, lihat Referensi Perintah AWS CLI, perhatikan bahwa Anda memulai perintah tanpa prefiks `aws`.

### Langkah 4: Membersihkan

Jika Anda menggunakan aws-shell, Anda dapat berhenti menggunakannya dengan menjalankan perintah **`.exit`** atau **`.quit`** .

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk petunjuk, lihat Menghapus lingkungan di AWS Cloud9.

## AWS CodeCommit tutorial untuk AWS Cloud9

Anda dapat menggunakan AWS CodeCommit tutorial untuk mengatur lingkungan AWS Cloud9 pengembangan untuk berinteraksi dengan repositori kode jarak jauh di. CodeCommit CodeCommit adalah layanan kontrol kode sumber yang dapat Anda gunakan untuk menyimpan dan mengelola secara pribadi Git repositori di. AWS Cloud Untuk informasi selengkapnya CodeCommit, lihat Panduan AWS CodeCommit Pengguna.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya untuk Anda Akun AWS. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan CodeCommit. Untuk informasi selengkapnya, lihat EC2 Harga dan AWS CodeCommit Harga Amazon.

- Prasyarat
- Langkah 1: Siapkan grup IAM Anda dengan izin akses yang diperlukan
- Langkah 2: Buat repositori di AWS CodeCommit
- Langkah 3: Hubungkan lingkungan Anda ke repositori jarak jauh
- Langkah 4: Kloning repositori jarak jauh ke lingkungan Anda
- Langkah 5: Tambahkan file ke repositori
- Langkah 6: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Mengatur grup IAM Anda dengan izin akses yang diperlukan

Misalkan AWS kredensi Anda terkait dengan pengguna administrator di Anda Akun AWS, dan Anda ingin menggunakan pengguna itu untuk bekerja dengan CodeCommit Kemudian, lewati ke Langkah 2: Buat Repositori di. AWS CodeCommit

Anda dapat menyelesaikan langkah ini menggunakan AWS Management Console atau Antarmuka Baris Perintah AWS (AWS CLI).

### Atur grup IAM Anda dengan izin akses yang diperlukan menggunakan konsol tersebut

1. Masuk ke AWS Management Console, jika Anda belum masuk.

Untuk langkah ini, kami sarankan Anda masuk menggunakan kredensi untuk pengguna administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

2. Buka konsol IAM. Untuk melakukannya, di bilah navigasi konsol tersebut, pilih Layanan. Kemudian, pilih IAM.
3. Pilih Grup.
4. Pilih nama grup.
5. Pada tab Izin, untuk Kebijakan Terkelola, pilih Lampirkan Kebijakan.
6. Dalam daftar nama kebijakan, pilih salah satu kotak berikut ini:
   - Pilih AWSCodeCommitPowerUseruntuk akses ke semua fungsionalitas CodeCommit dan sumber daya terkait repositori. Namun, ini tidak memungkinkan Anda untuk menghapus CodeCommit repositori atau membuat atau menghapus sumber daya terkait repositori di tempat lain, Layanan AWS seperti Amazon Events. CloudWatch
   - Pilih AWSCodeCommitFullAccessuntuk kontrol penuh atas CodeCommit repositori dan sumber daya terkait di. Akun AWS Ini termasuk kemampuan untuk menghapus repositori.

   Jika Anda tidak melihat salah satu dari nama kebijakan ini dalam daftar, masukkan nama kebijakan di kotak Filter untuk menampilkannya.
7. Pilih Lampirkan Kebijakan.

Untuk melihat daftar izin akses yang diberikan oleh kebijakan AWS terkelola ini ke grup, lihat Kebijakan AWS Terkelola (Standar) AWS CodeCommit di AWS CodeCommit Panduan Pengguna.

Lewati ke Langkah 2: Buat Repositori di. AWS CodeCommit

### Siapkan grup IAM Anda dengan izin akses yang diperlukan menggunakan AWS CLI

Jalankan `attach-group-policy` perintah IAM, tentukan nama grup dan Nama Sumber Daya Amazon (ARN) dari kebijakan AWS terkelola yang menjelaskan izin akses yang diperlukan. Sintaksnya adalah sebagai berikut.

```
aws iam attach-group-policy --group-name MyGroup --policy-arn POLICY_ARN
```

Dengan perintah sebelumnya, ganti `MyGroup` dengan nama grup. Ganti `POLICY_ARN` dengan ARN dari kebijakan yang AWS dikelola:

- `arn:aws:iam::aws:policy/AWSCodeCommitPowerUser`untuk akses ke semua fungsionalitas CodeCommit dan sumber daya terkait repositori. Namun, itu tidak memungkinkan Anda untuk menghapus CodeCommit repositori atau membuat atau menghapus sumber daya terkait repositori di tempat lain, Layanan AWS seperti Amazon Events. CloudWatch
- `arn:aws:iam::aws:policy/AWSCodeCommitFullAccess`untuk kontrol penuh atas CodeCommit repositori dan sumber daya terkait di. Akun AWS Ini termasuk kemampuan untuk menghapus repositori.

Untuk melihat daftar izin akses yang diberikan oleh kebijakan AWS terkelola ini ke grup, lihat Kebijakan AWS Terkelola (Standar) AWS CodeCommit di AWS CodeCommit Panduan Pengguna.

## Langkah 2: Buat repositori di CodeCommit

Pada langkah ini, Anda membuat repositori kode jarak jauh CodeCommit dengan menggunakan konsol. CodeCommit

Jika Anda sudah memiliki CodeCommit repositori, lanjutkan ke Langkah 3: Connect Your Environment to the Remote Repository.

Anda dapat menyelesaikan langkah ini menggunakan AWS Management Console atau Antarmuka Baris Perintah AWS (AWS CLI).

### Buat repositori dalam CodeCommit menggunakan konsol

1. Misalkan Anda masuk ke AWS Management Console sebagai pengguna administrator dari langkah sebelumnya, dan Anda tidak ingin menggunakan pengguna administrator untuk membuat repositori. Kemudian, keluar dari AWS Management Console.
2. Buka CodeCommit konsol, di https://console.aws.amazon.com/codecommit.
3. Di bilah navigasi konsol, gunakan pemilih Wilayah untuk memilih tempat Wilayah AWS Anda ingin membuat repositori (misalnya, US East (Ohio)).
4. Jika halaman selamat datang ditampilkan, pilih Memulai. Jika tidak, pilih Buat repositori.
5. Pada halaman Buat repositori, untuk nama Repositori, masukkan nama untuk repositori baru Anda (misalnya,). `MyDemoCloud9Repo` Jika Anda memilih nama yang berbeda, gantilah di seluruh sampel ini.
6. (Opsional) Untuk Deskripsi, masukkan sesuatu tentang repositori. Misalnya, Anda dapat memasukkan: `This is a demonstration repository for the AWS Cloud9 sample.`

7. Pilih Buat repositori. Panel Sambungkan ke repositori Anda ditampilkan. Pilih Tutup, karena Anda akan terhubung ke repositori Anda dengan cara yang berbeda nanti dalam topik ini.

Lewati ke Langkah 3: Hubungkan Lingkungan Anda ke Repositori Jarak Jauh.

### Buat repositori dalam CodeCommit menggunakan AWS CLI

Jalankan perintah AWS CodeCommit `create-repository`. Tentukan nama repositori, deskripsi opsional, dan Wilayah AWS untuk membuat repositori di.

```
aws codecommit create-repository --repository-name MyDemoCloud9Repo --repository-
description "This is a demonstration repository for the AWS Cloud9 sample." --region
 us-east-2
```

Pada perintah sebelumnya, ganti `us-east-2` dengan ID Wilayah AWS untuk membuat repositori di. Untuk daftar Wilayah yang didukung, lihat AWS CodeCommitdi Referensi Umum Amazon Web Services.

Jika Anda memilih untuk menggunakan nama repositori yang berbeda, gantilah di seluruh sampel ini.

## Langkah 3: Hubungkan lingkungan Anda ke repositori jarak jauh

Pada langkah ini, Anda menggunakan AWS Cloud9 IDE untuk terhubung ke CodeCommit repositori yang Anda buat atau identifikasi pada langkah sebelumnya.

> **Note**
>
> Jika Anda lebih suka bekerja dengan Git melalui antarmuka visual, Anda dapat mengkloning repositori jarak jauh. Kemudian, Anda dapat menambahkan file menggunakan fitur panel Git yang tersedia di IDE.

Lengkapi salah satu dari serangkaian prosedur berikut berdasarkan jenis lingkungan AWS Cloud9 pengembangan Anda.

| Jenis lingkungan | Ikuti prosedur ini |
| --- | --- |
| EC2 lingkungan | 1. Dari sesi terminal di IDE, jalankan dua perintah berikut: |

| Jenis lingkungan | Ikuti prosedur ini |
| --- | --- |
| | ```<br>git config --global credentia<br>l.helper '!aws codecommit credentia<br>l-helper $@'<br>git config --global credentia<br>l.UseHttpPath true<br>```<br>Untuk informasi selengkapnya, lihat Langkah 2: Mengkonfigurasi AWS CLI Credential Helper pada Lingkungan AWS Cloud9 EC2 Pengembangan Anda di Integrasikan AWS Cloud9 dengan AWS CodeCommit di AWS CodeCommit Panduan Pengguna.<br>2. Loncat ke Langkah 4: Kloning Repositori Jarak Jauh ke Lingkungan Anda nanti dalam topik ini. |

| Jenis lingkungan | Ikuti prosedur ini |
| --- | --- |
| Lingkungan SSH | 1. Jika Git belum diinstal di lingkungan, gunakan sesi terminal di IDE untuk menginstalnya. Untuk informasi selengkapnya, lihat Langkah 2: Instal Gitdi Langkah Pengaturan untuk Koneksi SSH ke AWS CodeCommit Repositori di Linux, macOS, atau Unix di Panduan Pengguna.AWS CodeCommit<br>2. Selesaikan Langkah 3: Konfigurasikan Kredensional di Linux, macOS, atau Unix dalam Langkah Pengaturan untuk Koneksi SSH ke Repositori AWS CodeCommit di Linux, macOS, atau Unix di Panduan Pengguna.AWS CodeCommit<br><br>Saat Anda diinstruksikan untuk masuk ke AWS Management Console dan membuka konsol IAM, kami sarankan Anda masuk menggunakan kredensi untuk pengguna administrator di konsol Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.<br>3. Loncat ke Langkah 4: Kloning Repositori Jarak Jauh ke Lingkungan Anda nanti dalam topik ini. |

## Langkah 4: Klon repositori jarak jauh ke lingkungan Anda

Pada langkah ini, Anda menggunakan AWS Cloud9 IDE untuk mengkloning repositori jarak jauh CodeCommit ke dalam lingkungan Anda.

Untuk mengkloning repositori, jalankan perintah. **`git clone`** Ganti *`CLONE_URL`* dengan URL klon repositori.

```
git clone CLONE_URL
```

Untuk EC2 lingkungan, Anda menyediakan URL klon HTTPS yang dimulai dengan`https://`. Untuk lingkungan SSH, Anda menyediakan URL klon SSH yang dimulai dengan `ssh://`.

Untuk mendapatkan URL klon lengkap repositori, lihat Menggunakan AWS CodeCommit Konsol untuk Melihat Detail Repositori di Panduan Pengguna.AWS CodeCommit

Jika repositori Anda tidak memiliki file apa pun di dalamnya, pesan peringatan akan ditampilkan, seperti `You appear to have cloned an empty repository.` Ini diharapkan. Anda akan membahas nanti.

## Langkah 5: Tambahkan file ke repositori

Pada langkah ini, Anda membuat tiga file sederhana di repositori kloning di lingkungan AWS Cloud9 Anda. Selanjutnya, Anda menambahkan file ke Git area pementasan di repositori kloning Anda. Terakhir, Anda mengkomit file bertahap dan mendorong komit ke repositori jarak jauh Anda. CodeCommit

Jika repositori kloning sudah memiliki file di dalamnya, Anda sudah selesai dan dapat melewati sisa sampel ini.

Untuk menambahkan file ke repositori

1. Buat file baru. Pada bilah menu, pilih File, File Baru.
2. Masukkan konten berikut ke dalam file, lalu pilih File, Simpan untuk menyimpan file seperti `bird.txt` di `MyDemoCloud9Repo` direktori di AWS Cloud9 lingkungan Anda.

   ```
   bird.txt
   --------
   Birds are a group of endothermic vertebrates, characterized by feathers,
   toothless beaked jaws, the laying of hard-shelled eggs, a high metabolic
   rate, a four-chambered heart, and a lightweight but strong skeleton.
   ```

   > **Note**
   >
   > Untuk mengonfirmasi bahwa Anda menyimpan file ini di direktori yang benar, di kotak dialog Save As, pilih `MyDemoCloud9Repo` folder. Kemudian, pastikan Folder ditampilkan/`MyDemoCloud9Repo`.

3. Buat dua file lagi, bernama `insect.txt` dan`reptile.txt`, dengan konten berikut. Simpan file di `MyDemoCloud9Repo` direktori yang sama.

```
insect.txt
----------
Insects are a class of invertebrates within the arthropod phylum that
have a chitinous exoskeleton, a three-part body (head, thorax, and abdomen),
three pairs of jointed legs, compound eyes, and one pair of antennae.
```

```
reptile.txt
-----------
Reptiles are tetrapod (four-limbed vertebrate) animals in the class
Reptilia, comprising today's turtles, crocodilians, snakes,
amphisbaenians, lizards, tuatara, and their extinct relatives.
```

4. Di terminal, jalankan **`cd`**perintah untuk beralih ke `MyDemoCloud9Repo` direktori.

```
cd MyDemoCloud9Repo
```

5. Konfirmasikan bahwa file berhasil disimpan di `MyDemoCloud9Repo` direktori dengan menjalankan **`git status`**perintah. Ketiga file akan terdaftar sebagai file yang tidak terlacak.

```
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        bird.txt
        insect.txt
        reptile.txt
```

6. Tambahkan file ke area pementasan Git dengan menjalankan **`git add`**perintah.

```
git add --all
```

7. Konfirmasikan bahwa file berhasil ditambahkan ke area pementasan Git dengan menjalankan **`git status`**perintah lagi. Ketiga file sekarang terdaftar sebagai perubahan untuk dilakukan.

```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   bird.txt
        new file:   insect.txt
```

```
        new file:   reptile.txt
```

8. Komit file bertahap dengan menjalankan **`git commit`**perintah.

```
git commit -m "Added information about birds, insects, and reptiles."
```

9. Dorong komit ke repositori jarak jauh Anda CodeCommit dengan menjalankan perintah. **`git push`**

```
git push -u origin master
```

10. Konfirmasi apakah file berhasil didorong. Buka CodeCommit konsol, jika belum terbuka, di https://console.aws.amazon.com/codecommit.
11. Di bilah navigasi atas, dekat tepi kanan, pilih Wilayah AWS tempat Anda membuat repositori (misalnya, US East (Ohio)).
12. Pada halaman Dashboard, pilih MyDemoCloud9Repo. Tiga file ditampilkan.

Untuk terus bereksperimen dengan CodeCommit repositori Anda, lihat Menelusuri Isi Repositori Anda di Panduan Pengguna.AWS CodeCommit

Jika Anda baru Git dan Anda tidak ingin mengacaukan CodeCommit repositori Anda, bereksperimen dengan sampel Git repositori pada Try Gitsitus web.

### Langkah 6: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke Anda Akun AWS setelah Anda selesai menggunakan sampel ini, hapus CodeCommit repositori. Untuk petunjuk, lihat Menghapus AWS CodeCommit Repositori di AWS CodeCommit Panduan Pengguna.

Pastikan juga untuk menghapus lingkungan. Untuk instruksi, lihat Menghapus Lingkungan.

## Amazon DynamoDB tutorial untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk mengatur lingkungan AWS Cloud9 pengembangan untuk bekerja dengan Amazon DynamoDB.

DynamoDB adalah layanan basis data NoSQL yang terkelola penuh. Anda dapat menggunakan DynamoDB untuk membuat tabel basis data yang dapat menyimpan dan mengambil data dalam jumlah berapa pun, dan melayani tingkat lalu lintas permintaan apapun. DynamoDB secara otomatis

menyebarkan data dan lalu lintas untuk tabel melalui sejumlah server yang cukup untuk menangani kapasitas permintaan yang ditentukan dan jumlah data yang disimpan, sambil mempertahankan performa yang konsisten dan cepat. Untuk informasi selengkapnya, lihat Amazon DynamoDB di situs web. AWS

Membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan DynamoDB. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon DynamoDB.

Untuk informasi tentang penawaran AWS database tambahan, lihat Amazon Relational Database Service (RDS), Amazon, ElastiCache dan Amazon Redshift di situs web. AWS Lihat juga AWS Database Migration Service di situs web AWS .

- Prasyarat
- Langkah 1: Instal dan konfigurasikan AWS CLI, yang AWS CloudShell, atau keduanya di lingkungan Anda
- Langkah 2: Buat tabel
- Langkah 3: Tambahkan item ke tabel
- Langkah 4: Tambahkan beberapa item ke tabel
- Langkah 5: Buat indeks sekunder global
- Langkah 6: Dapatkan item dari tabel
- Langkah 7: Bersihkan

## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

# Langkah 1: Instal dan konfigurasikan AWS CLI, yang AWS CloudShell, atau keduanya di lingkungan Anda

Pada langkah ini, Anda menggunakan AWS Cloud9 IDE untuk menginstal dan mengkonfigurasi AWS CLI, AWS CloudShell, atau keduanya di lingkungan Anda sehingga Anda dapat menjalankan perintah untuk berinteraksi dengan DynamoDB. Kemudian Anda menggunakan AWS CLI untuk menjalankan perintah basic DynamoDB untuk menguji pemasangan dan konfigurasi Anda.

1. Untuk mengatur manajemen kredensyal untuk AWS CLI atau AWS CloudShell dan untuk menginstal AWS CLI, AWS CloudShell, atau keduanya di lingkungan Anda, ikuti Langkah 1 dan 2 di AWS CLI dan AWS CloudShell Contoh, lalu kembali ke topik ini. Jika Anda sudah menginstal dan mengkonfigurasi AWS CLI AWS CloudShell, the, atau keduanya di lingkungan Anda, Anda tidak perlu melakukannya lagi.
2. Uji instalasi dan konfigurasi, aws-shell AWS CLI, atau keduanya dengan menjalankan **`list-tables`**perintah DynamoDB dari sesi terminal di lingkungan Anda untuk membuat daftar tabel DynamoDB yang ada, jika ada. Untuk memulai sesi terminal baru, pada bilah menu, pilih Windows, Terminal Baru.

   ```
   aws dynamodb list-tables # For the AWS CLI.
   dynamodb list-tables     # For the aws-shell.
   ```

   > **Note**
   >
   > Sepanjang sampel ini, jika Anda menggunakan aws-shell, hilangkan `aws` dari setiap perintah yang dimulai dengan `aws`. Untuk memulai aws-shell, jalankan perintah **`aws-shell`** . Untuk berhenti menggunakan aws-shell, jalankan perintah **`.exit`** atau **`.quit`** .

   Jika perintah ini berhasil, perintah ini akan mengeluarkan larik `TableNames` yang berisi daftar tabel DynamoDB yang sudah ada yang mungkin sudah Anda miliki. Jika Anda belum memiliki tabel DynamoDB, larik `TableNames` akan kosong.

   ```
   {
     "TableNames": []
   }
   ```

   Jika Anda memiliki tabel DynamoDB, larik `TableNames` berisi daftar nama tabel.

# Langkah 2: Buat Tabel

Pada langkah ini, Anda membuat tabel di DynamoDB dan menentukan nama tabel, tata letak, kunci primer sederhana, dan pengaturan throughput data.

Tabel sampel ini, dengan nama `Weather`, berisi informasi tentang prakiraan cuaca untuk beberapa kota di Amerika Serikat. Tabel menyimpan jenis informasi berikut (di DynamoDB, setiap bagian informasi dikenal sebagai atribut):

- ID kota unik yang diperlukan (`CityID`)
- Tanggal prakiraan yang diperlukan (`Date`)
- Nama kota (`City`)
- Nama negara (`State`)
- Prakiraan kondisi cuaca (`Conditions`)
- Prakiraan suhu (`Temperatures`)
  - Prakiraan tinggi, dalam derajat Fahrenheit (`HighF`)
  - Prakiraan rendah, dalam derajat Fahrenheit (`LowF`)

Untuk membuat tabel, dalam sesi terminal di AWS Cloud9 IDE, jalankan perintah DynamoDB **`create-table`**.

```
aws dynamodb create-table \
--table-name Weather \
--attribute-definitions \
  AttributeName=CityID,AttributeType=N AttributeName=Date,AttributeType=S \
--key-schema \
  AttributeName=CityID,KeyType=HASH AttributeName=Date,KeyType=RANGE \
--provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5
```

Dalam perintah ini:

- `--table-name` mewakili nama tabel (`Weather` dalam sampel ini). Nama tabel harus unik di setiap AWS Wilayah di AWS akun Anda.
- `--attribute-definitions` mewakili atribut yang digunakan untuk mengidentifikasi item tabel secara unik. Setiap item tabel ini diidentifikasi secara unik dengan kombinasi atribut `ID` numerik dan atribut `Date` yang direpresentasikan sebagai string berformat ISO-8601.

- `--key-schema` mewakili skema kunci tabel ini. Tabel memiliki kunci primer komposit `CityID` dan `Date`. Ini berarti bahwa setiap item tabel harus memiliki nilai atribut `CityID` dan nilai atribut `Date`, tetapi tidak ada dua item dalam tabel yang dapat memiliki nilai atribut `CityID` dan nilai atribut `Date` yang sama.
- `--provisioned-throughput` mewakili kapasitas baca-tulis tabel. DynamoDB memungkinkan hingga 5 bacaan konsisten per detik untuk item berukuran hingga 4 KB, atau hingga 5 bacaan akhir konsisten per detik untuk item berukuran hingga 4 KB. DynamoDB juga memungkinkan hingga 5 penulisan per detik untuk item berukuran hingga 1 KB.

> **Note**
>
> Menyetel throughput yang disediakan lebih tinggi dapat mengakibatkan biaya tambahan ke akun Anda. AWS
> Untuk informasi selengkapnya tentang ini dan perintah DynamoDB lainnya, lihat dynamodb di Referensi Perintah AWS CLI .

Jika perintah ini berhasil, ini akan menampilkan informasi ringkasan tentang tabel baru yang sedang dibuat. Untuk mengonfirmasi bahwa tabel berhasil dibuat, jalankan perintah DynamoDB **`describe-table`** , dengan menentukan nama tabel (`--table-name`).

```
aws dynamodb describe-table --table-name Weather
```

Ketika tabel berhasil dibuat, nilai `TableStatus` berubah dari `CREATING` menjadi `ACTIVE`. Jangan lanjutkan melewati langkah ini sampai tabel berhasil dibuat.

## Langkah 3: Tambahkan item ke tabel

Pada langkah ini, Anda menambahkan item ke tabel yang baru saja Anda buat.

1. Buat file dengan nama `weather-item.json` dengan konten berikut ini. Untuk membuat file baru, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.

```
{
  "CityID": { "N": "1" },
  "Date": { "S": "2017-04-12" },
  "City": { "S": "Seattle" },
  "State": { "S": "WA" },
  "Conditions": { "S": "Rain" },
```

```
  "Temperatures": { "M": {
      "HighF": { "N": "59" },
      "LowF": { "N": "46" }
    }
  }
}
```

Dalam kode ini, `N` mewakili nilai atribut yang berupa angka. `S` adalah nilai atribut string. `M` adalah atribut pemetaan, yang merupakan himpunan pasangan atribut-nilai. Anda harus menentukan tipe data atribut setiap kali Anda bekerja dengan item. Untuk tipe data atribut tambahan yang tersedia, lihat Tipe Data di Panduan Developer Amazon DynamoDB.

2. Jalankan perintah DynamoDB **`put-item`** , tentukan nama tabel (`--table-name`) dan jalur ke item berformat JSON (`--item`).

```
aws dynamodb put-item \
--table-name Weather \
--item file://weather-item.json
```

Jika perintah berhasil, itu berjalan tanpa kesalahan, dan tidak ada pesan konfirmasi yang ditampilkan.

3. Untuk mengonfirmasi bahwa tabel berhasil dibuat, jalankan perintah DynamoDB **`scan`** , dengan menentukan nama tabel (`--table-name`).

```
aws dynamodb scan --table-name Weather
```

Jika perintah berhasil, ringkasan informasi tentang tabel dan item yang baru saja Anda tambahkan akan ditampilkan.

## Langkah 4: Tambahkan beberapa item ke tabel

Pada langkah ini, Anda menambahkan beberapa item lagi ke tabel `Weather`.

1. Buat file dengan nama `more-weather-items.json` dengan konten berikut ini.

```
{
  "Weather": [
    {
      "PutRequest": {
```

```
        "Item": {
          "CityID": { "N": "1" },
          "Date": { "S": "2017-04-13" },
          "City": { "S": "Seattle" },
          "State": { "S": "WA" },
          "Conditions": { "S": "Rain" },
          "Temperatures": { "M": {
              "HighF": { "N": "52" },
              "LowF": { "N": "43" }
            }
          }
        }
      },
      {
        "PutRequest": {
          "Item": {
            "CityID": { "N": "1" },
            "Date": { "S": "2017-04-14" },
            "City": { "S": "Seattle" },
            "State": { "S": "WA" },
            "Conditions": { "S": "Rain" },
            "Temperatures": { "M": {
                "HighF": { "N": "49" },
                "LowF": { "N": "43" }
              }
            }
          }
        }
      },
      {
        "PutRequest": {
          "Item": {
            "CityID": { "N": "2" },
            "Date": { "S": "2017-04-12" },
            "City": { "S": "Portland" },
            "State": { "S": "OR" },
            "Conditions": { "S": "Thunderstorms" },
            "Temperatures": { "M": {
                "HighF": { "N": "59" },
                "LowF": { "N": "43" }
              }
            }
          }
```

```
      }
    },
    {
      "PutRequest": {
        "Item": {
          "CityID": { "N": "2" },
          "Date": { "S": "2017-04-13" },
          "City": { "S": "Portland" },
          "State": { "S": "OR" },
          "Conditions": { "S": "Rain" },
          "Temperatures": { "M": {
              "HighF": { "N": "51" },
              "LowF": { "N": "41" }
            }
          }
        }
      }
    },
    {
      "PutRequest": {
        "Item": {
          "CityID": { "N": "2" },
          "Date": { "S": "2017-04-14" },
          "City": { "S": "Portland" },
          "State": { "S": "OR" },
          "Conditions": { "S": "Rain Showers" },
          "Temperatures": { "M": {
              "HighF": { "N": "49" },
              "LowF": { "N": "39" }
            }
          }
        }
      }
    },
    {
      "PutRequest": {
        "Item": {
          "CityID": { "N": "3" },
          "Date": { "S": "2017-04-12" },
          "City": { "S": "Portland" },
          "State": { "S": "ME" },
          "Conditions": { "S": "Rain" },
          "Temperatures": { "M": {
              "HighF": { "N": "59" },
```

```
              "LowF": { "N": "40" }
            }
          }
        }
      }
    },
    {
      "PutRequest": {
        "Item": {
          "CityID": { "N": "3" },
          "Date": { "S": "2017-04-13" },
          "City": { "S": "Portland" },
          "State": { "S": "ME" },
          "Conditions": { "S": "Partly Sunny" },
          "Temperatures": { "M": {
              "HighF": { "N": "54" },
              "LowF": { "N": "37" }
            }
          }
        }
      }
    },
    {
      "PutRequest": {
        "Item": {
          "CityID": { "N": "3" },
          "Date": { "S": "2017-04-14" },
          "City": { "S": "Portland" },
          "State": { "S": "ME" },
          "Conditions": { "S": "Mostly Sunny" },
          "Temperatures": { "M": {
              "HighF": { "N": "53" },
              "LowF": { "N": "37" }
            }
          }
        }
      }
    }
  ]
}
```

Dalam kode ini, 8 objek `Item` mendefinisikan 8 item untuk ditambahkan ke tabel, mirip dengan satu item yang ditentukan pada langkah sebelumnya. Namun, saat Anda menjalankan perintah

DynamoDB **`batch-write-item`** di langkah berikutnya, Anda harus menyediakan objek berformat JSON yang menyertakan setiap objek `Item` di objek `PutRequest` yang berisi. Kemudian Anda harus menyertakan objek `PutRequest` tersebut dalam larik induk yang memiliki nama yang sama dengan tabel.

2. Jalankan perintah DynamoDB **`batch-write-item`** , dengan menetapkan jalur ke item berformat JSON untuk ditambahkan (`--request-items`).

```
aws dynamodb batch-write-item \
--request-items file://more-weather-items.json
```

Jika perintah berhasil, ini akan menampilkan pesan berikut, mengonfirmasi bahwa item berhasil ditambahkan.

```
{
  "UnprocessedItems": {}
}
```

3. Untuk mengonfirmasi konten tabel saat ini, jalankan kembali perintah DynamoDB **`scan`** .

```
aws dynamodb scan --table-name Weather
```

Jika perintah berhasil, 9 item sekarang ditampilkan.

## Langkah 5: Buat indeks sekunder global

Menjalankan perintah DynamoDB **`scan`** untuk mendapatkan informasi tentang item bisa lambat, terutama karena ukuran tabel bertambah atau jika jenis informasi yang ingin Anda dapatkan rumit. Anda dapat membuat satu atau beberapa indeks sekunder untuk mempercepat dan mempermudah mendapatkan informasi. Dalam langkah ini, Anda mempelajari tentang dua jenis indeks sekunder yang didukung DynamoDB untuk melakukan hal itu. Ini dikenal sebagai indeks sekunder lokal dan indeks sekunder global. Kemudian Anda membuat indeks sekunder global.

Untuk memahami tipe indeks sekunder ini, Anda harus terlebih dahulu mengetahui tentang kunci primer, yang secara unik mengidentifikasi item tabel. DynamoDB mendukung kunci primer sederhana atau kunci primer komposit. Kunci primer sederhana memiliki satu atribut, dan nilai atribut itu harus unik untuk setiap item dalam tabel. Atribut ini juga dikenal sebagai kunci partisi (atau atribut hash), yang dapat digunakan DynamoDB untuk mempartisi item untuk akses yang lebih cepat. Sebuah tabel juga dapat memiliki kunci primer komposit, yang berisi dua atribut. Atribut pertama adalah kunci

partisi, dan yang kedua adalah kunci pengurutan (juga dikenal sebagai atribut rentang). Dalam tabel dengan kunci primer komposit, dua item dapat memiliki nilai kunci partisi yang sama, tetapi mereka juga tidak dapat memiliki nilai kunci pengurutan yang sama. Tabel `Weather` memiliki kunci primer komposit.

Indeks sekunder lokal memiliki kunci partisi yang sama dengan tabel itu sendiri, tetapi tipe indeks ini dapat memiliki kunci pengurutan yang berbeda. Indeks sekunder global dapat memiliki kunci partisi dan kunci pengurutan yang keduanya berbeda dari tabel itu sendiri.

Misalnya, Anda sudah dapat menggunakan kunci primer untuk mengakses item `Weather` dengan `CityID`. Untuk mengakses item `Weather` dengan `State`, Anda dapat membuat indeks sekunder lokal yang memiliki kunci partisi `CityID` (harus sama dengan tabel itu sendiri) dan kunci pengurutan `State`. Untuk mengakses item `Weather` dengan `City`, Anda dapat membuat indeks sekunder global yang memiliki kunci partisi `City` dan kunci pengurutan `Date`.

Anda dapat membuat indeks sekunder lokal hanya saat Anda membuat tabel. Karena tabel `Weather` sudah ada, Anda tidak dapat menambahkan indeks sekunder lokal apa pun ke dalamnya. Namun, Anda dapat menambahkan indeks sekunder global. Berlatih menambahkan satu sekarang.

> **Note**
>
> Membuat indeks sekunder dapat mengakibatkan biaya tambahan ke akun AWS Anda.

1. Buat file dengan nama `weather-global-index.json` dengan konten berikut ini.

```
[
  {
    "Create": {
      "IndexName": "weather-global-index",
      "KeySchema": [
        {
          "AttributeName": "City",
          "KeyType": "HASH"
        },
        {
          "AttributeName": "Date",
          "KeyType": "RANGE"
        }
      ],
      "Projection": {
```

```
        "ProjectionType": "INCLUDE",
        "NonKeyAttributes": [
          "State",
          "Conditions",
          "Temperatures"
        ]
      },
      "ProvisionedThroughput": {
        "ReadCapacityUnits": 5,
        "WriteCapacityUnits": 5
      }
    }
  }
]
```

Dalam kode ini:

- Nama indeks sekunder global adalah `weather-global-index`.
- Atribut `City` adalah kunci partisi (atribut hash), dan atribut `Date` adalah kunci pengurutan (atribut range).
- `Projection` mendefinisikan atribut yang akan diambil secara default (selain atribut hash dan atribut rentang apa pun) untuk setiap item yang cocok dengan pencarian tabel yang menggunakan indeks ini. Dalam sampel ini, atribut `State`, `Conditions`, `HighF` (bagian dari `Temperatures`), dan `LowF` (juga bagian dari `Temperatures`) (serta atribut `City` dan `Date`) diambil untuk setiap item yang cocok.
- Mirip dengan tabel, indeks sekunder global harus menentukan pengaturan throughput yang disediakan.
- Pengaturan `IndexName`, `KeySchema`, `Projection`, dan `ProvisionedThroughput` harus dimuat dalam objek `Create`, yang menentukan indeks sekunder global yang akan dibuat saat Anda menjalankan perintah DynamoDB **`update-table`** di langkah berikutnya.

2. Jalankan perintah DynamoDB **`update-table`** .

```
aws dynamodb update-table \
--table-name Weather \
--attribute-definitions \
  AttributeName=City,AttributeType=S AttributeName=Date,AttributeType=S \
--global-secondary-index-updates file://weather-global-index.json
```

Dalam perintah ini:

- `--table-name` adalah nama tabel yang akan diperbarui.
- `--attribute-definitions` adalah atribut untuk dimasukkan dalam indeks. Kunci partisi selalu dicantumkan pertama, dan kunci pengurutan apa pun selalu dicantumkan kedua.
- `--global-secondary-index-updates` adalah jalur ke file yang mendefinisikan indeks sekunder global.

Jika perintah ini berhasil, ini akan menampilkan informasi ringkasan tentang indeks sekunder global baru yang sedang dibuat. Untuk mengonfirmasi bahwa indeks sekunder global berhasil dibuat, jalankan perintah DynamoDB **`describe-table`** , dengan menentukan nama tabel (`--table-name`).

```
aws dynamodb describe-table --table-name Weather
```

Ketika indeks sekunder global berhasil dibuat, nilai `TableStatus` berubah dari `UPDATING` ke `ACTIVE`, dan nilai `IndexStatus` berubah dari `CREATING` ke `ACTIVE`. Jangan lanjutkan melewati langkah ini sampai indeks sekunder global berhasil dibuat. Ini dapat memakan waktu beberapa menit.

## Langkah 6: Dapatkan item dari tabel

Ada banyak cara untuk mendapatkan item dari tabel. Dalam langkah ini, Anda mendapatkan item dengan menggunakan kunci primer tabel, dengan menggunakan atribut tabel lainnya, dan dengan menggunakan indeks sekunder global.

### Untuk mendapatkan satu item dari tabel berdasarkan nilai kunci primer item

Jika Anda mengetahui nilai kunci primer item, Anda bisa mendapatkan item yang cocok dengan menjalankan perintah DynamoDB **`get-item`** , **`scan`** , atau **`query`** . Berikut ini adalah perbedaan utama dalam perintah ini:

- **`get-item`** mengembalikan satu set atribut untuk item dengan kunci primer yang diberikan.
- **`scan`** mengembalikan satu atau lebih item dan atribut item dengan mengakses setiap item dalam tabel atau indeks sekunder.
- **`query`** menemukan item berdasarkan nilai kunci primer. Anda dapat membuat kueri tabel atau indeks sekunder apa pun yang memiliki kunci primer komposit (kunci partisi dan kunci pengurutan).

Dalam sampel ini, berikut adalah cara menggunakan masing-masing perintah ini untuk mendapatkan item yang berisi nilai atribut `CityID` dari `1` dan nilai atribut `Date` dari `2017-04-12`.

1. Untuk menjalankan perintah DynamoDB **`get-item`** , tentukan nama tabel (`--table-name`), nilai kunci primer (`--key`), dan nilai atribut untuk item yang akan ditampilkan (`--projection-expression`). Karena `Date` adalah kata kunci yang dicadangkan di DynamoDB, Anda juga harus memberikan alias untuk nilai atribut `Date` (`--expression-attribute-names`). (`State` juga merupakan kata kunci yang dicadangkan, jadi Anda akan melihat alias yang disediakan untuknya di langkah selanjutnya.)

   ```
   aws dynamodb get-item \
   --table-name Weather \
   --key '{ "CityID": { "N": "1" }, "Date": { "S": "2017-04-12" } }' \
   --projection-expression \
     "City, #D, Conditions, Temperatures.HighF, Temperatures.LowF" \
   --expression-attribute-names '{ "#D": "Date" }'
   ```

   Dalam perintah ini dan perintah lainnya, untuk menampilkan semua atribut item, jangan sertakan `--projection-expression`. Dalam contoh ini, karena Anda tidak menyertakan `--projection-expression`, Anda juga tidak perlu menyertakan `--expression-attribute-names`.

   ```
   aws dynamodb get-item \
   --table-name Weather \
   --key '{ "CityID": { "N": "1" }, "Date": { "S": "2017-04-12" } }'
   ```

2. Untuk menjalankan perintah DynamoDB **`scan`** , tentukan:

   - Nama tabel (`--table-name`).
   - Pencarian untuk dijalankan (`--filter-expression`).
   - Kriteria pencarian untuk digunakan (`--expression-attribute-values`).
   - Jenis atribut yang akan ditampilkan untuk item yang cocok (`--select`).
   - Nilai atribut untuk item yang akan ditampilkan (`--projection-expression`).
   - Jika ada atribut Anda yang menggunakan kata kunci yang dicadangkan di DynamoDB, alias untuk atribut tersebut (`--expression-attribute-names`).

   ```
   aws dynamodb scan \
   --table-name Weather \
   --filter-expression "(CityID = :cityID) and (#D = :date)" \
   ```

```
--expression-attribute-values \
  '{ ":cityID": { "N": "1" }, ":date": { "S": "2017-04-12" } }' \
--select SPECIFIC_ATTRIBUTES \
--projection-expression \
  "City, #D, Conditions, Temperatures.HighF, Temperatures.LowF" \
--expression-attribute-names '{ "#D": "Date" }'
```

3. Untuk menjalankan perintah DynamoDB **`query`** , tentukan:
   - Nama tabel (`--table-name`).
   - Pencarian untuk dijalankan (`--key-condition-expression`).
   - Nilai atribut yang akan digunakan dalam pencarian (`--expression-attribute-values`).
   - Jenis atribut yang akan ditampilkan untuk item yang cocok (`--select`).
   - Nilai atribut untuk item yang akan ditampilkan (`--projection-expression`).
   - Jika ada atribut Anda yang menggunakan kata kunci yang dicadangkan di DynamoDB, alias untuk atribut tersebut (`--expression-attribute-names`).

```
aws dynamodb query \
--table-name Weather \
--key-condition-expression "(CityID = :cityID) and (#D = :date)" \
--expression-attribute-values \
  '{ ":cityID": { "N": "1" }, ":date": { "S": "2017-04-12" } }' \
--select SPECIFIC_ATTRIBUTES \
--projection-expression \
  "City, #D, Conditions, Temperatures.HighF, Temperatures.LowF" \
--expression-attribute-names '{ "#D": "Date" }'
```

   Perhatikan bahwa perintah **`scan`** diperlukan untuk memindai semua 9 item untuk mendapatkan hasilnya, sedangkan perintah **`query`** hanya perlu memindai 1 item.

## Untuk mendapatkan beberapa item dari tabel berdasarkan nilai kunci primer item

Jika Anda mengetahui nilai kunci primer item, Anda bisa mendapatkan item yang cocok dengan menjalankan perintah DynamoDB **`batch-get-item`** . Dalam sampel ini, berikut cara mendapatkan item yang berisi nilai atribut `CityID` dari `3` dan nilai atribut `Date` dari `2017-04-13` atau `2017-04-14`.

Jalankan perintah DynamoDB **`batch-get-item`** , dengan menetapkan jalur ke file yang menjelaskan item yang akan diperoleh (`--request-items`).

```
aws dynamodb batch-get-item --request-items file://batch-get-item.json
```

Untuk sampel ini, kode dalam file `batch-get-item.json` menentukan untuk mencari tabel `Weather` untuk item dengan `CityID` dari `3` dan `Date` dari `2017-04-13` atau `2017-04-14`. Untuk setiap item yang ditemukan, nilai atribut untuk `City`, `State`, `Date`, dan `HighF` (bagian dari `Temperatures`) ditampilkan, jika ada.

```
{
  "Weather" : {
    "Keys": [
      {
        "CityID": { "N": "3" },
        "Date": { "S": "2017-04-13" }
      },
      {
        "CityID": { "N": "3" },
        "Date": { "S": "2017-04-14" }
      }
    ],
    "ProjectionExpression": "City, #S, #D, Temperatures.HighF",
    "ExpressionAttributeNames": { "#S": "State", "#D": "Date" }
  }
}
```

## Untuk mendapatkan semua item yang cocok dari sebuah tabel

Jika Anda mengetahui sesuatu tentang nilai atribut dalam tabel, Anda bisa mendapatkan item yang cocok dengan menjalankan perintah DynamoDB **`scan`** . Dalam sampel ini, berikut cara mendapatkan tanggal saat nilai atribut `Conditions` berisi `Sunny` dan nilai atribut `HighF` (bagian dari `Temperatures`) lebih besar dari `53`.

Jalankan perintah DynamoDB **`scan`** , dengan menetapkan:

- Nama tabel (`--table-name`).
- Pencarian untuk dijalankan (`--filter-expression`).
- Kriteria pencarian untuk digunakan (`--expression-attribute-values`).
- Jenis atribut yang akan ditampilkan untuk item yang cocok (`--select`).
- Nilai atribut untuk item yang akan ditampilkan (`--projection-expression`).

- Jika ada atribut Anda yang menggunakan kata kunci yang dicadangkan di DynamoDB, alias untuk atribut tersebut (`--expression-attribute-names`).

```
aws dynamodb scan \
--table-name Weather \
--filter-expression \
  "(contains (Conditions, :sun)) and (Temperatures.HighF > :h)" \
--expression-attribute-values \
  '{ ":sun": { "S" : "Sunny" }, ":h": { "N" : "53" } }' \
--select SPECIFIC_ATTRIBUTES \
--projection-expression "City, #S, #D, Conditions, Temperatures.HighF" \
--expression-attribute-names '{ "#S": "State", "#D": "Date" }'
```

### Untuk mendapatkan semua item yang cocok dari indeks sekunder global

Untuk mencari menggunakan indeks sekunder global, gunakan perintah DynamoDB **`query`** . Dalam sampel ini, berikut adalah cara menggunakan indeks sekunder `weather-global-index` untuk mendapatkan kondisi prakiraan kota bernama `Portland` untuk tanggal `2017-04-13` dan `2017-04-14`.

Jalankan perintah DynamoDB **`query`** , dengan menetapkan:

- Nama tabel (`--table-name`).
- Nama indeks sekunder global (`--index-name`).
- Pencarian untuk dijalankan (`--key-condition-expression`).
- Nilai atribut yang akan digunakan dalam pencarian (`--expression-attribute-values`).
- Jenis atribut yang akan ditampilkan untuk item yang cocok (`--select`).
- Jika ada atribut Anda yang menggunakan kata kunci yang dicadangkan di DynamoDB, alias untuk atribut tersebut (`--expression-attribute-names`).

```
aws dynamodb query \
--table-name Weather \
--index-name weather-global-index \
--key-condition-expression "(City = :city) and (#D between :date1 and :date2)" \
--expression-attribute-values \
  '{ ":city": { "S" : "Portland" }, ":date1": { "S": "2017-04-13" }, ":date2": { "S":
 "2017-04-14" } }' \
```

```
--select SPECIFIC_ATTRIBUTES \
--projection-expression "City, #S, #D, Conditions, Temperatures.HighF" \
--expression-attribute-names '{ "#S": "State", "#D": "Date" }'
```

## Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus tabel. Menghapus tabel akan menghapus indeks sekunder global juga. Anda juga harus menghapus lingkungan Anda.

Untuk menghapus tabel, jalankan perintah DynamoDB **`delete-table`** , dengan menentukan nama tabel (`--table-name`).

```
aws dynamodb delete-table --table-name Weather
```

Jika perintah berhasil, informasi tentang tabel ditampilkan, termasuk nilai `TableStatus` dari `DELETING`.

Untuk mengonfirmasi bahwa tabel berhasil dihapus, jalankan perintah DynamoDB **`describe-table`** , dengan menentukan nama tabel (`--table-name`).

```
aws dynamodb describe-table --table-name Weather
```

Jika tabel berhasil dihapus, pesan yang berisi frasa `Requested resource not found` ditampilkan.

Untuk menghapus lingkungan Anda, lihat Menghapus Lingkungan.

# AWS CDK tutorial untuk AWS Cloud9

Tutorial ini menunjukkan cara bekerja dengan lingkungan AWS Cloud9 pengembangan. AWS Cloud Development Kit (AWS CDK) AWS CDK Ini adalah seperangkat perangkat perangkat lunak dan perpustakaan yang dapat digunakan pengembang untuk memodelkan komponen AWS infrastruktur sebagai kode.

AWS CDK Termasuk AWS Construct Library yang dapat Anda gunakan untuk menyelesaikan banyak tugas dengan cepat. AWS Misalnya, Anda dapat menggunakan konstruksi `Fleet` untuk men-deploy kode secara penuh dan aman ke armada host. Anda dapat membuat konstruksi Anda

sendiri untuk memodelkan berbagai elemen arsitektur Anda, membaginya dengan orang lain, atau mempublikasikannya ke komunitas. Untuk informasi selengkapnya, lihat Panduan Developer AWS Cloud Development Kit.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2, Amazon SNS, dan Amazon SQS. Untuk informasi selengkapnya, lihat EC2 Harga Amazon, Harga Amazon SNS, dan Harga Amazon SQS.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan
- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Membersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda menginstal semua alat di lingkungan Anda yang AWS CDK perlu menjalankan sampel yang ditulis dalam bahasa TypeScript pemrograman.

1. Node Version Manager, atau **`nvm`** , yang Anda gunakan untuk memasang Node.js nanti.

2. Node.js, yang diperlukan oleh sampel dan berisi Node Package Manager **npm**, atau, yang Anda gunakan untuk menginstal TypeScript dan AWS CDK nanti.
3. TypeScript, yang dibutuhkan oleh sampel ini. (Ini AWS CDK juga menyediakan dukungan untuk beberapa bahasa pemrograman lainnya.)
4. AWS CDK.

## Langkah 1.1: Pasang Node Version Manager (nvm)

1. Dalam sesi terminal di AWS Cloud9 IDE, pastikan pembaruan keamanan terbaru dan perbaikan bug diinstal. Untuk melakukan ini, jalankan perintah **yum update** (untuk Amazon Linux) atau **apt update** (untuk Server Ubuntu). (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.)

   Untuk Amazon Linux:

   ```
   sudo yum -y update
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt update
   ```

2. Konfirmasi apakah **nvm** sudah terpasang. Untuk melakukan ini, jalankan perintah **nvm** dengan opsi **--version** .

   ```
   nvm --version
   ```

   Jika berhasil, output berisi nomor versi **nvm** , dan Anda dapat langsung melanjutkan ke Langkah 1.2: Pasang Node.js.

3. Unduh dan pasang **nvm** . Untuk melakukan ini, jalankan skrip pemasangan. Dalam contoh ini, v0.33.0 terpasang, tetapi Anda dapat memeriksa versi terbaru **nvm** di sini.

   ```
   curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.0/install.sh | bash
   ```

4. Mulai menggunakan **nvm** . Anda dapat menutup sesi terminal dan kemudian memulai kembali, atau cari file `~/.bashrc` yang berisi perintah untuk memuat **nvm** ..

   ```
   . ~/.bashrc
   ```

## Langkah 1.2: Pasang Node.js

1. Konfirmasikan apakah Anda sudah menginstal Node.js, dan jika Anda melakukannya, konfirmasikan bahwa versi yang diinstal adalah 16.17.0 atau lebih tinggi. Sampel ini telah diuji dengan Node.js 16.17.0. Untuk memeriksa, dengan sesi terminal masih terbuka di IDE, jalankan perintah **`node`** dengan opsi **`--version`** .

   ```
   node --version
   ```

   Jika Anda telah memasang Node.js, output berisi nomor versi. Jika nomor versi v16.17.0 lompat ke depan. Langkah 1.3: Instal TypeScript

2. Instal Node.js 16 dengan menjalankan **`nvm`**perintah dengan **`install`**tindakan.

   > **Note**
   >
   > Anda juga dapat menjalankan **`nvm install node`**untuk menginstal versi dukungan jangka panjang (LTS) dari Node.js. AWS Cloud9 dukungan melacak versi LTS dari Node.js.

   ```
   nvm install v16
   ```

3. Mulai gunakan Node.js 16. Untuk melakukannya, jalankan perintah **`nvm`** dengan tindakan **`alias`** , nomor versi ke alias, dan versi yang digunakan untuk alias itu, sebagai berikut.

   ```
   nvm alias default 16
   ```

   > **Note**
   >
   > Perintah sebelumnya menetapkan Node.js 16 sebagai versi default Node.js. Atau, Anda dapat menjalankan perintah **`nvm`** bersama dengan tindakan **`use`** alih-alih tindakan **`alias`** (misalnya, **`nvm use 16.17.0`** ). Namun, tindakan **`use`** menyebabkan versi Node.js tersebut hanya berjalan saat sesi terminal saat ini sedang berjalan.

4. Untuk mengonfirmasi bahwa Anda menggunakan Node.js 16 jalankan **`node --version`**perintah lagi. Jika versi yang benar diinstal, output berisi versi v16.

## Langkah 1.3: Instal TypeScript

1. Konfirmasikan apakah Anda sudah TypeScript menginstal. Untuk melakukan ini, dengan sesi terminal masih terbuka di IDE, jalankan TypeScript kompiler baris perintah dengan **`--version`**opsi.

   ```
   tsc --version
   ```

   Jika Anda telah TypeScript menginstal, output berisi nomor TypeScript versi. Jika TypeScript diinstal, lewati ke depanLangkah 1.4: Instal AWS CDK.

2. Instal TypeScript. Untuk melakukan ini, jalankan **`npm`**perintah dengan **`install`**tindakan, **`-g`**opsi, dan nama TypeScript paket. Ini diinstal TypeScript sebagai paket global di lingkungan.

   ```
   npm install -g typescript
   ```

3. Konfirmasikan bahwa TypeScript sudah diinstal. Untuk melakukan ini, jalankan TypeScript compiler baris perintah dengan **`--version`**opsi.

   ```
   tsc --version
   ```

   Jika TypeScript diinstal, output berisi nomor TypeScript versi.

## Langkah 1.4: Instal AWS CDK

1. Konfirmasikan apakah Anda sudah AWS CDK menginstal. Untuk melakukan ini, dengan sesi terminal masih terbuka di IDE, jalankan perintah **`cdk`** dengan opsi **`--version`** .

   ```
   cdk --version
   ```

   Jika AWS CDK diinstal, output berisi AWS CDK versi dan nomor build. Lewati ke Langkah 2: Menambahkan kode.

2. Instal AWS CDK dengan menjalankan **`npm`**perintah bersama dengan `install` tindakan, nama AWS CDK paket yang akan diinstal, dan `-g` opsi untuk menginstal paket secara global di lingkungan.

   ```
   npm install -g aws-cdk
   ```

3. Konfirmasikan bahwa AWS CDK sudah diinstal dan direferensikan dengan benar. Untuk melakukan ini, jalankan perintah **`cdk`** dengan opsi **`--version`**.

   ```
   cdk --version
   ```

   Jika berhasil, AWS CDK versi dan nomor build akan ditampilkan.

## Langkah 2: Menambahkan kode

Pada langkah ini, Anda membuat TypeScript proyek sampel yang berisi semua kode sumber yang Anda butuhkan untuk menyebarkan AWS CDK tumpukan secara terprogram. AWS CloudFormation Tumpukan ini membuat topik Amazon SNS dan antrean Amazon SQS di akun AWS Anda dan kemudian berlangganan antrian ke topik tersebut.

1. Dengan sesi terminal masih terbuka di IDE, buat direktori untuk menyimpan kode sumber proyek, misalnya direktori `~/environment/hello-cdk` di lingkungan Anda. Kemudian beralih ke direktori itu.

   ```
   rm -rf ~/environment/hello-cdk # Remove this directory if it already exists.
   mkdir ~/environment/hello-cdk  # Create the directory.
   cd ~/environment/hello-cdk     # Switch to the directory.
   ```

2. Siapkan direktori sebagai proyek TypeScript bahasa untuk file AWS CDK. Untuk melakukan ini, jalankan perintah **`cdk`** dengan tindakan **`init`**, templat **`sample-app`**, dan opsi **`--language`** bersama dengan nama bahasa pemrograman.

   ```
   cdk init sample-app --language typescript
   ```

   Ini membuat file dan subdirektori berikut di direktori.

   - Subdirektori `.git` tersembunyi dan file `.gitignore` tersembunyi, yang membuat proyek kompatibel dengan alat pengendali sumber seperti Git.
   - Subdirektori `lib`, yang menyertakan file `hello-cdk-stack.ts`. File ini berisi kode untuk AWS CDK tumpukan Anda. Kode ini dijelaskan pada langkah berikutnya dalam prosedur ini.
   - Subdirektori `bin`, yang menyertakan file `hello-cdk.ts`. File ini berisi titik masuk untuk AWS CDK aplikasi Anda.

- Subdirektori `node_modules`, yang berisi paket kode pendukung yang dapat digunakan aplikasi dan tumpukan sesuai kebutuhan.
- File `.npmignore` tersembunyi, yang mencantumkan jenis subdirektori dan file yang tidak dibutuhkan **`npm`** saat membuat kode.
- File `cdk.json`, yang berisi informasi untuk membuat menjalankan perintah **`cdk`** lebih mudah.
- File `package-lock.json`, yang berisi informasi yang dapat digunakan **`npm`** untuk mengurangi kemungkinan kesalahan pembangunan dan pelaksanaan.
- File `package.json`, yang berisi informasi untuk membuat menjalankan perintah **`npm`** lebih mudah dan dengan kemungkinan kesalahan pembangunan dan pelaksanaan yang lebih sedikit.
- `README.md`File, yang mencantumkan perintah berguna yang dapat Anda jalankan **`npm`**dan file AWS CDK.
- File `tsconfig.json`, yang berisi informasi untuk membuat menjalankan perintah **`tsc`** lebih mudah dan dengan kemungkinan kesalahan pembangunan dan pelaksanaan yang lebih sedikit.

3. Di jendela Lingkungan, buka file `lib/hello-cdk-stack.ts`, dan telusuri kode berikut di file itu.

```
import sns = require('@aws-cdk/aws-sns');
import sqs = require('@aws-cdk/aws-sqs');
import cdk = require('@aws-cdk/cdk');

export class HelloCdkStack extends cdk.Stack {
  constructor(parent: cdk.App, name: string, props?: cdk.StackProps) {
    super(parent, name, props);

    const queue = new sqs.Queue(this, 'HelloCdkQueue', {
      visibilityTimeoutSec: 300
    });

    const topic = new sns.Topic(this, 'HelloCdkTopic');

    topic.subscribeQueue(queue);
  }
}
```

- `TopicKelas``Stack`,`App`,`StackProps`,`Queue`, dan mewakili AWS CloudFormation tumpukan dan propertinya, program yang dapat dieksekusi, antrian Amazon SQS, dan topik Amazon SNS, masing-masing.
- `HelloCdkStack`Kelas mewakili AWS CloudFormation tumpukan untuk aplikasi ini. Tumpukan ini berisi antrean Amazon SQS baru dan topik Amazon SNS untuk aplikasi ini.

4. Di jendela Lingkungan, buka file `bin/hello-cdk.ts`, dan telusuri kode berikut di file itu.

```
#!/usr/bin/env node
import cdk = require('@aws-cdk/cdk');
import { HelloCdkStack } from '../lib/hello-cdk-stack';

const app = new cdk.App();
new HelloCdkStack(app, 'HelloCdkStack');
app.run();
```

   Kode ini memuat, membuat instans, dan kemudian menjalankan kelas `HelloCdkStack` dari file `lib/hello-cdk-stack.ts`.

5. Gunakan **`npm`**untuk menjalankan TypeScript kompiler untuk memeriksa kesalahan pengkodean, dan kemudian aktifkan AWS CDK untuk mengeksekusi `bin/hello-cdk.js` file proyek. Untuk melakukannya, dari direktori root proyek, jalankan perintah **`npm`** dengan tindakan **`run`** , dengan menetapkan nilai perintah **`build`** dalam file `package.json`, sebagai berikut.

```
npm run build
```

   Perintah sebelumnya menjalankan TypeScript kompiler, yang menambahkan dukungan `bin/hello-cdk.d.ts` dan file. `lib/hello-cdk-stack.d.ts` Kompiler juga mentranspilasikan file `hello-cdk.ts` dan `hello-cdk-stack.ts` menjadi file `hello-cdk.js` dan `hello-cdk-stack.js`.

## Langkah 3: Jalankan kode

Pada langkah ini, Anda menginstruksikan AWS CDK untuk membuat template AWS CloudFormation tumpukan berdasarkan kode dalam `bin/hello-cdk.js` file. Anda kemudian menginstruksikan AWS CDK untuk menerapkan tumpukan, yang membuat topik Amazon SNS dan antrean Amazon SQS dan kemudian berlangganan antrian ke topik. Anda kemudian mengonfirmasi bahwa topik dan antrean berhasil ter-deploy dengan mengirimkan pesan dari topik ke antrean.

1. Minta template AWS CDK buat AWS CloudFormation tumpukan. Untuk melakukan ini, dengan sesi terminal masih terbuka di IDE, dari direktori root proyek, jalankan perintah **`cdk`** dengan tindakan **`synth`** dan nama tumpukan.

   ```
   cdk synth HelloCdkStack
   ```

   Jika berhasil, output menampilkan `Resources` bagian AWS CloudFormation stack template ini.

2. Pertama kali Anda menerapkan AWS CDK aplikasi ke lingkungan untuk AWS akun tertentu dan kombinasi AWS Wilayah, Anda harus menginstal tumpukan bootstrap. Tumpukan ini mencakup berbagai sumber daya yang AWS CDK dibutuhkan untuk menyelesaikan berbagai operasinya. Misalnya, tumpukan ini menyertakan bucket Amazon S3 yang AWS CDK digunakan untuk menyimpan templat dan aset selama proses penerapannya. Untuk memasang tumpukan bootstrap, jalankan perintah **`cdk`** dengan tindakan **`bootstrap`** .

   ```
   cdk bootstrap
   ```

   > **Note**
   >
   > Jika Anda menjalankan `cdk bootstrap` tanpa menentukan opsi apa pun, AWS akun default dan AWS Wilayah digunakan. Anda juga dapat mem-bootstrap lingkungan tertentu dengan menentukan profil dan kombinasi akun/Wilayah. Sebagai contoh:
   >
   > ```
   > cdk bootstrap --profile test 123456789012/us-east-1
   > ```

3. AWS CDK Jalankan template AWS CloudFormation tumpukan untuk menyebarkan tumpukan. Untuk melakukan ini, dari direktori root proyek, jalankan perintah **`cdk`** dengan tindakan **`deploy`** dan nama tumpukan.

   ```
   cdk deploy HelloCdkStack
   ```

   Jika berhasil, output menampilkan bahwa tumpukan `HelloCdkStack` di-deploy tanpa kesalahan.

   > **Note**
   >
   > Jika output menampilkan pesan bahwa tumpukan tidak menentukan lingkungan dan AWS kredensialnya tidak dapat diperoleh dari lokasi standar atau tidak ada wilayah yang

dikonfigurasi, pastikan AWS kredensialnya disetel dengan benar di IDE, lalu jalankan perintahnya lagi. **`cdk deploy`** Untuk informasi selengkapnya, lihat Menelepon Layanan AWS dari lingkungan di AWS Cloud9.

4. Untuk mengonfirmasi bahwa topik Amazon SNS dan antrean Amazon SQS berhasil di-deploy, kirim pesan ke topik tersebut, lalu periksa antrean untuk pesan yang diterima. Untuk melakukan ini, Anda dapat menggunakan alat seperti AWS Command Line Interface (AWS CLI) atau AWS CloudShell. Untuk informasi lebih lanjut tentang alat ini, lihat AWS CLI dan tutorial aws-shell untuk AWS Cloud9.

   Misalnya, untuk mengirim pesan ke topik, dengan sesi terminal masih terbuka di IDE, gunakan perintah AWS CLI untuk menjalankan **`publish`**perintah Amazon SNS, memasok subjek dan isi pesan, AWS Wilayah untuk topik, dan Nama Sumber Daya Amazon topik (ARN).

   ```
   aws sns publish --subject "Hello from the AWS CDK" --message "This is a message
    from the AWS CDK." --topic-arn arn:aws:sns:us-east-2:123456789012:HelloCdkStack-
   HelloCdkTopic1A234567-8BCD9EFGHIJ0K
   ```

   Dalam perintah sebelumnya, ganti `arn:aws:sns:us-east-2:123456789012:HelloCdkStack-HelloCdkTopic1A234567-8BCD9EFGHIJ0K` dengan ARN yang AWS CloudFormation menetapkan topik. Untuk mendapatkan ID, Anda dapat menjalankan perintah **`list-topics`** Amazon SNS.

   ```
   aws sns list-topics --output table --query 'Topics[*].TopicArn'
   ```

   Jika berhasil, output dari perintah **`publish`** menampilkan nilai `MessageId` untuk pesan yang dipublikasikan.

   Untuk memeriksa antrean pesan yang diterima, jalankan perintah **`receive-message`** Amazon SQS, yang menyediakan URL antrean.

   ```
   aws sqs receive-message --queue-url https://queue.amazonaws.com/123456789012/
   HelloCdkStack-HelloCdkQueue1A234567-8BCD9EFGHIJ0K
   ```

   Pada perintah sebelumnya, ganti `https://queue.amazonaws.com/123456789012/HelloCdkStack-HelloCdkQueue1A234567-8BCD9EFGHIJ0K` dengan ARN yang AWS CloudFormation menetapkan antrian. Untuk mendapatkan URL, Anda dapat menjalankan perintah **`list-queues`** Amazon SQS.

```
aws sqs list-queues --output table --query 'QueueUrls[*]'
```

Jika berhasil, output dari perintah **`receive-message`** menampilkan informasi tentang pesan yang diterima.

## Langkah 4: Membersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus AWS CloudFormation tumpukan. Ini akan menghapus topik Amazon SNS dan antrean Amazon SQS. Anda juga harus menghapus lingkungan.

### Langkah 4.1: Hapus tumpukan

Dengan sesi terminal masih terbuka di IDE, dari direktori root proyek, jalankan perintah **`cdk`** dengan tindakan **`destroy`** dan nama tumpukan.

```
cdk destroy HelloCdkStack
```

Saat diminta untuk menghapus tumpukan, ketik `y`, lalu tekan `Enter`.

Jika berhasil, output menampilkan bahwa tumpukan `HelloCdkStack` dihapus tanpa kesalahan.

### Langkah 4.2: Hapus lingkungan

Untuk menghapus lingkungan, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial LAMP untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk mengatur dan menjalankan LAMP (Linux, Apache Server HTTP, MySQL, dan PHP) dalam lingkungan AWS Cloud9 pembangunan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya untuk Anda Akun AWS. Ini termasuk kemungkinan biaya untuk Layanan AWS seperti Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

Topik


- Prasyarat
- Langkah 1: Pasang alatnya

- Langkah 2: Mengatur MySQL
- Langkah 3: Mengatur situs web
- Langkah 4: Membersihkan

## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alatnya

Dalam langkah ini, Anda memasang alat-alat berikut:

- Apache HTTP Server, host server web.
- PHP, bahasa scripting yang sangat cocok untuk pengembangan web dan dapat disematkan ke dalam HTML.
- MySQL, sistem manajemen basis data.

Anda kemudian menyelesaikan langkah ini dengan memulai Apache HTTP Server dan kemudian MySQL.

1. Pastikan pembaruan keamanan terbaru dan perbaikan bug telah terpasang pada instans. Untuk melakukan ini, dalam sesi terminal di AWS Cloud9 IDE, jalankan **`yum update`**for (Amazon Linux) atau **`apt update`**untuk (Ubuntu Server) perintah. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.)

   Untuk Amazon Linux:

```
sudo yum -y update
```

Untuk Ubuntu Peladen:

```
sudo apt -y update
```

2. Periksa apakah Apache HTTP Server sudah diinstal. Untuk melakukan ini, jalankan **`httpd -v`**(untuk Amazon Linux) atau **`apache2 -v`**(untuk Ubuntu Server) perintah.

   Jika berhasil, output berisi Apache Nomor versi HTTP Server.

   Jika Anda melihat kesalahan, maka instal Apache HTTP Server dengan menjalankan **`install`**perintah.

   Untuk Amazon Linux:

   ```
   sudo yum install -y httpd24
   ```

   Untuk Ubuntu Peladen:

   ```
   sudo apt install -y apache2
   ```

3. Konfirmasikan apakah PHP sudah diinstal dengan menjalankan **`php -v`**perintah.

   Jika berhasil, output berisi nomor versi PHP.

   Jika Anda melihat kesalahan, maka instal PHP dengan menjalankan **`install`**perintah.

   Untuk Amazon Linux:

   ```
   sudo yum install -y php56
   ```

   Untuk Ubuntu Peladen:

   ```
   sudo apt install -y php libapache2-mod-php php-xml
   ```

4. Konfirmasikan apakah MySQL sudah diinstal dengan menjalankan **`mysql --version`**perintah.

   Jika berhasil, output berisi MySQL nomor versi.

Jika Anda melihat kesalahan, maka instal MySQL dengan menjalankan **`install`**perintah.

Untuk Amazon Linux:

```
sudo yum install -y mysql-server
```

Untuk Ubuntu Peladen:

```
sudo apt install -y mysql-server
```

5. Setelah Anda menginstal Apache Server HTTP, PHP, dan MySQL, mulai Apache HTTP Server, dan kemudian konfirmasikan telah dimulai, dengan menjalankan perintah berikut.

   Untuk Amazon Linux (Anda mungkin perlu menjalankan perintah ini dua kali):

   ```
   sudo service httpd start && sudo service httpd status
   ```

   Untuk Ubuntu Server (untuk kembali ke command prompt, tekanq):

   ```
   sudo service apache2 start && sudo service apache2 status
   ```

6. Mulai MySQL, dan kemudian konfirmasikan sudah dimulai, dengan menjalankan perintah berikut.

   Untuk Amazon Linux:

   ```
   sudo service mysqld start && sudo service mysqld status
   ```

   Untuk Ubuntu Server (untuk kembali ke command prompt, tekanq):

   ```
   sudo service mysql start && sudo service mysql status
   ```

## Langkah 2: Mengatur MySQL

Pada langkah ini, Anda mengatur MySQL untuk mengikuti MySQL praktik terbaik keamanan. Praktik terbaik keamanan ini mencakup pengaturan kata sandi untuk akun root dan menghapus akun root yang dapat diakses dari luar host lokal. Praktik terbaik lainnya yang harus diperhatikan adalah

menghapus pengguna anonim, menghapus database pengujian, dan menghapus hak istimewa yang memungkinkan siapa pun untuk mengakses database dengan nama yang dimulai dengan. `test_`

Anda kemudian menyelesaikan langkah ini dengan berlatih memulai dan kemudian keluar dari MySQL klien baris perintah.

1. Implementasikan MySQL praktik terbaik keamanan untuk MySQL instalasi dengan menjalankan perintah berikut dalam sesi terminal di AWS Cloud9 IDE.

   ```
   sudo mysql_secure_installation
   ```

2. Saat diminta, jawablah pertanyaan berikut seperti yang ditentukan.

   Untuk Amazon Linux:

   1. Masukkan kata sandi saat ini untuk root (masukkan untuk tidak ada) – Tekan `Enter` (tanpa kata sandi).
   2. Atur kata sandi root – Ketik `Y`, lalu tekan `Enter`.
   3. Kata sandi baru – Ketik kata sandi, lalu tekan `Enter`.
   4. Masukkan kembali kata sandi baru – Ketikkan kata sandi lagi, lalu tekan `Enter`. (Pastikan untuk menyimpan kata sandi di lokasi yang aman untuk digunakan nanti.)
   5. Hapus pengguna anonim – Ketik `Y`, lalu tekan `Enter`.
   6. Larang login root dari jarak jauh – Ketik `Y`, lalu tekan `Enter`.
   7. Hapus basis data pengujian dan akses ke sana – Ketik `Y`, lalu tekan `Enter`.
   8. Muat ulang tabel hak istimewa sekarang – Ketik `Y`, lalu tekan `Enter`.

   Untuk Ubuntu Peladen:

   1. Apakah Anda ingin mengatur plugin VALIDATE PASSWORD — Entery, lalu tekan`Enter`.
   2. Ada tiga tingkat kebijakan validasi kata sandi - Enter`0`,`1`, atau`2`, dan kemudian tekan`Enter`.
   3. Kata sandi baru — Masukkan kata sandi, lalu tekan`Enter`.
   4. Masukkan kembali kata sandi baru — Masukkan kata sandi lagi, lalu tekan`Enter`. Pastikan Anda menyimpan kata sandi di lokasi yang aman untuk digunakan nanti.
   5. Apakah Anda ingin melanjutkan dengan kata sandi yang disediakan — Entery, lalu tekan`Enter`.
   6. Hapus pengguna anonim — Entery, lalu tekan`Enter`.

7. Larang login root dari jarak jauh — Entery, lalu tekan. `Enter`
8. Hapus database uji dan akses ke sana — Entery, lalu tekan`Enter`.
9. Muat ulang tabel hak istimewa sekarang — Entery, lalu tekan. `Enter`

3. Untuk berinteraksi langsung dengan MySQL, mulai MySQL command line client sebagai root user dengan menjalankan perintah berikut. Saat diminta, ketik kata sandi pengguna root yang Anda atur sebelumnya, lalu tekan `Enter`. Prompt berubah menjadi `mysql>` saat Anda berada di MySQL klien baris perintah.

```
sudo mysql -uroot -p
```

4. Untuk keluar dari MySQL command line client, jalankan perintah berikut. Prompt berubah kembali ke$.

```
exit;
```

## Langkah 3: Mengatur situs web

Pada langkah ini, Anda mengatur root situs web default untuk Apache Server HTTP dengan pemilik yang direkomendasikan dan izin akses. Anda kemudian membuat PHPhalaman web berbasis dalam root situs web default itu.

Anda kemudian mengaktifkan lalu lintas web yang masuk untuk melihat halaman web tersebut dengan menyiapkan grup keamanan di Amazon EC2 dan daftar kontrol akses jaringan (network ACL) di Amazon Virtual Private Cloud (Amazon VPC) yang terkait dengan lingkungan ini. EC2 Setiap EC2 lingkungan harus dikaitkan dengan grup keamanan di Amazon EC2 dan ACL jaringan di Amazon VPC. Namun, meskipun ACL jaringan default dalam Akun AWS memungkinkan semua lalu lintas masuk dan keluar untuk lingkungan, grup keamanan default hanya mengizinkan lalu lintas masuk menggunakan SSH melalui port 22. Untuk informasi selengkapnya, lihat the section called “Pengaturan Amazon VPC”.

Anda kemudian menyelesaikan langkah ini dengan berhasil melihat halaman web dari luar AWS Cloud9 IDE.

1. Siapkan root situs web default untuk Apache HTTP Server (`/var/www/html`) dengan pemilik yang direkomendasikan dan izin akses. Untuk melakukan ini, jalankan enam perintah berikut, satu per satu dalam urutan berikut, dalam sesi terminal di AWS Cloud9 IDE. Untuk memahami apa yang dilakukan setiap perintah, baca informasi setelah karakter # setelah setiap perintah.

Untuk Amazon Linux:

```
sudo groupadd web-content # Create a group named web-content.

sudo usermod -G web-content -a ec2-user # Add the user ec2-user (your default user
 for this environment) to the group web-content.

sudo usermod -G web-content -a apache # Add the user apache (Apache HTTP Server) to
 the group web-content.

sudo chown -R ec2-user:web-content /var/www/html # Change the owner of /var/www/
html and its files to user ec2-user and group web-content.

sudo find /var/www/html -type f -exec chmod u=rw,g=rx,o=rx {} \; # Change all file
 permissions within /var/www/html to user read/write, group read-only, and others
 read/execute.

sudo find /var/www/html -type d -exec chmod u=rwx,g=rx,o=rx {} \; # Change /var/
www/html directory permissions to user read/write/execute, group read/execute, and
 others read/execute.
```

Untuk Ubuntu Peladen:

```
sudo groupadd web-content # Create a group named web-content.

sudo usermod -G web-content -a ubuntu # Add the user ubuntu (your default user for
 this environment) to the group web-content.

sudo usermod -G web-content -a www-data # Add the user www-data (Apache HTTP
 Server) to the group web-content.

sudo chown -R ubuntu:web-content /var/www/html # Change the owner of /var/www/html
 and its files to user ubuntu and group web-content.

sudo find /var/www/html -type f -exec chmod u=rw,g=rx,o=rx {} \; # Change all file
 permissions within /var/www/html to user read/write, group read-only, and others
 read/execute.

sudo find /var/www/html -type d -exec chmod u=rwx,g=rx,o=rx {} \; # Change /var/
www/html directory permissions to user read/write/execute, group read/execute, and
 others read/execute.
```

2. Buat PHPhalaman web berbasis yang dinamai `index.php` dalam folder root situs web default untuk Apache HTTP Server (yaitu`/var/www/html`) dengan menjalankan perintah berikut.

   Untuk Amazon Linux:

   ```
   sudo touch /var/www/html/index.php && sudo chown -R ec2-user:web-content /var/www/
   html/index.php && sudo chmod u=rw,g=rx,o=rx /var/www/html/index.php && sudo printf
    '%s\n%s\n%s' '<?php' '  phpinfo();' '?>' >> /var/www/html/index.php
   ```

   Perintah sebelumnya untuk Amazon Linux juga mengubah pemilik file menjadi`ec2-user`, mengubah grup file menjadi`web-content`, dan mengubah izin file read/write for the user, and read/execute untuk grup dan lainnya.

   Untuk Ubuntu Peladen:

   ```
   sudo touch /var/www/html/index.php && sudo chown -R ubuntu:web-content /var/www/
   html/index.php && sudo chmod u=rw,g=rx,o=rx /var/www/html/index.php && sudo printf
    '%s\n%s\n%s' '<?php' '  phpinfo();' '?>' >> /var/www/html/index.php
   ```

   Perintah sebelumnya untuk Ubuntu Server juga mengubah pemilik file menjadi`ubuntu`, mengubah grup file menjadi`web-content`, dan mengubah izin file read/write for the user, and read/execute untuk grup dan lainnya.

   Jika berhasil, perintah sebelumnya membuat file `index.php` dengan konten berikut.

   ```
   <?php
     phpinfo();
   ?>
   ```

3. Aktifkan lalu lintas web masuk melalui port 80 untuk melihat halaman web baru dengan menyiapkan ACL jaringan di Amazon VPC dan grup keamanan Amazon EC2 yang terkait dengan lingkungan ini. EC2 Untuk melakukannya, jalankan delapan perintah berikut, satu per satu dalam urutan berikut. Untuk memahami apa yang dilakukan setiap perintah, baca informasi setelah karakter # untuk setiap perintah.

   > **Important**
   >
   > Menjalankan perintah berikut memungkinkan lalu lintas web masuk melalui port 80 untuk semua EC2 lingkungan dan EC2 instans Amazon yang terkait dengan grup keamanan

dan ACL jaringan untuk lingkungan ini. Hal ini dapat mengakibatkan secara tak terduga memungkinkan lalu lintas web masuk melalui port 80 untuk lingkungan EC2 dan EC2 contoh Amazon selain yang satu ini.

**Note**

Perintah kedua hingga keempat berikut memungkinkan grup keamanan untuk mengizinkan lalu lintas web masuk melalui port 80. Jika Anda memiliki grup keamanan default, yang hanya mengizinkan lalu lintas SSH masuk melalui port 22, maka Anda harus menjalankan perintah pertama diikuti oleh perintah kedua hingga keempat ini. Namun, jika Anda memiliki grup keamanan khusus yang sudah memungkinkan lalu lintas web masuk melalui port 80, Anda dapat melewati menjalankan perintah tersebut. Perintah kelima hingga kedelapan berikut mengaktifkan ACL jaringan untuk mengizinkan lalu lintas web masuk melalui port 80. Jika Anda memiliki ACL jaringan default, yang sudah mengizinkan semua lalu lintas masuk melalui semua port, maka Anda dapat melewati perintah tersebut dengan aman. Namun, misalkan Anda memiliki ACL jaringan khusus yang tidak mengizinkan lalu lintas web masuk melalui port 80. Kemudian, jalankan perintah pertama diikuti oleh perintah kelima hingga kedelapan ini.

```
MY_INSTANCE_ID=$(curl http://169.254.169.254/latest/meta-data/instance-id) # Get
 the ID of the instance for the environment, and store it temporarily.

MY_SECURITY_GROUP_ID=$(aws ec2 describe-instances --instance-id $MY_INSTANCE_ID
 --query 'Reservations[].Instances[0].SecurityGroups[0].GroupId' --output text)
 # Get the ID of the security group associated with the instance, and store it
 temporarily.

aws ec2 authorize-security-group-ingress --group-id $MY_SECURITY_GROUP_ID --
protocol tcp --cidr 0.0.0.0/0 --port 80 # Add an inbound rule to the security group
 to allow all incoming IPv4-based traffic over port 80.

aws ec2 authorize-security-group-ingress --group-id $MY_SECURITY_GROUP_ID --ip-
permissions IpProtocol=tcp,Ipv6Ranges='[{CidrIpv6=::/0}]',FromPort=80,ToPort=80 #
 Add an inbound rule to the security group to allow all incoming IPv6-based traffic
 over port 80.
```

```
MY_SUBNET_ID=$(aws ec2 describe-instances --instance-id $MY_INSTANCE_ID --query
 'Reservations[].Instances[0].SubnetId' --output text) # Get the ID of the subnet
 associated with the instance, and store it temporarily.

MY_NETWORK_ACL_ID=$(aws ec2 describe-network-acls --filters
 Name=association.subnet-id,Values=$MY_SUBNET_ID --query
 'NetworkAcls[].Associations[0].NetworkAclId' --output text) # Get the ID of the
 network ACL associated with the subnet, and store it temporarily.

aws ec2 create-network-acl-entry --network-acl-id $MY_NETWORK_ACL_ID --ingress --
protocol tcp --rule-action allow --rule-number 10000 --cidr-block 0.0.0.0/0 --port-
range From=80,To=80 # Add an inbound rule to the network ACL to allow all IPv4-
based traffic over port 80. Advanced users: change this suggested rule number as
 desired.

aws ec2 create-network-acl-entry --network-acl-id $MY_NETWORK_ACL_ID --ingress --
protocol tcp --rule-action allow --rule-number 10100 --ipv6-cidr-block ::/0 --port-
range From=80,To=80 # Add an inbound rule to the network ACL to allow all IPv6-
based traffic over port 80. Advanced users: change this suggested rule number as
 desired.
```

4. Dapatkan URL ke file `index.php` di dalam root server web. Untuk melakukan ini, jalankan perintah berikut, dan gunakan tab browser web baru atau browser web lain yang terpisah dari AWS Cloud9 IDE untuk pergi ke URL yang ditampilkan. Jika berhasil, halaman web menampilkan informasi tentang Apache Server HTTP, MySQL, PHP, dan pengaturan terkait lainnya.

```
MY_PUBLIC_IP=$(curl http://169.254.169.254/latest/meta-data/public-ipv4) && echo
 http://$MY_PUBLIC_IP/index.php # Get the URL to the index.php file within the web
 server root.
```

## Langkah 4: Membersihkan

Misalkan Anda ingin tetap menggunakan lingkungan ini tetapi Anda ingin menonaktifkan lalu lintas web masuk melalui port 80. Kemudian, jalankan delapan perintah berikut, satu per satu dalam urutan berikut, untuk menghapus peraturan lalu lintas masuk yang sesuai yang Anda tetapkan sebelumnya di grup keamanan dan ACL jaringan yang terkait dengan lingkungan. Untuk memahami apa yang dilakukan setiap perintah, baca informasi setelah karakter # untuk setiap perintah.

**Important**

Menjalankan perintah berikut menonaktifkan lalu lintas web yang masuk melalui port 80 untuk semua EC2 lingkungan dan EC2 instans Amazon yang terkait dengan grup keamanan dan ACL jaringan untuk lingkungan ini. Hal ini dapat mengakibatkan secara tak terduga menonaktifkan lalu lintas web masuk melalui port 80 untuk lingkungan EC2 dan EC2 contoh Amazon selain yang satu ini.

**Note**

Perintah kelima hingga kedelapan berikut menghapus aturan yang ada untuk memblokir ACL jaringan agar tidak memungkinkan lalu lintas web masuk melalui port 80. Jika Anda memiliki jaringan default ACL, yang sudah memungkinkan semua lalu lintas masuk ke semua port, maka Anda dapat melewati menjalankan perintah tersebut. Namun, misalkan Anda memiliki ACL jaringan khusus dengan aturan yang ada yang memungkinkan lalu lintas web masuk melalui port 80 dan Anda ingin menghapus aturan tersebut. Kemudian, Anda perlu menjalankan perintah pertama diikuti oleh perintah kelima hingga kedelapan ini.

```
MY_INSTANCE_ID=$(curl http://169.254.169.254/latest/meta-data/instance-id) # Get the ID
 of the instance for the environment, and store it temporarily.

MY_SECURITY_GROUP_ID=$(aws ec2 describe-instances --instance-id $MY_INSTANCE_ID --query
 'Reservations[].Instances[0].SecurityGroups[0].GroupId' --output text) # Get the ID of
 the security group associated with the instance, and store it temporarily.

aws ec2 revoke-security-group-ingress --group-id $MY_SECURITY_GROUP_ID --protocol tcp
 --cidr 0.0.0.0/0 --port 80 # Delete the existing inbound rule from the security group
 to block all incoming IPv4-based traffic over port 80.

aws ec2 revoke-security-group-ingress --group-id $MY_SECURITY_GROUP_ID --ip-permissions
 IpProtocol=tcp,Ipv6Ranges='[{CidrIpv6=::/0}]',FromPort=80,ToPort=80 # Delete the
 existing inbound rule from the security group to block all incoming IPv6-based traffic
 over port 80.

MY_SUBNET_ID=$(aws ec2 describe-instances --instance-id $MY_INSTANCE_ID --query
 'Reservations[].Instances[0].SubnetId' --output text) # Get the ID of the subnet
 associated with the instance, and store it temporarily.
```

```
MY_NETWORK_ACL_ID=$(aws ec2 describe-network-acls --filters Name=association.subnet-
id,Values=$MY_SUBNET_ID --query 'NetworkAcls[].Associations[0].NetworkAclId' --output
 text) # Get the ID of the network ACL associated with the subnet, and store it
 temporarily.

aws ec2 delete-network-acl-entry --network-acl-id $MY_NETWORK_ACL_ID --ingress --rule-
number 10000 # Delete the existing inbound rule from the network ACL to block all IPv4-
based traffic over port 80. Advanced users: if you originally created this rule with a
 different number, change this suggested rule number to match.

aws ec2 delete-network-acl-entry --network-acl-id $MY_NETWORK_ACL_ID --ingress --rule-
number 10100 # Delete the existing inbound rule from the network ACL to block all IPv6-
based traffic over port 80. Advanced users: if you originally created this rule with a
 different number, change this suggested rule number to match.
```

Jika Anda selesai menggunakan lingkungan ini, hapus lingkungan untuk mencegah tagihan yang sedang berlangsung ke lingkungan Anda Akun AWS. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# WordPress tutorial untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menginstal dan menjalankan WordPress dalam lingkungan AWS Cloud9 pengembangan. WordPress adalah sistem manajemen konten open-source (CMS) yang banyak digunakan untuk pengiriman konten web.

> **Note**
>
> Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon Elastic Compute Cloud (Amazon EC2). Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis

lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.

- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

- Anda memiliki up-to-date EC2 instance dengan semua paket perangkat lunak terbaru. Di jendela terminal AWS Cloud9 IDE, Anda dapat menjalankan `yum update` dengan `-y` opsi untuk menginstal pembaruan tanpa meminta konfirmasi. Jika Anda ingin memeriksa pembaruan sebelum menginstal, Anda dapat menghapus opsi ini.

```
sudo yum update -y
```

## Gambaran umum instalasi

Menginstal WordPress pada EC2 instance lingkungan Anda melibatkan langkah-langkah berikut:

1. Menginstal dan mengkonfigurasi MariaDB Server, yang merupakan database relasional open-source yang menyimpan informasi untuk instalasi WordPress
2. Menginstal dan mengonfigurasi WordPress, yang mencakup pengeditan file `wordpress.conf` konfigurasi
3. Mengkonfigurasi server Apache yang menghosting situs WordPress
4. Mempratinjau konten WordPress web yang di-host oleh server Apache

## Langkah 1: Menginstal dan mengkonfigurasi MariaDB Server

1. Di AWS Cloud9 IDE, pilih Window, New Terminal dan masukkan perintah berikut untuk menginstal dan memulai instalasi MariaDB Server:

   ```
   sudo yum install -y mariadb-server
   sudo systemctl start mariadb
   ```

2. Selanjutnya, jalankan `mysql_secure_installation` penulisan untuk meningkatkan keamanan instalasi MariaDB Server Anda.

Saat memberikan tanggapan terhadap skrip, tekan Masukkanuntuk pertanyaan pertama agar kata sandi admin tetap kosong. Tekan n untuk `Set root password?` dan kemudian y untuk masing-masing pilihan keamanan lainnya.

```
mysql_secure_installation
```

3. Sekarang buat tabel database untuk menyimpan WordPress informasi menggunakan klien MariaDB.

   (Tekan Masukkan ketika diminta untuk memasukkan kata sandi Anda.)

```
sudo mysql -u root -p
MariaDB [(none)]> create database wp_test;
MariaDB [(none)]> grant all privileges on wp_test.* to root@localhost identified by
 ';'
```

4. Untuk log out dari klien MariaDB, jalankan `exit` perintah.

## Langkah 2: Menginstal dan mengkonfigurasi WordPress

1. Di jendela terminal IDE, arahkan ke `environment` direktori, lalu buat direktori `config` dan `wordpress`. Kemudian jalankan perintah `touch` untuk membuat sebuah file bernama `wordpress.conf` di `config` direktori:

```
cd /home/ec2-user/environment
mkdir config wordpress
touch config/wordpress.conf
```

2. Gunakan editor IDE atau vim untuk memperbarui `wordpress.conf` dengan informasi konfigurasi host yang memungkinkan server Apache untuk menyajikan WordPress konten:

```
# Ensure that Apache listens on port 80
Listen 8080
<VirtualHost *:8080>
    DocumentRoot "/var/www/wordpress"
    ServerName www.example.org
    # Other directives here
</VirtualHost>
```

3. Sekarang jalankan perintah berikut untuk mengambil file arsip yang diperlukan dan menginstal WordPress:

```
cd /home/ec2-user/environment
wget https://wordpress.org/latest.tar.gz
tar xvf latest.tar.gz
```

4. Jalankan perintah `touch` untuk membuat sebuah file bernama `wp-config.php` di `environment/wordpress` direktori:

```
touch wordpress/wp-config.php
```

5. Gunakan editor IDE atau vim untuk memperbarui `wp-config.php` dan mengganti data sampel dengan penyiapan Anda:

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wp_test' );

/** MySQL database username */
define( 'DB_USER', 'wp_user' );

/** MySQL database password */
define( 'DB_PASSWORD', 'YourSecurePassword' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );

/** Database Charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The Database Collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

define('FORCE_SSL', true);

if ($_SERVER['HTTP_X_FORWARDED_PROTO'] == 'https') $_SERVER['HTTPS'] = 'on';
```

## Langkah 3: Mengkonfigurasi Apache HTTP Server

1. Di jendela terminal AWS Cloud9 IDE, pastikan bahwa Anda telah menginstal Apache:

```
httpd -v
```

Jika server Apache tidak diinstal, jalankan perintah berikut:

```
sudo yum install -y httpd
```

2. Arahkan ke direktori `/etc/httpd/conf.d`, yang merupakan lokasi untuk file konfigurasi host virtual Apache. Kemudian gunakan perintah `ln` untuk menautkan `wordpress.conf` yang Anda buat sebelumnya ke direktori kerja saat ini (`/etc/httpd/conf.d`):

```
cd /etc/httpd/conf.d
sudo ln -s /home/ec2-user/environment/config/wordpress.conf
```

3. Sekarang navigasikan ke direktori `/var/www`, yang merupakan folder admin default untuk server Apache. Dan gunakan perintah `ln` untuk menautkan direktori `wordpress` yang Anda buat sebelumnya ke direktori kerja saat ini (`/var/www`):

```
cd /var/www
sudo ln -s /home/ec2-user/environment/wordpress
```

4. Jalankan perintah chmod untuk mengizinkan server Apache menjalankan konten di subdirektori `wordpress`:

```
sudo chmod +x /home/ec2-user/
```

5. Sekarang mulai ulang server Apache agar ia dapat mendeteksi konfigurasi baru:

```
sudo service httpd restart
```

## Langkah 4: Mempratinjau konten WordPress web

1. Menggunakan AWS Cloud9 IDE, buat file baru yang disebut `index.html` dalam direktori berikut:`environment/wordpress`.
2. Tambahkan teks berformat HTML ke `index.html`. Sebagai contoh:

```
<h1>Hello World!</h1>
```

3. Di jendela Environment, pilih `index.html` file, lalu pilih Preview, Preview Running Application.

   Halaman web, yang menampilkan pesan Halo Dunia!, muncul di tab pratinjau aplikasi. Untuk melihat konten web di browser pilihan Anda, pilih Munculkan ke Jendela Baru.

   Jika Anda menghapus `index.html` file dan menyegarkan tab pratinjau aplikasi, halaman WordPress konfigurasi akan ditampilkan.

## Mengelola kesalahan konten campuran

Browser web menampilkan kesalahan konten campuran untuk WordPress situs jika memuat skrip HTTPS dan HTTP atau konten pada saat yang bersamaan. Kata-kata pesan kesalahan tergantung pada peramban web yang Anda gunakan, tetapi Anda diberi tahu bahwa koneksi Anda ke situs tidak aman atau tidak sepenuhnya aman. Dan peramban web Anda memblokir akses ke konten campuran.

> **Important**
>
> Secara default, semua halaman web yang Anda akses di tab pratinjau aplikasi AWS Cloud9 IDE secara otomatis menggunakan protokol HTTPS. Jika URI halaman memiliki fitur `http` protokol yang tidak aman, ia secara otomatis digantikan oleh `https`. Dan Anda tidak dapat mengakses konten yang tidak aman dengan mengubah `https` kembali ke `http` secara manual.
> Untuk panduan tentang penerapan HTTPS untuk situs web Anda, lihat WordPressdokumentasi.

# Tutorial Java untuk AWS Cloud9

> **Important**
>
> Jika Anda menggunakan lingkungan AWS Cloud9 pengembangan yang didukung oleh EC2 instance dengan 2 GiB atau lebih memori, kami sarankan Anda mengaktifkan dukungan Java yang disempurnakan. Ini menyediakan akses ke fitur produktivitas seperti penyelesaian kode, linting untuk kesalahan, tindakan spesifik konteks, dan opsi debugging seperti breakpoint dan stepping.
> Untuk informasi selengkapnya, lihat Peningkatan dukungan untuk pengembangan Java.

Tutorial ini memungkinkan Anda untuk menjalankan beberapa kode Java di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan
- Langkah 2: Menambahkan kode
- Langkah 3: Bangun dan jalankan kode
- Langkah 4: Siapkan untuk menggunakan AWS SDK untuk Java
- Langkah 5: Siapkan manajemen AWS kredensil di lingkungan Anda
- Langkah 6: Tambahkan kode AWS SDK
- Langkah 7: Bangun dan jalankan kode AWS SDK
- Langkah 8: Membersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

# Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda menginstal satu set alat pengembangan Java di lingkungan AWS Cloud9 pengembangan Anda. Jika Anda sudah memiliki seperangkat alat developer Java seperti Oracle JDK atau OpenJDK yang terpasang di lingkungan Anda, Anda dapat langsung beralih ke Langkah 2: Menambahkan kode. Sampel ini dikembangkan dengan OpenJDK 8, yang dapat Anda pasang di lingkungan Anda dengan menyelesaikan prosedur berikut.

1. Konfirmasi apakah OpenJDK 8 sudah terpasang. Untuk melakukan ini, dalam sesi terminal di AWS Cloud9 IDE, jalankan versi baris perintah dari runner Java dengan **`-version`**opsi. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.)

   ```
   java -version
   ```

   Berdasarkan output dari perintah sebelumnya, lakukan salah satu hal berikut:

   - Jika output menyatakan bahwa perintah `java` tidak ditemukan, lanjutkan dengan langkah 2 dalam prosedur ini untuk memasang OpenJDK 8.
   - Jika output berisi nilai yang dimulai dengan `Java(TM)`, `Java Runtime Environment`, `Java SE`, `J2SE`, atau `Java2`, OpenJDK tidak dipasang atau tidak diatur sebagai perangkat pengembangan Java default. Lanjutkan dengan langkah 2 dalam prosedur ini untuk memasang OpenJDK 8, dan kemudian beralih menggunakan OpenJDK 8.
   - Jika output berisi nilai yang dimulai dengan `java version 1.8` dan `OpenJDK`, lanjutkan ke Langkah 2: Menambahkan kode. OpenJDK 8 dipasang dengan benar untuk sampel ini.
   - Jika output berisi `java version` kurang dari `1.8` dan nilai dimulai dengan `OpenJDK`, lanjutkan dengan langkah 2 dalam prosedur ini untuk memutakhirkan versi OpenJDK yang dipasang ke OpenJDK 8.
2. Pastikan pembaruan keamanan terbaru dan perbaikan bug telah terpasang. Untuk melakukan ini, jalankan alat yum (untuk Amazon Linux) atau alat apt (untuk Server Ubuntu) dengan perintah **`update`** .

   Untuk Amazon Linux:

   ```
   sudo yum -y update
   ```

   Untuk Ubuntu Server:

```
sudo apt update
```

3. Pasang OpenJDK 8. Untuk melakukannya, jalankan alat yum (untuk Amazon Linux) atau alat apt (untuk Server Ubuntu) dengan perintah **`install`** , dengan menentukan paket OpenJDK 8.

   Untuk Amazon Linux:

   ```
   sudo yum -y install java-1.8.0-openjdk-devel
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt install -y openjdk-8-jdk
   ```

   Untuk informasi lebih lanjut, lihat Cara mengunduh dan memasang paket OpenJDK yang telah dibangun sebelumnya di situs web OpenJDK.

4. Ganti atau tingkatkan perangkat pengembangan Java default ke OpenJDK 8. Untuk melakukan ini, jalankan perintah **`update-alternatives`** dengan opsi **`--config`** . Jalankan perintah ini dua kali untuk beralih atau memutakhirkan versi baris perintah dari Java runner dan compiler.

   ```
   sudo update-alternatives --config java
   sudo update-alternatives --config javac
   ```

   Pada setiap prompt, ketik nomor pilihan untuk OpenJDK 8 (yang berisi `java-1.8`).

5. Konfirmasikan bahwa versi baris perintah Java runner dan compiler menggunakan OpenJDK 8. Untuk melakukannya, jalankan versi baris perintah dari Java runner dan compiler dengan opsi `-version`.

   ```
   java -version
   javac -version
   ```

   Jika OpenJDK 8 dipasang dan diatur dengan benar, output versi Java runner berisi nilai yang dimulai dengan `openjdk version 1.8`, dan output versi kompiler Java dimulai dengan nilai `javac 1.8`.

## Langkah 2: Menambahkan kode

Di AWS Cloud9 IDE, buat file dengan kode berikut, dan simpan file dengan namahello.java. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.)

```
public class hello {

  public static void main(String []args) {
    System.out.println("Hello, World!");

    System.out.println("The sum of 2 and 3 is 5.");

    int sum = Integer.parseInt(args[0]) + Integer.parseInt(args[1]);

    System.out.format("The sum of %s and %s is %s.\n",
      args[0], args[1], Integer.toString(sum));
  }
}
```

## Langkah 3: Bangun dan jalankan kode

1. Gunakan versi baris perintah dari compiler Java untuk mengkompilasi file hello.java ke dalam file hello.class. Untuk melakukan ini, menggunakan terminal di AWS Cloud9 IDE, dari direktori yang sama dengan hello.java file, jalankan kompiler Java, tentukan file. hello.java

   ```
   javac hello.java
   ```

2. Gunakan versi baris perintah dari Java runner untuk menjalankan file hello.class. Untuk melakukan ini, dari direktori yang sama dengan file hello.class, jalankan Java runner, tentukan nama kelas hello yang dideklarasikan dalam file hello.java, dengan dua bilangan bulat untuk ditambahkan (misalnya, 5 dan 9).

   ```
   java hello 5 9
   ```

3. Bandingkan output Anda.

   ```
   Hello, World!
   The sum of 2 and 3 is 5.
   The sum of 5 and 9 is 14.
   ```

# Langkah 4: Siapkan untuk menggunakan AWS SDK untuk Java

Anda dapat menyempurnakan sampel ini AWS SDK untuk Java untuk menggunakan bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

Pada langkah ini, Anda memasang Apache Maven atau Gradle di lingkungan Anda. Maven dan Gradle adalah sistem otomatisasi bangun yang umum yang dapat digunakan dengan proyek Java. Setelah Anda memasang Maven atau Gradle, Anda menggunakannya untuk menghasilkan proyek Java baru. Dalam proyek baru ini, Anda menambahkan referensi ke AWS SDK untuk Java. Ini AWS SDK untuk Java menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari kode Java Anda.

Topik


- Mengatur dengan Maven
- Mengatur dengan Gradle


## Mengatur dengan Maven

1. Pasang Maven di lingkungan Anda. Untuk melihat apakah Maven sudah diinstal, menggunakan terminal di AWS Cloud9 IDE, jalankan Maven dengan opsi. **`-version`**

   ```
   mvn -version
   ```

   Jika berhasil, output berisi nomor versi Maven. Jika Maven sudah terpasang, lanjutkan ke langkah 4 dalam prosedur ini untuk menggunakan Maven untuk menghasilkan proyek Java baru di lingkungan Anda.
2. Pasang Maven dengan menggunakan terminal untuk menjalankan perintah berikut.

   Untuk Amazon Linux, perintah berikut mendapatkan informasi tentang repositori paket tempat Maven disimpan, lalu menggunakan informasi ini untuk memasang Maven.

   ```
   sudo wget http://repos.fedorapeople.org/repos/dchen/apache-maven/epel-apache-
   maven.repo -O /etc/yum.repos.d/epel-apache-maven.repo
   sudo sed -i s/\$releasever/6/g /etc/yum.repos.d/epel-apache-maven.repo
   sudo yum install -y apache-maven
   ```

   Untuk informasi selengkapnya tentang perintah sebelumnya, lihat Paket Ekstra untuk Linux Korporasi (EPEL) di situs web Fedora Project Wiki.

Untuk Server Ubuntu, jalankan perintah berikut sebagai gantinya.

```
sudo apt install -y maven
```

3. Konfirmasi pemasangan dengan menjalankan Maven dengan opsi **`-version`** .

```
mvn -version
```

4. Gunakan Maven untuk menghasilkan proyek Java baru. Untuk melakukan ini, gunakan terminal untuk menjalankan perintah berikut dari direktori tempat Anda ingin Maven menghasilkan proyek (misalnya, direktori root lingkungan Anda).

```
mvn archetype:generate -DgroupId=com.mycompany.app -DartifactId=my-app -
DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
```

Perintah sebelumnya membuat struktur direktori berikut untuk proyek di lingkungan Anda.

```
my-app
  |- src
  |   `- main
  |        `- java
  |             `- com
  |                 `- mycompany
  |                      `- app
  |                          `-App.java
  |- test
  |   `- java
  |        `- com
  |            `- mycompany
  |                 `- app
  |                     `- AppTest.java
  `- pom.xml
```

Untuk informasi selengkapnya tentang struktur direktori sebelumnya, lihat Pola Dasar Cepat Maven dan Pengenalan Tata Letak Direktori Standar di situs web Proyek Maven Apache.

5. Ubah file Project Object Model (POM) untuk proyek tersebut. (File POM mendefinisikan pengaturan proyek Maven.) Untuk melakukan ini, di jendela Lingkungan, buka file `my-app/pom.xml`. Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `pom.xml`.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/
maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-assembly-plugin</artifactId>
        <version>3.6.0</version>
        <configuration>
          <descriptorRefs>
            <descriptorRef>jar-with-dependencies</descriptorRef>
          </descriptorRefs>
          <archive>
            <manifest>
              <mainClass>com.mycompany.app.App</mainClass>
            </manifest>
          </archive>
        </configuration>
        <executions>
          <execution>
            <phase>package</phase>
              <goals>
                <goal>single</goal>
              </goals>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
```

```
    <dependency>
      <groupId>com.amazonaws</groupId>
      <artifactId>aws-java-sdk</artifactId>
      <version>1.11.330</version>
    </dependency>
  </dependencies>
</project>
```

File POM sebelumnya menyertakan pengaturan proyek yang menentukan deklarasi seperti berikut ini:

- Pengaturan `artifactid` dari `my-app` menetapkan nama direktori root proyek, dan pengaturan `group-id` dari `com.mycompany.app` menetapkan struktur subdirektori `com/mycompany/app` dan deklarasi `package` dalam file `App.Java` dan `AppTest.java`.
- Pengaturan `artifactId` dari `my-app`, dengan pengaturan `packaging` dari `jar`, pengaturan `version` dari `1.0-SNAPSHOT`, dan pengaturan `descriptorRef` dari `jar-with-dependencies` mengatur nama file JAR output dari `my-app-1.0-SNAPSHOT-jar-with-dependencies.jar`.
- Bagian `plugin` menyatakan bahwa JAR tunggal, yang mencakup semua dependensi, akan dibuat.
- `dependency`Bagian dengan `groupId` pengaturan `com.amazon.aws` dan `artifactId` pengaturan `aws-java-sdk` termasuk file AWS SDK untuk Java perpustakaan. Versi AWS SDK untuk Java yang akan digunakan dideklarasikan oleh pengaturan `version`. Untuk menggunakan versi yang berbeda, ganti nomor versi ini.

Lewati ke Langkah 5: Siapkan manajemen AWS kredensil di lingkungan Anda.

## Mengatur dengan Gradle

1. Pasang Gradle di lingkungan Anda. Untuk melihat apakah Gradle sudah diinstal, menggunakan terminal di AWS Cloud9 IDE, jalankan Gradle dengan opsi. **`-version`**

   ```
   gradle -version
   ```

   Jika berhasil, output berisi nomor versi Gradle. Jika Gradle sudah terpasang, lanjutkan ke langkah 4 dalam prosedur ini untuk menggunakan Gradle guna menghasilkan proyek Java baru di lingkungan Anda.

2. Pasang Gradle dengan menggunakan terminal untuk menjalankan perintah berikut. Perintah ini memasang dan menjalankan alat SDKMAN!, dan kemudian menggunakan SDKMAN! untuk memasang versi terbaru Gradle.

```
curl -s "https://get.sdkman.io" | bash
source "$HOME/.sdkman/bin/sdkman-init.sh"
sdk install gradle
```

   Untuk informasi lebih lanjut tentang perintah sebelumnya, lihat Pemasangan di situs web SDKMAN! dan Pasang dengan pengelola paket di situs web Gradle.

3. Konfirmasi pemasangan dengan menjalankan Gradle dengan opsi **`-version`** .

```
gradle -version
```

4. Gunakan Gradle untuk membuat proyek Java baru di lingkungan Anda. Untuk melakukannya, gunakan terminal untuk menjalankan perintah berikut untuk membuat direktori proyek, dan kemudian beralih ke direktori itu.

```
mkdir my-app
cd my-app
```

5. Jalankan perintah berikut agar Gradle membuat proyek aplikasi Java baru di direktori `my-app` di lingkungan Anda.

```
gradle init --type java-application
```

   Perintah sebelumnya membuat struktur direktori berikut untuk proyek di lingkungan Anda.

```
my-app
  |- .gradle
  |   `- (various supporting project folders and files)
  |- gradle
  |   `- (various supporting project folders and files)
  |- src
  |   |- main
  |   |    `- java
  |   |         `- App.java
  |   `- test
  |        `- java
  |             `- AppTest.java
```

```
  |- build.gradle
  |- gradlew
  |- gradlew.bat
  `- settings.gradle
```

6. Ubah `AppTest.java` untuk proyek. (Jika Anda tidak melakukan ini, proyek mungkin tidak dibangun atau berjalan seperti yang diharapkan). Untuk melakukan ini, di jendela Lingkungan, buka file `my-app/src/test/java/AppTest.java`. Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `AppTest.java`.

```
import org.junit.Test;
import static org.junit.Assert.*;

public class AppTest {
  @Test public void testAppExists () {
    try {
      Class.forName("com.mycompany.app.App");
    } catch (ClassNotFoundException e) {
      fail("Should have a class named App.");
    }
  }
}
```

7. Ubah file `build.gradle` untuk proyek. (File `build.gradle` mendefinisikan pengaturan proyek Gradle.) Untuk melakukan ini, di jendela Lingkungan, buka file `my-app/build.gradle`. Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `build.gradle`.

```
apply plugin: 'java'
apply plugin: 'application'

repositories {
  jcenter()
  mavenCentral()
}

buildscript {
  repositories {
    mavenCentral()
  }
  dependencies {
    classpath "io.spring.gradle:dependency-management-plugin:1.0.3.RELEASE"
  }
```

```
}

apply plugin: "io.spring.dependency-management"

dependencyManagement {
  imports {
    mavenBom 'com.amazonaws:aws-java-sdk-bom:1.11.330'
  }
}

dependencies {
  compile 'com.amazonaws:aws-java-sdk-s3'
  testCompile group: 'junit', name: 'junit', version: '4.12'
}

run {
  if (project.hasProperty("appArgs")) {
    args Eval.me(appArgs)
  }
}

mainClassName = 'App'
```

File `build.gradle` sebelumnya menyertakan pengaturan proyek yang menentukan deklarasi seperti berikut ini:

- `io.spring.dependency-management`Plugin ini digunakan untuk mengimpor AWS SDK untuk Java Maven Bill of Materials (BOM) untuk mengelola AWS SDK untuk Java dependensi proyek. `classpath`mendeklarasikan versi yang akan digunakan. Untuk menggunakan versi yang berbeda, ganti nomor versi ini.
- `com.amazonaws:aws-java-sdk-s3` menyertakan bagian Amazon S3 dari file pustaka AWS SDK untuk Java . `mavenBom` mendeklarasikan versi yang akan digunakan. Jika Anda ingin menggunakan versi lain, ganti nomor versi ini.

## Langkah 5: Siapkan manajemen AWS kredensil di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK untuk Java untuk memanggil AWS layanan, Anda harus memberikan satu set AWS kredensil dengan panggilan tersebut. Kredensil ini menentukan apakah AWS SDK untuk Java memiliki izin yang sesuai untuk melakukan panggilan itu. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat Mengatur Kredensial dan Wilayah AWS untuk Pengembangan di Panduan Developer AWS SDK untuk Java .

## Langkah 6: Tambahkan kode AWS SDK

Pada langkah ini, Anda menambahkan kode untuk berinteraksi dengan Amazon S3 untuk membuat bucket, membuat daftar bucket Anda yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

Dari jendela Lingkungan, buka file `my-app/src/main/java/com/mycompany/app/App.java` untuk Maven atau file `my-app/src/main/java/App.java` untuk Gradle. Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `App.java`.

```
package com.mycompany.app;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.AmazonS3Exception;
import com.amazonaws.services.s3.model.Bucket;
import com.amazonaws.services.s3.model.CreateBucketRequest;

import java.util.List;

public class App {

    private static AmazonS3 s3;

    public static void main(String[] args) {
        if (args.length < 2) {
            System.out.format("Usage: <the bucket name> <the AWS Region to use>\n" +
                    "Example: my-test-bucket us-east-2\n");
            return;
        }

        String bucket_name = args[0];
        String region = args[1];

        s3 = AmazonS3ClientBuilder.standard()
                .withCredentials(new ProfileCredentialsProvider())
```

```
                .withRegion(region)
                .build();

        // List current buckets.
        ListMyBuckets();

        // Create the bucket.
        if (s3.doesBucketExistV2(bucket_name)) {
            System.out.format("\nCannot create the bucket. \n" +
                    "A bucket named '%s' already exists.", bucket_name);
            return;
        } else {
            try {
                System.out.format("\nCreating a new bucket named '%s'...\n\n",
bucket_name);
                s3.createBucket(new CreateBucketRequest(bucket_name, region));
            } catch (AmazonS3Exception e) {
                System.err.println(e.getErrorMessage());
            }
        }

        // Confirm that the bucket was created.
        ListMyBuckets();

        // Delete the bucket.
        try {
            System.out.format("\nDeleting the bucket named '%s'...\n\n", bucket_name);
            s3.deleteBucket(bucket_name);
        } catch (AmazonS3Exception e) {
            System.err.println(e.getErrorMessage());
        }

        // Confirm that the bucket was deleted.
        ListMyBuckets();

    }

    private static void ListMyBuckets() {
        List<Bucket> buckets = s3.listBuckets();
        System.out.println("My buckets now are:");

        for (Bucket b : buckets) {
            System.out.println(b.getName());
        }
```

```
    }

}
```

## Langkah 7: Bangun dan jalankan kode AWS SDK

Untuk menjalankan kode dari langkah sebelumnya, jalankan perintah berikut dari terminal. Perintah ini menggunakan Maven atau Gradle untuk membuat file JAR yang dapat dieksekusi untuk proyek tersebut, lalu menggunakan runner Java untuk menjalankan JAR. JAR berjalan dengan nama bucket untuk dibuat di Amazon S3 (misalnya,`my-test-bucket`) dan ID AWS Wilayah untuk membuat bucket sebagai input (misalnya,`us-east-2`).

Untuk Maven, jalankan perintah berikut.

```
cd my-app
mvn package
java -cp target/my-app-1.0-SNAPSHOT-jar-with-dependencies.jar com.mycompany.app.App my-
test-bucket us-east-2
```

Untuk Gradle, jalankan perintah berikut.

```
gradle build
gradle run -PappArgs="['my-test-bucket', 'us-east-2']"
```

Bandingkan hasil Anda dengan output berikut.

```
My buckets now are:

Creating a new bucket named 'my-test-bucket'...

My buckets now are:

my-test-bucket

Deleting the bucket named 'my-test-bucket'...

My buckets now are:
```

## Langkah 8: Membersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial C++ untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menjalankan kode C ++ di lingkungan AWS Cloud9 pengembangan. Kode ini juga menggunakan sumber daya yang disediakan oleh AWS SDK untuk C++, pustaka sumber terbuka lintas platform termodulasi yang dapat Anda gunakan untuk terhubung ke Amazon Web Services.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang g++ dan paket dev yang diperlukan
- Langkah 2: Instal CMake
- Langkah 3: Dapatkan dan bangun SDK for C++
- Langkah 4: Buat file C ++ dan CMake Daftar
- Langkah 5: Membangun dan menjalankan kode C++
- Langkah 6: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.

- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang g++ dan paket dev yang diperlukan

Untuk membangun dan menjalankan aplikasi C++, Anda memerlukan utilitas seperti `g++`, yang merupakan kompiler C++ yang disediakan oleh GNU Complier Collection (GCC).

Anda juga perlu menambahkan file header (paket `-dev`) untuk `libcurl`, `libopenssl`, `libuuid`, `zlib` dan, secara opsional, `libpulse` untuk dukungan Amazon Polly.

Proses memasang alat developer sedikit berbeda tergantung pada apakah Anda menggunakan instans Amazon Linux/Amazon Linux 2 atau instans Ubuntu.

Amazon Linux-based systems

Anda dapat memeriksa apakah Anda sudah `gcc` menginstal dengan menjalankan perintah berikut di AWS Cloud9 terminal:

```
g++ --version
```

Jika `g++` tidak terpasang, Anda dapat dengan mudah memasangnya bagian dari grup paket yang disebut "Alat Developer". Alat-alat ini ditambahkan ke sebuah instans dengan perintah `yum groupinstall`:

```
sudo yum groupinstall "Development Tools"
```

Jalankan `g++ --version` lagi untuk mengonfirmasi bahwa kompiler telah terpasang.

Sekarang pasang paket untuk perpustakaan yang diperlukan menggunakan pengelola paket sistem Anda:

```
sudo yum install libcurl-devel openssl-devel libuuid-devel pulseaudio-libs-devel
```

Ubuntu-based systems

Anda dapat memeriksa apakah Anda sudah `gcc` menginstal dengan menjalankan perintah berikut di AWS Cloud9 terminal:

```
g++ --version
```

Jika gcc tidak terpasang, Anda dapat memasangnya di sistem berbasis Ubuntu dengan menjalankan perintah berikut:

```
sudo apt update
sudo apt install build-essential
sudo apt-get install manpages-dev
```

Jalankan `g++ --version` lagi untuk mengonfirmasi bahwa kompiler telah terpasang.

Sekarang pasang paket untuk perpustakaan yang diperlukan menggunakan pengelola paket sistem Anda:

```
sudo apt-get install libcurl4-openssl-dev libssl-dev uuid-dev zlib1g-dev libpulse-
dev
```

## Langkah 2: Instal CMake

Anda perlu memasang alat `cmake`, yang mengotomatisasi proses pembangunan file yang dapat dieksekusi dari kode sumber.

1. Di jendela terminal IDE, jalankan perintah berikut untuk mendapatkan arsip yang diperlukan:

   ```
   wget https://cmake.org/files/v3.18/cmake-3.18.0.tar.gz
   ```

2. Ekstrak file dari arsip dan arahkan ke direktori yang berisi file yang belum dibongkar:

   ```
   tar xzf cmake-3.18.0.tar.gz
   cd cmake-3.18.0
   ```

3. Selanjutnya, jalankan skrip bootstrap dan pasang `cmake` dengan menjalankan perintah berikut:

   ```
   ./bootstrap
   make
   sudo make install
   ```

4. Konfirmasikan bahwa Anda telah memasang alat dengan menjalankan perintah berikut:

```
cmake --version
```

## Langkah 3: Dapatkan dan bangun SDK for C++

Untuk menyiapkan AWS SDK for C++, Anda dapat membuat SDK sendiri langsung dari sumbernya atau mengunduh pustaka menggunakan pengelola paket. Anda dapat menemukan detail tentang opsi yang tersedia di Memulai Menggunakan AWS SDK for C++ AWS SDK untuk C++ di Panduan Pengembang.

Sampel ini menunjukkan penggunaan `git` untuk mengkloning kode sumber SDK dan `cmake` untuk membangun SDK for C++.

1. Kloning repositori jarak jauh dan dapatkan semua submodul git secara rekursif untuk AWS Cloud9 lingkungan Anda dengan menjalankan perintah berikut di terminal:

   ```
   git clone --recurse-submodules https://github.com/aws/aws-sdk-cpp
   ```

2. Arahkan ke `aws-sdk-cpp` direktori baru, buat sub-direktori untuk membangun AWS SDK for C++, lalu arahkan ke direktori tersebut:

   ```
   cd aws-sdk-cpp
   mkdir sdk_build
   cd sdk_build
   ```

3. > **Note**
   >
   > Untuk menghemat waktu, langkah ini hanya membangun bagian Amazon S3 dari AWS SDK untuk C++. Jika Anda ingin membangun SDK lengkap, hilangkan `-DBUILD_ONLY=s3` dari perintah `cmake`.
   > Membangun SDK for C++ lengkap dapat memakan waktu lebih dari satu jam untuk diselesaikan, tergantung pada sumber daya komputasi yang tersedia untuk instans EC2 Amazon atau server Anda sendiri.

   Gunakan `cmake` untuk membangun bagian Amazon S3 dari SDK for C++ ke dalam direktori `sdk_build` dengan menjalankan perintah berikut:

```
cmake .. -DBUILD_ONLY=s3
```

4. Sekarang jalankan perintah `make install` sehingga SDK yang dibangun dapat diakses:

```
sudo make install
cd ..
```

## Langkah 4: Buat file C ++ dan CMake Daftar

Pada langkah ini, Anda membuat file C++ yang memungkinkan pengguna proyek berinteraksi dengan bucket Amazon S3.

Anda juga membuat file `CMakeLists.txt` yang menyediakan instruksi yang digunakan oleh `cmake` untuk membangun pustaka C++ Anda.

1. Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan nama `s3-demo.cpp` di root (`/`) lingkungan Anda.

```
#include <iostream>
#include <aws/core/Aws.h>
#include <aws/s3/S3Client.h>
#include <aws/s3/model/Bucket.h>
#include <aws/s3/model/CreateBucketConfiguration.h>
#include <aws/s3/model/CreateBucketRequest.h>
#include <aws/s3/model/DeleteBucketRequest.h>

// Look for a bucket among all currently available Amazon S3 buckets.
bool FindTheBucket(const Aws::S3::S3Client &s3Client,
                   const Aws::String &bucketName) {

    Aws::S3::Model::ListBucketsOutcome outcome = s3Client.ListBuckets();

    if (outcome.IsSuccess()) {

        std::cout << "Looking for a bucket named '" << bucketName << "'..."
                  << std::endl << std::endl;

        Aws::Vector<Aws::S3::Model::Bucket> bucket_list =
                outcome.GetResult().GetBuckets();
```

```
        for (Aws::S3::Model::Bucket const &bucket: bucket_list) {
            if (bucket.GetName() == bucketName) {
                std::cout << "Found the bucket." << std::endl << std::endl;

                return true;
            }
        }

        std::cout << "Could not find the bucket." << std::endl << std::endl;
    } else {
        std::cerr << "listBuckets error: "
                  << outcome.GetError().GetMessage() << std::endl;
    }

    return outcome.IsSuccess();
}

// Create an Amazon S3 bucket.
bool CreateTheBucket(const Aws::S3::S3Client &s3Client,
                     const Aws::String &bucketName,
                     const Aws::String &region) {

    std::cout << "Creating a bucket named '"
              << bucketName << "'..." << std::endl << std::endl;

    Aws::S3::Model::CreateBucketRequest request;
    request.SetBucket(bucketName);

    if (region != "us-east-1") {
        Aws::S3::Model::CreateBucketConfiguration createBucketConfig;
        createBucketConfig.SetLocationConstraint(

 Aws::S3::Model::BucketLocationConstraintMapper::GetBucketLocationConstraintForName(
                        region));
        request.SetCreateBucketConfiguration(createBucketConfig);
    }

    Aws::S3::Model::CreateBucketOutcome outcome =
            s3Client.CreateBucket(request);

    if (outcome.IsSuccess()) {
        std::cout << "Bucket created." << std::endl << std::endl;
    } else {
        std::cerr << "createBucket error: "
```

```
                << outcome.GetError().GetMessage() << std::endl;
    }

    return outcome.IsSuccess();
}

// Delete an existing Amazon S3 bucket.
bool DeleteTheBucket(const Aws::S3::S3Client &s3Client,
                     const Aws::String &bucketName) {

    std::cout << "Deleting the bucket named '"
              << bucketName << "'..." << std::endl << std::endl;

    Aws::S3::Model::DeleteBucketRequest request;
    request.SetBucket(bucketName);

    Aws::S3::Model::DeleteBucketOutcome outcome =
            s3Client.DeleteBucket(request);

    if (outcome.IsSuccess()) {
        std::cout << "Bucket deleted." << std::endl << std::endl;
    } else {
        std::cerr << "deleteBucket error: "
                  << outcome.GetError().GetMessage() << std::endl;
    }

    return outcome.IsSuccess();
}

#ifndef EXCLUDE_MAIN_FUNCTION
// Create an S3 bucket and then delete it.
// Before and after creating the bucket, and again after deleting the bucket,
// try to determine whether that bucket still exists.
int main(int argc, char *argv[]) {

    if (argc < 3) {
        std::cout << "Usage: s3-demo <bucket name> <AWS Region>" << std::endl
                  << "Example: s3-demo my-bucket us-east-1" << std::endl;
        return 1;
    }

    Aws::SDKOptions options;
    Aws::InitAPI(options);
    {
```

```
        Aws::String bucketName = argv[1];
        Aws::String region = argv[2];

        Aws::Client::ClientConfiguration config;

        config.region = region;

        Aws::S3::S3Client s3Client(config);

        if (!FindTheBucket(s3Client, bucketName)) {
            return 1;
        }

        if (!CreateTheBucket(s3Client, bucketName, region)) {
            return 1;
        }

        if (!FindTheBucket(s3Client, bucketName)) {
            return 1;
        }

        if (!DeleteTheBucket(s3Client, bucketName)) {
            return 1;
        }

        if (!FindTheBucket(s3Client, bucketName)) {
            return 1;
        }
    }
    Aws::ShutdownAPI(options);

    return 0;
}
#endif  // EXCLUDE_MAIN_FUNCTION
```

2. Buat file kedua dengan konten ini, dan simpan file dengan nama `CMakeLists.txt` di root (`/`) lingkungan Anda. File ini memungkinkan Anda untuk membangun kode Anda menjadi file yang dapat dieksekusi.

```
# A minimal CMakeLists.txt file for the AWS SDK for C++.

# The minimum version of CMake that will work.
cmake_minimum_required(VERSION 2.8)
```

```
# The project name.
project(s3-demo)

# Locate the AWS SDK for C++ package.
set(AWSSDK_ROOT_DIR, "/usr/local/")
set(BUILD_SHARED_LIBS ON)
find_package(AWSSDK REQUIRED COMPONENTS s3)

# The executable name and its source files.
add_executable(s3-demo s3-demo.cpp)

# The libraries used by your executable.
target_link_libraries(s3-demo ${AWSSDK_LINK_LIBRARIES})
```

## Langkah 5: Membangun dan menjalankan kode C++

1. Di direktori root lingkungan Anda tempat Anda menyimpan `s3-demo.cpp` dan `CMakeLists.txt`, jalankan `cmake` untuk membangun proyek Anda:

   ```
   cmake .
   make
   ```

2. Anda sekarang dapat menjalankan program Anda dari baris perintah. Dalam perintah berikut, ganti `my-unique-bucket-name` dengan nama unik untuk bucket Amazon S3 dan, jika perlu, ganti `us-east-1` dengan pengenal Wilayah AWS lain tempat Anda ingin membuat bucket.

   ```
   ./s3-demo my-unique-bucket-name us-east-1
   ```

   Jika program berhasil dijalankan, output yang mirip dengan berikut akan dikembalikan:

   ```
   Looking for a bucket named 'my-unique-bucket-name'...

   Could not find the bucket.

   Creating a bucket named 'my-unique-bucket-name'...

   Bucket created.

   Looking for a bucket named 'my-unique-bucket-name'...
   ```

```
Found the bucket.

Deleting the bucket named 'my-unique-bucket-name'...

Bucket deleted.

Looking for a bucket named 'my-unique-bucket-name'...

Could not find the bucket.
```

## Langkah 6: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah Anda selesai dengan sampel ini, hapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial Python untuk AWS Cloud9

Tutorial ini menunjukkan cara menjalankan kode Python di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon Elastic Compute Cloud (Amazon EC2) dan Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang Python
- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Instal dan konfigurasikan AWS SDK untuk Python (Boto3)
- Langkah 5: Tambahkan kode AWS SDK
- Langkah 6: Jalankan kode AWS SDK
- Langkah 7: Bersihkan

## Prasyarat

Sebelum Anda memulai tutorial ini, pastikan untuk memenuhi persyaratan berikut.

- Anda memiliki lingkungan AWS Cloud9 EC2 pengembangan

  Tutorial ini mengasumsikan bahwa Anda memiliki EC2 lingkungan, dan lingkungan terhubung ke EC2 instance Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Lihat Menciptakan EC2 Lingkungan untuk detail.

  Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan instruksi tutorial ini.
- Anda telah membuka AWS Cloud9 IDE untuk lingkungan itu

  Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Lihat Membuka lingkungan di AWS Cloud9 untuk detail.

## Langkah 1: Pasang Python

1. Dalam sesi terminal di AWS Cloud9 IDE, konfirmasikan apakah Python sudah diinstal dengan menjalankan perintah. **`python --version`** (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika Python sudah terpasang, lewati ke Langkah 2: Menambahkan kode.
2. Jalankan perintah **`yum update`** untuk (Amazon Linux) atau **`apt update`** untuk (Ubuntu Server) untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug terpasang.

   Untuk Amazon Linux:

   ```
   sudo yum -y update
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt update
   ```
3. Pasang Python dengan menjalankan perintah **`install`** .

   Untuk Amazon Linux:

```
sudo yum -y install python3
```

Untuk Ubuntu Server:

```
sudo apt-get install python3
```

## Langkah 2: Menambahkan kode

Di AWS Cloud9 IDE, buat file dengan konten berikut dan simpan file dengan namanya`hello.py`. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.)

```
import sys

print('Hello, World!')

print('The sum of 2 and 3 is 5.')

sum = int(sys.argv[1]) + int(sys.argv[2])

print('The sum of {0} and {1} is {2}.'.format(sys.argv[1], sys.argv[2], sum))
```

## Langkah 3: Jalankan kode

1. Di AWS Cloud9 IDE, pada bilah menu pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Pada tab [Baru] - Berhenti, masukkan `hello.py 5 9` untuk Perintah. Dalam kode, 5 mewakili `sys.argv[1]`, dan 9 mewakili `sys.argv[2]`.
3. Pilih Jalankan, dan bandingkan output Anda.

   ```
   Hello, World!
   The sum of 2 and 3 is 5.
   The sum of 5 and 9 is 14.
   ```

4. Secara default, AWS Cloud9 secara otomatis memilih runner untuk kode Anda. Untuk mengubah runner, pilih Runner, lalu pilih Python 2 atau Python 3.

> **Note**
>
> Anda dapat membuat runner kustom untuk versi tertentu dari Python. Untuk detailnya, lihat Buat Builder atau Runner.

## Langkah 4: Instal dan konfigurasikan AWS SDK untuk Python (Boto3)

AWS SDK untuk Python (Boto3) Ini memungkinkan Anda untuk menggunakan kode Python untuk berinteraksi dengan AWS layanan seperti Amazon S3. Misalnya, Anda dapat menggunakan SDK untuk membuat bucket Amazon S3, membuat daftar bucket Anda yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

### Memasang PIP

Di AWS Cloud9 IDE, konfirmasikan `pip` apakah sudah diinstal untuk versi aktif Python dengan menjalankan perintah. **`python -m pip --version`** Jika `pip` terpasang, lewati ke bagian berikutnya.

Untuk memasang `pip`, jalankan perintah berikut. Karena sudo berada di lingkungan yang berbeda dari pengguna Anda, Anda harus menentukan versi Python yang akan digunakan jika berbeda dari versi alias saat ini.

```
curl -O https://bootstrap.pypa.io/get-pip.py # Get the install script.
sudo python3 get-pip.py                      # Install pip for Python 3.
python -m pip --version                       # Verify pip is installed.
rm get-pip.py                                 # Delete the install script.
```

Untuk informasi selengkapnya, lihat Pemasangan di situs web `pip`.

### Instal AWS SDK untuk Python (Boto3)

Setelah Anda menginstal`pip`, instal AWS SDK untuk Python (Boto3) dengan menjalankan **`pip install`**perintah.

```
sudo python3 -m pip install boto3  # Install boto3 for Python 3.
python -m pip show boto3            # Verify boto3 is installed for the current version
 of Python.
```

Untuk informasi selengkapnya, lihat bagian “Instalasi” dari Mulai Cepat di. AWS SDK untuk Python (Boto3)

### Mengatur kredensial di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK untuk Python (Boto3) untuk memanggil AWS layanan, Anda harus memberikan satu set kredensional dengan panggilan tersebut. Kredensial ini menentukan apakah SDK memiliki izin yang diperlukan untuk melakukan panggilan. Jika kredensial tidak mencakup izin yang diperlukan, panggilan gagal.

Untuk menyimpan kredensial Anda di dalam lingkungan, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi selengkapnya, lihat Kredensial di AWS SDK untuk Python (Boto3).

## Langkah 5: Tambahkan kode AWS SDK

Tambahkan kode yang menggunakan Amazon S3 untuk membuat bucket, mencantumkan bucket Anda yang tersedia, dan secara opsional menghapus bucket yang baru saja Anda buat.

Di AWS Cloud9 IDE, buat file dengan konten berikut dan simpan file dengan namanya `s3.py`.

```
import sys
import boto3
from botocore.exceptions import ClientError


def list_my_buckets(s3_resource):
    print("Buckets:\n\t", *[b.name for b in s3_resource.buckets.all()], sep="\n\t")


def create_and_delete_my_bucket(s3_resource, bucket_name, keep_bucket):
    list_my_buckets(s3_resource)

    try:
        print("\nCreating new bucket:", bucket_name)
        bucket = s3_resource.create_bucket(
            Bucket=bucket_name,
            CreateBucketConfiguration={
                "LocationConstraint": s3_resource.meta.client.meta.region_name
            },
        )
    except ClientError as e:
```

```
        print(
            f"Couldn't create a bucket for the demo. Here's why: "
            f"{e.response['Error']['Message']}"
        )
        raise

    bucket.wait_until_exists()
    list_my_buckets(s3_resource)

    if not keep_bucket:
        print("\nDeleting bucket:", bucket.name)
        bucket.delete()

        bucket.wait_until_not_exists()
        list_my_buckets(s3_resource)
    else:
        print("\nKeeping bucket:", bucket.name)


def main():
    import argparse

    parser = argparse.ArgumentParser()
    parser.add_argument("bucket_name", help="The name of the bucket to create.")
    parser.add_argument("region", help="The region in which to create your bucket.")
    parser.add_argument(
        "--keep_bucket",
        help="Keeps the created bucket. When not "
        "specified, the bucket is deleted "
        "at the end of the demo.",
        action="store_true",
    )

    args = parser.parse_args()
    s3_resource = (
        boto3.resource("s3", region_name=args.region)
        if args.region
        else boto3.resource("s3")
    )
    try:
        create_and_delete_my_bucket(s3_resource, args.bucket_name, args.keep_bucket)
    except ClientError:
        print("Exiting the demo.")
```

```
if __name__ == "__main__":
    main()
```

## Langkah 6: Jalankan kode AWS SDK

1. Pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.
2. Untuk Commands`3.py my-test-bucket us-west-2`, enter, di mana `my-test-bucket` nama bucket yang akan dibuat, dan `us-west-2` merupakan ID AWS Wilayah tempat bucket Anda dibuat. Secara default, bucket Anda dihapus sebelum skrip keluar. Untuk menyimpan bucket Anda, tambahkan `--keep_bucket` ke perintah Anda. Untuk daftar AWS Wilayah IDs, lihat Titik Akhir dan Kuota Layanan Penyimpanan Sederhana Amazon di. Referensi Umum AWS

   > **Note**
   >
   > Nama bucket Amazon S3 harus unik di seluruh AWS—bukan hanya akun Anda. AWS

3. Pilih Jalankan, dan bandingkan output Anda.

```
Buckets:

        a-pre-existing-bucket

Creating new bucket: my-test-bucket
Buckets:

        a-pre-existing-bucket
        my-test-bucket

Deleting bucket: my-test-bucket
Buckets:

        a-pre-existing-bucket
```

## Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah Anda selesai dengan tutorial ini, hapus AWS Cloud9 lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial.NET untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menjalankan beberapa kode .NET di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan
- Langkah 2 (Opsional): Pasang ekstensi .NET CLI untuk fungsi Lambda
- Langkah 3: Buat proyek aplikasi konsol.NET
- Langkah 4: Tambahkan kode
- Langkah 5: Bangun dan jalankan kode
- Langkah 6: Buat dan siapkan proyek aplikasi konsol.NET yang menggunakan AWS SDK untuk .NET
- Langkah 7: Tambahkan kode AWS SDK
- Langkah 8: Bangun dan jalankan kode AWS SDK
- Langkah 9: Membersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis

lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.

- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda menginstal .NET SDK ke lingkungan Anda, yang diperlukan untuk menjalankan sampel ini.

1. Konfirmasikan apakah versi terbaru dari .NET SDK sudah diinstal di lingkungan Anda. Untuk melakukan ini, dalam sesi terminal di AWS Cloud9 IDE, jalankan antarmuka baris perintah .NET Core (CLI) dengan opsi. **`--version`**

   ```
   dotnet --version
   ```

   Jika versi Alat Baris Perintah .NET ditampilkan, dan versinya 2.0 atau lebih tinggi, langsung lanjutkan ke Langkah 3: Buat proyek aplikasi konsol.NET. Jika versi kurang dari 2.0, atau jika kesalahan seperti `bash: dotnet: command not found` ditampilkan, lanjutkan untuk menginstal .NET SDK.
2. Untuk Amazon Linux, dalam sesi terminal di AWS Cloud9 IDE, jalankan perintah berikut untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug diinstal, dan untuk menginstal `libunwind` paket yang dibutuhkan .NET SDK. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.)

   ```
   sudo yum -y update
   sudo yum -y install libunwind
   ```

   Untuk Ubuntu Server, dalam sesi terminal di AWS Cloud9 IDE, jalankan perintah berikut untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug diinstal. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.)

   ```
   sudo apt -y update
   ```
3. Unduh skrip installer.NET SDK ke lingkungan Anda dengan menjalankan perintah berikut.

```
wget https://dot.net/v1/dotnet-install.sh
```

4. Jadikan skrip pemasangan dapat dieksekusi oleh pengguna saat ini dengan menjalankan perintah berikut.

```
sudo chmod u=rx dotnet-install.sh
```

5. Jalankan skrip penginstal, yang mengunduh dan menginstal .NET SDK, dengan menjalankan perintah berikut.

```
./dotnet-install.sh -c Current
```

6. Tambahkan .NET SDK ke file AndaPATH. Untuk melakukannya, di profil shell untuk lingkungan (misalnya, file `.bashrc`), tambahkan subdirektori `$HOME/.dotnet` ke variabel `PATH` untuk lingkungan, sebagai berikut.

   a. Buka file `.bashrc` untuk mengedit dengan menggunakan perintah **`vi`** .

   ```
   vi ~/.bashrc
   ```

   b. Untuk Amazon Linux, menggunakan tombol panah bawah atau `j`, pindah ke baris yang dimulai dengan `export PATH`.

   Untuk Server Ubuntu, pindah ke baris terakhir file dengan mengetik `G`.

   c. Menggunakan tombol panah kanan atau `$`, pindah ke akhir baris itu.

   d. Beralih ke mode penyisipan dengan menekan tombol `i`. (`-- INSERT ---` akan muncul di akhir tampilan.)

   e. Untuk Amazon Linux, tambahkan subdirektori `$HOME/.dotnet` ke variabel **`PATH`** dengan mengetik `:$HOME/.dotnet`. Pastikan untuk menyertakan karakter titik dua (`:`). Baris sekarang akan terlihat seperti berikut.

   ```
   export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet
   ```

   Untuk Server Ubuntu, tekan tombol panah kanan lalu tekan `Enter` dua kali, diikuti dengan mengetik baris berikut di akhir file sendiri.

   ```
   export PATH=$HOME/.dotnet:$PATH
   ```

f. Simpan file tersebut. Untuk melakukannya, tekan tombol `Esc` (`-- INSERT ---` akan menghilang dari akhir tampilan), ketik `:wq` (untuk menulis ke dan kemudian keluar dari file), lalu tekan `Enter`.

7. Muat .NET SDK dengan sumber `.bashrc` file.

```
. ~/.bashrc
```

8. Konfirmasikan .NET SDK dimuat dengan menjalankan .NET CLI dengan **`--help`**opsi.

```
dotnet --help
```

Jika berhasil, nomor versi.NET SDK ditampilkan, dengan informasi penggunaan tambahan.

9. Jika Anda tidak lagi ingin menyimpan skrip installer.NET SDK di lingkungan Anda, Anda dapat menghapusnya sebagai berikut.

```
rm dotnet-install.sh
```

## Langkah 2 (Opsional): Pasang ekstensi .NET CLI untuk fungsi Lambda

Meskipun tidak diperlukan untuk tutorial ini, Anda dapat menyebarkan AWS Lambda fungsi dan AWS Serverless Application Model aplikasi menggunakan CLI .NET jika Anda juga menginstal `Amazon.Lambda.Tools` paket.

1. Untuk memasang paket, jalankan perintah berikut:

```
dotnet tool install -g Amazon.Lambda.Tools
```

2. Sekarang atur variabel lingkungan `PATH` dan `DOTNET_ROOT` untuk menunjuk ke alat Lambda yang terpasang. Di file `.bashrc`, temukan bagian `export PATH` dan edit sehingga tampak seperti berikut ini (lihat Langkah 1 untuk detail tentang mengedit file ini):

```
export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet:$HOME/.dotnet/tools
export DOTNET_ROOT=$HOME/.dotnet
```

## Langkah 3: Buat proyek aplikasi konsol.NET

Pada langkah ini, Anda menggunakan .NET untuk membuat proyek bernama`hello`. Proyek ini berisi semua file yang dibutuhkan .NET untuk menjalankan aplikasi sederhana dari terminal di IDE. Kode aplikasi ditulis dalam C#.

Buat proyek aplikasi konsol.NET. Untuk melakukan ini, jalankan CLI .NET dengan **`new`**perintah, tentukan jenis template proyek aplikasi konsol dan bahasa pemrograman yang akan digunakan (dalam contoh ini, C #).

Opsi `-n` menunjukkan bahwa proyek dikeluarkan ke direktori baru, `hello`. Kami kemudian menavigasi ke direktori itu.

```
dotnet new console -lang C# -n hello
cd hello
```

Perintah sebelumnya menambahkan subdirektori dengan nama `obj` dengan beberapa file, dan beberapa file mandiri tambahan, ke direktori `hello`. Anda harus mencatat dua file kunci berikut:

- File `hello/hello.csproj` berisi informasi tentang proyek aplikasi konsol tersebut.
- File `hello/Program.cs` berisi kode aplikasi untuk dijalankan.

## Langkah 4: Tambahkan kode

Pada langkah ini, Anda menambahkan beberapa kode ke aplikasi.

Dari jendela Lingkungan di AWS Cloud9 IDE, buka `hello/Program.cs` file.

Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `Program.cs`.

```
using System;

namespace hello
{
  class Program
  {
    static void Main(string[] args)
    {
     if (args.Length < 2) {
       Console.WriteLine("Please provide 2 numbers");
```

```
        return;
      }

      Console.WriteLine("Hello, World!");

      Console.WriteLine("The sum of 2 and 3 is 5.");

      int sum = Int32.Parse(args[0]) + Int32.Parse(args[1]);

      Console.WriteLine("The sum of {0} and {1} is {2}.",
      args[0], args[1], sum);

    }
  }
}
```

## Langkah 5: Bangun dan jalankan kode

Pada langkah ini, Anda membangun proyek dan dependensinya ke dalam kumpulan file biner, termasuk file aplikasi yang dapat dijalankan. Kemudian anda menjalankan aplikasi tersebut.

1. Dalam IDE, buat pembangun untuk.NET sebagai berikut.

   a. Pada bilah menu, pilih Jalankan, Bangun Sistem, Sistem Bangun Baru.

   b. Pada tab Builder.build Saya, ganti konten tab dengan kode berikut.

   ```
   {
     "cmd" : ["dotnet", "build"],
     "info" : "Building..."
   }
   ```

   c. Pilih File, Simpan Sebagai.

   d. Untuk Filename, ketik `.NET.build`.

   e. Untuk Folder, ketik `/.c9/builders`.

   f. Pilih Simpan.

2. Dengan isi `Program.cs` file yang ditampilkan di editor, pilih Jalankan, Bangun Sistem, .NET. Kemudian pilih Jalankan, Bangun.

   Pembangun ini menambahkan subdirektori dengan nama `bin` dan menambahkan subdirektori dengan nama `Debug` ke subdirektori `hello/obj`. Perhatikan tiga file kunci berikut.

- File `hello/bin/Debug/netcoreapp3.1/hello.dll` adalah file aplikasi yang dapat dijalankan.
- File `hello/bin/Debug/netcoreapp3.1/hello.deps.json` mencantumkan dependensi aplikasi.
- File `hello/bin/Debug/netcoreapp3.1/hello.runtimeconfig.json` menentukan waktu aktif bersama dan versinya untuk aplikasi.

> **Note**
>
> Nama folder,`netcoreapp3.1`, mencerminkan versi SDK .NET yang digunakan dalam contoh ini. Anda mungkin melihat nomor yang berbeda dalam nama folder tergantung pada versi yang telah Anda pasang.

3. Buat runner untuk.NET sebagai berikut.

   a. Pada bilah menu, pilih Jalankan, Jalankan Dengan, Runner Baru.

   b. Pada tab My Runner.run, ganti isi tab dengan kode berikut.

   ```
   {
     "cmd" : ["dotnet", "run", "$args"],
     "working_dir": "$file_path",
     "info" : "Running..."
   }
   ```

   c. Pilih File, Simpan Sebagai.

   d. Untuk Filename, ketik `.NET.run`.

   e. Untuk Folder, ketik `/.c9/runners`.

   f. Pilih Simpan.

4. Jalankan aplikasi dengan dua bilangan bulat untuk ditambahkan (misalnya, 5 dan 9) sebagai berikut.

   a. Dengan isi file `Program.cs` yang ditampilkan di editor, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.

   b. Di tab [Baru] - Idle, pilih Runner: Auto, lalu pilih .NET.

   c. Di kotak Perintah, ketik `hello 5 9`.

d. Pilih Jalankan.

Secara default, runner ini menginstruksikan.NET untuk menjalankan `hello.dll` file di direktori. `hello/bin/Debug/netcoreapp3.1`

Bandingkan output Anda dengan yang berikut ini.

```
Hello, World!
The sum of 2 and 3 is 5.
The sum of 5 and 9 is 14.
```

## Langkah 6: Buat dan siapkan proyek aplikasi konsol.NET yang menggunakan AWS SDK untuk .NET

Anda dapat menyempurnakan sampel ini AWS SDK untuk .NET untuk menggunakan bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

Dalam proyek baru ini, Anda menambahkan referensi ke AWS SDK untuk .NET. AWS SDK untuk .NET Ini menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari kode .NET Anda. Anda kemudian mengatur manajemen AWS kredensyal di lingkungan Anda. AWS SDK untuk .NET Kebutuhan kredensyal ini untuk berinteraksi dengan AWS layanan.

### Untuk membuat proyek

1. Buat proyek aplikasi konsol.NET. Untuk melakukan ini, jalankan CLI .NET dengan **new**perintah, tentukan jenis template proyek aplikasi konsol dan bahasa pemrograman yang akan digunakan.

   Opsi `-n` menunjukkan bahwa proyek dikeluarkan ke direktori baru, `s3`. Kami kemudian menavigasi ke direktori itu.

   ```
   dotnet new console -lang C# -n s3
   cd s3
   ```

2. Tambahkan referensi proyek ke paket Amazon S3 di AWS SDK untuk .NET. Untuk melakukan ini, jalankan .NET CLI dengan **add package**perintah, tentukan nama paket Amazon S3 di. NuGet (NuGetmendefinisikan bagaimana paket untuk .NET dibuat, di-host, dan dikonsumsi, dan menyediakan alat untuk masing-masing peran tersebut.)

```
dotnet add package AWSSDK.S3
```

Saat Anda menambahkan referensi proyek ke paket Amazon S3, tambahkan NuGet juga referensi proyek ke paket lainnya. AWS SDK untuk .NET

> **Note**
>
> Untuk nama dan versi paket AWS terkait lainnya NuGet, lihat NuGetpaket yang ditandai dengan aws-sdk di situs web. NuGet

### Untuk mengatur manajemen AWS kredensial

Setiap kali Anda menggunakan AWS SDK untuk .NET untuk memanggil AWS layanan, Anda harus memberikan satu set AWS kredensyal dengan panggilan tersebut. Kredensyal ini menentukan apakah AWS SDK untuk .NET memiliki izin yang sesuai untuk melakukan panggilan itu. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Untuk menyimpan kredensial Anda di dalam lingkungan, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi selengkapnya, lihat Mengonfigurasi AWS Kredensyal di Panduan Pengembang.AWS SDK untuk .NET

## Langkah 7: Tambahkan kode AWS SDK

Pada langkah ini, Anda menambahkan kode untuk berinteraksi dengan Amazon S3 untuk membuat bucket, menghapus bucket yang baru saja Anda buat, lalu mencantumkan bucket yang tersedia.

Dari jendela Lingkungan di AWS Cloud9 IDE, buka `s3/Program.cs` file. Di editor, ganti konten file saat ini dengan kode berikut, lalu simpan file `Program.cs`.

```
using Amazon;
using Amazon.S3;
using Amazon.S3.Model;
using Amazon.S3.Util;
using System;
using System.Threading.Tasks;

namespace s3
```

```
{
  class Program
  {
   async static Task Main(string[] args)
   {
    if (args.Length < 2) {
      Console.WriteLine("Usage: <the bucket name> <the AWS Region to use>");
      Console.WriteLine("Example: my-test-bucket us-east-2");
      return;
    }

    if (args[1] != "us-east-2") {
      Console.WriteLine("Cannot continue. The only supported AWS Region ID is " +
      "'us-east-2'.");
       return;
     }

      var bucketRegion = RegionEndpoint.USEast2;
      // Note: You could add more valid AWS Regions above as needed.

      using (var s3Client = new AmazonS3Client(bucketRegion)) {
      var bucketName = args[0];

      // Create the bucket.
      try
      {
       if (await AmazonS3Util.DoesS3BucketExistV2Async(s3Client, bucketName))
       {
         Console.WriteLine("Cannot continue. Cannot create bucket. \n" +
         "A bucket named '{0}' already exists.", bucketName);
         return;
       } else {
         Console.WriteLine("\nCreating the bucket named '{0}'...", bucketName);
         await s3Client.PutBucketAsync(bucketName);
         }
       }
       catch (AmazonS3Exception e)
       {
        Console.WriteLine("Cannot continue. {0}", e.Message);
       }
       catch (Exception e)
       {
        Console.WriteLine("Cannot continue. {0}", e.Message);
       }
```

```
        // Confirm that the bucket was created.
        if (await AmazonS3Util.DoesS3BucketExistV2Async(s3Client, bucketName))
        {
           Console.WriteLine("Created the bucket named '{0}'.", bucketName);
        } else {
          Console.WriteLine("Did not create the bucket named '{0}'.", bucketName);
        }

        // Delete the bucket.
        Console.WriteLine("\nDeleting the bucket named '{0}'...", bucketName);
        await s3Client.DeleteBucketAsync(bucketName);

        // Confirm that the bucket was deleted.
        if (await AmazonS3Util.DoesS3BucketExistV2Async(s3Client, bucketName))
        {
           Console.WriteLine("Did not delete the bucket named '{0}'.", bucketName);
        } else {
          Console.WriteLine("Deleted the bucket named '{0}'.", bucketName);
        };

         // List current buckets.
        Console.WriteLine("\nMy buckets now are:");
        var response = await s3Client.ListBucketsAsync();

        foreach (var bucket in response.Buckets)
        {
        Console.WriteLine(bucket.BucketName);
        }
       }
     }
   }
}
```

## Langkah 8: Bangun dan jalankan kode AWS SDK

Pada langkah ini, Anda membangun proyek dan dependensinya ke dalam kumpulan file biner, termasuk file aplikasi yang dapat dijalankan. Kemudian anda menjalankan aplikasi tersebut.

1. Bangun proyek. Untuk melakukan ini, dengan konten file `s3/Program.cs` yang ditampilkan di editor, pada bilah menu, pilih Jalankan, Bangun.

2. Jalankan aplikasi dengan nama bucket Amazon S3 yang akan dibuat dan ID Wilayah AWS untuk membuat bucket di (misalnya, `my-test-bucket` dan `us-east-2`) sebagai berikut.

   a. Dengan isi file `s3/Program.cs` yang masih ditampilkan di editor, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.

   b. Di tab [Baru] - Idle, pilih Runner: Auto, lalu pilih .NET.

   c. Di kotak Command, ketikkan nama aplikasi, nama bucket Amazon S3 yang akan dibuat, dan ID AWS Wilayah untuk membuat bucket (misalnya,`s3 my-test-bucket us-east-2`).

   d. Pilih Jalankan.

      Secara default, runner ini menginstruksikan.NET untuk menjalankan `s3.dll` file di direktori. `s3/bin/Debug/netcoreapp3.1`

      Bandingkan hasil Anda dengan output berikut.

```
Creating a new bucket named 'my-test-bucket'...
Created the bucket named 'my-test-bucket'.

Deleting the bucket named 'my-test-bucket'...
Deleted the bucket named 'my-test-bucket'.

My buckets now are:
```

## Langkah 9: Membersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial Node.js untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menjalankan beberapa skrip Node.js di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan
- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js
- Langkah 5: Tambahkan kode AWS SDK
- Langkah 6: Jalankan kode AWS SDK
- Langkah 7: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda memasang Node.js, yang diperlukan untuk menjalankan sampel ini.

1. Dalam sesi terminal di AWS Cloud9 IDE, konfirmasikan apakah Node.js sudah diinstal dengan menjalankan **`node --version`**perintah. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika berhasil, output berisi nomor versi Node.js. Jika Node.js sudah terpasang, lewati ke Langkah 2: Menambahkan kode.
2. Jalankan perintah **`yum update`** untuk (Amazon Linux) atau **`apt update`** untuk (Ubuntu Server) untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug terpasang.

Untuk Amazon Linux:

```
sudo yum -y update
```

Untuk Ubuntu Server:

```
sudo apt update
```

3. Untuk menginstal Node.js, mulailah dengan menjalankan perintah ini untuk mengunduh Node Version Manager (nvm). (nvm adalah skrip shell Bash sederhana yang berguna untuk menginstal dan mengelola versi Node.js. Untuk informasi selengkapnya, lihat Pengelola Versi Node di GitHub situs web.)

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.5/install.sh | bash
```

4. Untuk mulai menggunakan nvm, tutup sesi terminal dan mulai lagi, atau cari file `~/.bashrc` yang berisi perintah untuk memuat nvm.

```
. ~/.bashrc
```

5. Jalankan perintah ini untuk menginstal Node.js 16 di Amazon Linux 2, Amazon Linux 1 dan Ubuntu 18.04. Instans Amazon Linux 1 dan Ubuntu 18.04 hanya mendukung Node.js hingga v16.

```
nvm install 16
```

Jalankan perintah ini untuk menginstal versi terbaru Node.js di Amazon Linux 2023 dan Ubuntu 22.04:

```
nvm install --lts && nvm alias default lts/*
```

> **Note**
>
> AWS Cloud9 Gambar AL2 023 terbaru telah diinstal Node.js 20, dan AWS Cloud9 gambar Amazon Linux 2 terbaru telah diinstal Node.js 18. Jika Anda ingin menginstal Node.js 18 di Amazon Linux 2 AWS Cloud9 secara manual, jalankan perintah berikut di terminal AWS Cloud9 IDE:

```
C9_NODE_INSTALL_DIR=~/.nvm/versions/node/v18.17.1
C9_NODE_URL=https://d3kgj69l4ph6w4.cloudfront.net/static/node-amazon/node-
v18.17.1-linux-x64.tar.gz
mkdir -p $C9_NODE_INSTALL_DIR
curl -fSsl $C9_NODE_URL  | tar xz --strip-components=1 -C
 "$C9_NODE_INSTALL_DIR"
nvm alias default v18.17.1
nvm use default
echo -e 'nvm use default' >> ~/.bash_profile
```

## Langkah 2: Menambahkan kode

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`hello.js`. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.)

```
console.log('Hello, World!');

console.log('The sum of 2 and 3 is 5.');

var sum = parseInt(process.argv[2], 10) + parseInt(process.argv[3], 10);

console.log('The sum of ' + process.argv[2] + ' and ' +
  process.argv[3] + ' is ' + sum + '.');
```

## Langkah 3: Jalankan kode

1. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Node.js.
3. Untuk Perintah, ketik `hello.js 5 9`. Dalam kode, 5 mewakili `process.argv[2]`, dan 9 mewakili `process.argv[3]`. (`process.argv[0]`mewakili nama runtime (`node`), dan `process.argv[1]` mewakili nama file (`hello.js`).)
4. Pilih tombol Jalankan, dan bandingkan output Anda.

```
Hello, World!
The sum of 2 and 3 is 5.
```

```
The sum of 5 and 9 is 14.
```

## Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js

Saat menjalankan skrip Node.js di AWS Cloud9, Anda dapat memilih antara AWS SDK untuk JavaScript versi 3 (V3) dan AWS SDK yang lebih lama untuk JavaScript versi 2 (V2). Seperti halnya V2, V3 memungkinkan Anda untuk dengan mudah bekerja dengan Amazon Web Services, tetapi telah ditulis TypeScript dan menambahkan beberapa fitur yang sering diminta, seperti paket termodulasi.

AWS SDK for JavaScript (V3)

> Anda dapat menyempurnakan sampel ini untuk menggunakan AWS SDK for JavaScript di Node.js untuk membuat bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja dibuat.
>
> Pada langkah ini, Anda menginstal dan mengonfigurasi modul klien layanan Amazon S3 dari AWS SDK untuk JavaScript di Node.js, yang menyediakan cara mudah untuk berinteraksi dengan AWS layanan Amazon S3, dari kode Anda. JavaScript
>
> Jika Anda ingin menggunakan AWS layanan lain, Anda perlu menginstalnya secara terpisah. Untuk informasi selengkapnya tentang menginstal AWS modul, lihat di Panduan AWS Pengembang (V3). Untuk informasi tentang cara memulai Node.js dan AWS SDK for JavaScript (V3), lihat Memulai Node.js di AWS SDK for JavaScript Developers Guide (V3).

Setelah Anda menginstal AWS SDK untuk JavaScript di Node.js, Anda harus menyiapkan manajemen kredensional di lingkungan Anda. AWS SDK untuk JavaScript di Node.js membutuhkan kredensil ini untuk berinteraksi dengan layanan. AWS

Untuk menginstal AWS SDK untuk JavaScript di Node.js

Gunakan NPM untuk menjalankan perintah **`install`**.

```
npm install @aws-sdk/client-s3
```

Untuk informasi selengkapnya, lihat Menginstal SDK untuk JavaScript di Panduan AWS SDK untuk JavaScript Pengembang.

Untuk mengatur manajemen kredensil di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK for JavaScript di Node.js untuk memanggil AWS layanan, Anda harus menyediakan satu set kredensional dengan panggilan tersebut. Kredensi ini menentukan apakah AWS SDK untuk JavaScript di Node.js memiliki izin yang sesuai untuk melakukan panggilan tersebut. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat Mengatur Kredensial di Node.js di Panduan Developer AWS SDK untuk JavaScript .

AWS SDK for JavaScript (V2)

Anda dapat menyempurnakan sampel ini untuk menggunakan AWS SDK for JavaScript di Node.js untuk membuat bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja dibuat.

Pada langkah ini, Anda menginstal dan mengonfigurasi AWS SDK untuk JavaScript di Node.js, yang menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari JavaScript kode Anda. Setelah Anda menginstal AWS SDK untuk JavaScript di Node.js, Anda harus menyiapkan manajemen kredensional di lingkungan Anda. AWS SDK untuk JavaScript di Node.js membutuhkan kredensil ini untuk berinteraksi dengan layanan. AWS

Untuk menginstal AWS SDK untuk JavaScript di Node.js

Gunakan NPM untuk menjalankan perintah **`install`**.

```
npm install aws-sdk
```

Untuk informasi selengkapnya, lihat Menginstal SDK untuk JavaScript di Panduan AWS SDK untuk JavaScript Pengembang.

Untuk mengatur manajemen kredensil di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK for JavaScript di Node.js untuk memanggil AWS layanan, Anda harus menyediakan satu set kredensional dengan panggilan tersebut. Kredensi ini menentukan apakah AWS SDK untuk JavaScript di Node.js memiliki izin yang sesuai untuk melakukan panggilan tersebut. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat Mengatur Kredensial di Node.js di Panduan Developer AWS SDK untuk JavaScript .

## Langkah 5: Tambahkan kode AWS SDK

AWS SDK for JavaScript (V3)

Pada langkah ini, Anda menambahkan beberapa kode lagi, kali ini untuk berinteraksi dengan Amazon S3 untuk membuat bucket, daftarkan bucket Anda yang tersedia, lalu hapus bucket yang baru saja Anda buat. Anda akan menjalankan kode ini nanti.

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`s3.js`.

```
import {
  CreateBucketCommand,
  DeleteBucketCommand,
  ListBucketsCommand,
  S3Client,
} from "@aws-sdk/client-s3";

const wait = async (milliseconds) => {
```

```
  return new Promise((resolve) => setTimeout(resolve, milliseconds));
};

export const main = async () => {
  const client = new S3Client({});
  const now = Date.now();
  const BUCKET_NAME = `easy-bucket-${now.toString()}`;

  const createBucketCommand = new CreateBucketCommand({ Bucket: BUCKET_NAME });
  const listBucketsCommand = new ListBucketsCommand({});
  const deleteBucketCommand = new DeleteBucketCommand({ Bucket: BUCKET_NAME });

  try {
    console.log(`Creating bucket ${BUCKET_NAME}.`);
    await client.send(createBucketCommand);
    console.log(`${BUCKET_NAME} created`);

    await wait(2000);

    console.log(`Here are your buckets:`);
    const { Buckets } = await client.send(listBucketsCommand);
    Buckets.forEach((bucket) => {
      console.log(` • ${bucket.Name}`);
    });

    await wait(2000);

    console.log(`Deleting bucket ${BUCKET_NAME}.`);
    await client.send(deleteBucketCommand);
    console.log(`${BUCKET_NAME} deleted`);
  } catch (err) {
    console.error(err);
  }
};

main();
```

AWS SDK for JavaScript (V2)

Pada langkah ini, Anda menambahkan beberapa kode lagi, kali ini untuk berinteraksi dengan Amazon S3 untuk membuat bucket, daftarkan bucket Anda yang tersedia, lalu hapus bucket yang baru saja Anda buat. Anda akan menjalankan kode ini nanti.

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`s3.js`.

```
if (process.argv.length < 4) {
  console.log(
    "Usage: node s3.js <the bucket name> <the AWS Region to use>\n" +
      "Example: node s3.js my-test-bucket us-east-2"
  );
  process.exit(1);
}

var AWS = require("aws-sdk"); // To set the AWS credentials and region.
var async = require("async"); // To call AWS operations asynchronously.

AWS.config.update({
  region: region,
});

var s3 = new AWS.S3({ apiVersion: "2006-03-01" });
var bucket_name = process.argv[2];
var region = process.argv[3];

var create_bucket_params = {
  Bucket: bucket_name,
  CreateBucketConfiguration: {
    LocationConstraint: region,
  },
};

var delete_bucket_params = { Bucket: bucket_name };

// List all of your available buckets in this AWS Region.
function listMyBuckets(callback) {
  s3.listBuckets(function (err, data) {
    if (err) {
    } else {
      console.log("My buckets now are:\n");

      for (var i = 0; i < data.Buckets.length; i++) {
        console.log(data.Buckets[i].Name);
      }
    }

    callback(err);
```

```
  });
}

// Create a bucket in this AWS Region.
function createMyBucket(callback) {
  console.log("\nCreating a bucket named " + bucket_name + "...\n");

  s3.createBucket(create_bucket_params, function (err, data) {
    if (err) {
      console.log(err.code + ": " + err.message);
    }

    callback(err);
  });
}

// Delete the bucket you just created.
function deleteMyBucket(callback) {
  console.log("\nDeleting the bucket named " + bucket_name + "...\n");

  s3.deleteBucket(delete_bucket_params, function (err, data) {
    if (err) {
      console.log(err.code + ": " + err.message);
    }

    callback(err);
  });
}

// Call the AWS operations in the following order.
async.series([
  listMyBuckets,
  createMyBucket,
  listMyBuckets,
  deleteMyBucket,
  listMyBuckets,
]);
```

## Langkah 6: Jalankan kode AWS SDK

1. Aktifkan kode untuk memanggil operasi Amazon S3 secara asinkron dengan menggunakan npm untuk menjalankan perintah **`install`** .

```
npm install async
```

2. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
3. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Node.js.
4. Jika Anda menggunakan AWS SDK for JavaScript (V3), untuk tipe Command. `s3.js` Jika Anda menggunakan AWS SDK untuk Javascript (v2), untuk tipe Commands`s3.js my-test-bucket us-east-2`, di mana `my-test-bucket` nama bucket yang ingin Anda buat lalu hapus, dan `us-east-2` merupakan ID AWS Wilayah tempat Anda ingin membuat bucket. Untuk selengkapnya IDs, lihat Amazon Simple Storage Service (Amazon S3) di halaman. Referensi Umum Amazon Web Services

> **Note**
>
> Nama bucket Amazon S3 harus unik di seluruh AWS—bukan hanya akun Anda. AWS

5. Pilih tombol Jalankan, dan bandingkan output Anda.

```
My buckets now are:

Creating a new bucket named 'my-test-bucket'...

My buckets now are:

my-test-bucket

Deleting the bucket named 'my-test-bucket'...

My buckets now are:
```

## Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# PHP tutorial untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menjalankan beberapa skrip PHP di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan
- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Instal dan konfigurasikan AWS SDK untuk PHP
- Langkah 5: Tambahkan kode AWS SDK
- Langkah 6: Jalankan kode AWS SDK
- Langkah 7: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

# Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda memasang PHP, yang diperlukan untuk menjalankan sampel ini.

> **Note**
>
> Prosedur berikut hanya memasang PHP. Untuk menginstal alat terkait seperti server web Apache dan database MySQL, lihat Tutorial: Menginstal Server Web LAMP di Amazon Linux di Panduan Pengguna Amazon. EC2

1. Dalam sesi terminal di AWS Cloud9 IDE, konfirmasikan apakah PHP sudah diinstal dengan menjalankan **`php --version`**perintah. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika berhasil, output berisi nomor versi PHP. Jika PHP sudah terpasang, lewati ke Langkah 2: Menambahkan kode.
2. Jalankan perintah **`yum update`** untuk (Amazon Linux) atau **`apt update`** untuk (Ubuntu Server) untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug terpasang.

   Untuk Amazon Linux 2 dan Amazon Linux:

   ```
   sudo yum -y update
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt update
   ```
3. Pasang PHP dengan menjalankan perintah **`install`** .

   Untuk Amazon Linux 2:

   ```
   sudo amazon-linux-extras install -y php7.2
   ```

   Untuk Amazon Linux:

   ```
   sudo yum -y install php72
   ```

> **Note**
>
> Anda dapat melihat versi Amazon Linux Anda menggunakan perintah berikut:
>
> ```
> cat /etc/system-release
> ```

Untuk Ubuntu Server:

```
sudo apt install -y php php-xml
```

Untuk informasi selengkapnya, lihat Pemasangan dan Konfigurasi di situs web PHP.

## Langkah 2: Menambahkan kode

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`hello.php`. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan, ketik `hello.php` untuk Filename, lalu pilih Simpan.)

```
<?php
  print('Hello, World!');

  print("\nThe sum of 2 and 3 is 5.");

  $sum = (int)$argv[1] + (int)$argv[2];

  print("\nThe sum of $argv[1] and $argv[2] is $sum.");
?>
```

> **Note**
>
> Kode sebelumnya tidak bergantung pada file eksternal apa pun. Namun, jika Anda pernah menyertakan atau memerlukan file PHP lain dalam file Anda, dan Anda ingin menggunakan file-file tersebut AWS Cloud9 untuk melakukan penyelesaian kode saat Anda mengetik, aktifkan Project, PHP Support, Enable PHP code completion setting di Preferences, dan kemudian tambahkan path ke file-file tersebut ke pengaturan Project, PHP Support, PHP

Completion Include Paths. (Untuk melihat dan mengubah preferensi Anda, pilih AWS Cloud9, Preferensi pada bilah menu.)

## Langkah 3: Jalankan kode

1. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih PHP.(cli).
3. Untuk Perintah, ketik `hello.php 5 9`. Dalam kode, 5 mewakili `$argv[1]`, dan 9 mewakili `$argv[2]`. (`$argv[0]`mewakili nama file (`hello.php`).)
4. Pilih tombol Jalankan, dan bandingkan output Anda.

```
Hello, World!
The sum of 2 and 3 is 5.
The sum of 5 and 9 is 14.
```

## Langkah 4: Instal dan konfigurasikan AWS SDK untuk PHP

Anda dapat menyempurnakan sampel ini AWS SDK untuk PHP untuk menggunakan bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

Pada langkah ini, Anda menginstal dan mengonfigurasi AWS SDK untuk PHP, yang menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari kode PHP Anda.

Sebelum Anda dapat menginstal AWS SDK untuk PHP, Anda harus menginstal Composer. Setelah Anda menginstal AWS SDK untuk PHP, Anda harus mengatur manajemen kredensional di lingkungan Anda. AWS SDK untuk PHP Kebutuhan kredensil ini untuk berinteraksi dengan AWS layanan.

### Untuk memasang Composer

Jalankan perintah **`curl`** dengan opsi diam (`-s`) dan tunjukkan error (`-S`), merampingkan pemasangan Composer ke dalam file arsip PHP (PHAR), dengan nama `composer.phar` menurut konvensi.

```
curl -sS https://getcomposer.org/installer | php
```

### Untuk menginstal AWS SDK untuk PHP

Untuk Ubuntu Server, pasang paket tambahan yang dibutuhkan Composer untuk memasang AWS SDK untuk PHP.

```
sudo apt install -y php-xml php-curl
```

Untuk Amazon Linux atau Ubuntu Server, gunakan perintah php untuk menjalankan pemasang Composer untuk memasang AWS SDK untuk PHP.

```
php composer.phar require aws/aws-sdk-php
```

Perintah ini membuat beberapa folder dan file di lingkungan Anda. File utama yang akan Anda gunakan adalah `autoload.php`, yang ada di folder `vendor` di lingkungan Anda.

> **Note**
>
> Setelah pemasangan, Composer mungkin menyarankan Anda memasang dependensi tambahan. Anda dapat melakukan ini dengan perintah seperti berikut, menentukan daftar dependensi yang akan dipasang. Misalnya, perintah berikut menginstruksikan Composer untuk memasang daftar dependensi berikut.
>
> ```
> php composer.phar require psr/log ext-curl doctrine/cache aws/aws-php-sns-message-validator
> ```

Untuk informasi selengkapnya, lihat Pemasangan di dalam Panduan Developer AWS SDK untuk PHP .

### Untuk mengatur pengelolaan kredensial di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK untuk PHP untuk memanggil AWS layanan, Anda harus memberikan satu set kredensil dengan panggilan tersebut. Kredensil ini menentukan apakah AWS SDK untuk PHP memiliki izin yang sesuai untuk melakukan panggilan itu. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat bagian "Membuat klien" dari Penggunaan Basic di Panduan Developer AWS SDK untuk PHP .

## Langkah 5: Tambahkan kode AWS SDK

Pada langkah ini, Anda menambahkan beberapa kode lagi, kali ini untuk berinteraksi dengan Amazon S3 untuk membuat bucket, daftarkan bucket Anda yang tersedia, lalu hapus bucket yang baru saja Anda buat. Anda akan menjalankan kode ini nanti.

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`s3.php`.

```
<?php
require './vendor/autoload.php';

if ($argc < 4) {
    exit("Usage: php s3.php <the time zone> <the bucket name> <the AWS Region to use>
\n" .
        "Example: php s3.php America/Los_Angeles my-test-bucket us-east-2");
}

$timeZone = $argv[1];
$bucketName = $argv[2];
$region = $argv[3];

date_default_timezone_set($timeZone);

$s3 = new Aws\S3\S3Client([
    'region' => $region,
    'version' => '2006-03-01'
```

```
]);

# Lists all of your available buckets in this AWS Region.
function listMyBuckets($s3)
{
    print("\nMy buckets now are:\n");

    $promise = $s3->listBucketsAsync();

    $result = $promise->wait();

    foreach ($result['Buckets'] as $bucket) {
        print("\n");
        print($bucket['Name']);
    }
}

listMyBuckets($s3);

# Create a new bucket.
print("\n\nCreating a new bucket named '$bucketName'...\n");

try {
    $promise = $s3->createBucketAsync([
        'Bucket' => $bucketName,
        'CreateBucketConfiguration' => [
            'LocationConstraint' => $region
        ]
    ]);

    $promise->wait();
} catch (Exception $e) {
    if ($e->getCode() == 'BucketAlreadyExists') {
        exit("\nCannot create the bucket. " .
            "A bucket with the name '$bucketName' already exists. Exiting.");
    }
}

listMyBuckets($s3);

# Delete the bucket you just created.
print("\n\nDeleting the bucket named '$bucketName'...\n");

$promise = $s3->deleteBucketAsync([
```

```
    'Bucket' => $bucketName
]);

$promise->wait();

listMyBuckets($s3);

?>
```

## Langkah 6: Jalankan kode AWS SDK

1. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih PHP.(cli).
3. Untuk Perintah, ketik `s3.php America/Los_Angeles my-test-bucket us-east-2`, di mana:

   - `America/Los_Angeles` adalah ID zona waktu default Anda. Untuk selengkapnya IDs, lihat Daftar Zona Waktu yang Didukung di situs web PHP.
   - `my-test-bucket` adalah nama bucket yang ingin Anda buat dan kemudian hapus.

     > **Note**
     >
     > Nama bucket Amazon S3 harus unik di seluruh AWS—bukan hanya akun Anda. AWS

   - `us-east-2`adalah ID AWS Wilayah tempat Anda ingin membuat bucket. Untuk selengkapnya IDs, lihat Amazon Simple Storage Service (Amazon S3) di halaman. Referensi Umum Amazon Web Services
4. Pilih tombol Jalankan, dan bandingkan output Anda.

```
My buckets now are:

Creating a new bucket named 'my-test-bucket'...

My buckets now are:

my-test-bucket

Deleting the bucket named 'my-test-bucket'...
```

```
My buckets now are:
```

### Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk petunjuk, silakan lihat Menghapus lingkungan di AWS Cloud9.

### Memecahkan masalah dengan PHP runner untuk AWS Cloud9

Jika Anda mengalami masalah dengan runner PHP CLI, Anda harus memastikan bahwa runner telah diatur ke PHP dan mode debugger diaktifkan.

## AWS SDK for Ruby di AWS Cloud9

Untuk informasi tentang penggunaan AWS Cloud9 dengan AWS SDK for Ruby, AWS Cloud9 lihat Menggunakan AWS dengan SDK for Ruby di Panduan AWS Pengembang SDK for Ruby.

> **Note**
>
> Mengikuti tutorial ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

## Pergi tutorial untuk AWS Cloud9

Tutorial ini memungkinkan Anda untuk menjalankan beberapa kode Go di lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik


- Prasyarat
- Langkah 1: Pasang alat yang diperlukan

- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Instal dan konfigurasikan AWS SDK untuk Go
- Langkah 5: Tambahkan kode AWS SDK
- Langkah 6: Jalankan kode AWS SDK
- Langkah 7: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda memasang dan mengonfigurasi Go, yang diperlukan untuk menjalankan sampel ini.

1. Dalam sesi terminal di AWS Cloud9 IDE, konfirmasikan apakah Go sudah diinstal dengan menjalankan **`go version`**perintah. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika berhasil, output harus berisi nomor versi Go. Jika tidak, pesan kesalahan harus menjadi output. Jika Go sudah terpasang, lewati ke Langkah 2: Menambahkan kode.
2. Jalankan perintah **`yum update`** for (Amazon Linux) atau **`apt update`** for (Ubuntu Server) untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug terpasang.

   Untuk Amazon Linux:

```
sudo yum -y update
```

Untuk Ubuntu Server:

```
sudo apt update
```

3. Untuk memasang Go, jalankan perintah ini, satu per satu.

```
wget https://storage.googleapis.com/golang/go1.9.3.linux-amd64.tar.gz # Download
 the Go installer.
sudo tar -C /usr/local -xzf ./go1.9.3.linux-amd64.tar.gz              # Install Go.
rm ./go1.9.3.linux-amd64.tar.gz                                       # Delete the
 installer.
```

Perintah sebelumnya mengasumsikan versi stabil terbaru dari Go pada saat topik ini ditulis. Untuk informasi lebih lanjut, lihat Unduhan di situs web Bahasa Pemrograman Go.

4. Tambahkan path ke biner Go ke variabel lingkungan `PATH` Anda, seperti ini.

   a. Buka file profil shell Anda (misalnya, `~/.bashrc`) untuk diedit.

   b. Di akhir baris kode ini, ketikkan yang berikut ini, sehingga kodenya sekarang terlihat seperti ini.

   ```
   PATH=$PATH:/usr/local/go/bin
   ```

   c. Simpan file tersebut.

5. Cari file `~/.bashrc` sehingga terminal sekarang dapat menemukan biner Go yang baru saja Anda referensikan.

```
. ~/.bashrc
```

6. Konfirmasikan bahwa Go sekarang berhasil dipasang dan dikonfigurasi dengan menjalankan perintah **`go version`** . Jika berhasil, output berisi nomor versi Go.

## Langkah 2: Menambahkan kode

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`hello.go`. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.)

```
package main

import (
  "fmt"
  "os"
  "strconv"
)

func main() {
  fmt.Printf("Hello, World!\n")

  fmt.Printf("The sum of 2 and 3 is 5.\n")

  first, _ := strconv.Atoi(os.Args[1])
  second, _ := strconv.Atoi(os.Args[2])
  sum := first + second

  fmt.Printf("The sum of %s and %s is %s.",
    os.Args[1], os.Args[2], strconv.Itoa(sum))
}
```

## Langkah 3: Jalankan kode

1. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Go.

> **Note**
>
> Jika Go tidak tersedia, Anda dapat membuat runner khusus untuk Go.
>
> 1. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Runner Baru.
> 2. Pada tab Runner.run Saya, ganti konten tab dengan kode ini.
>
> ```
> {
>   "cmd" : ["go", "run", "$file", "$args"],
>   "info" : "Running $project_path$file_name...",
>   "selector" : "source.go"
> }
> ```

3. Pilih File, Simpan Sebagai pada menu bar, dan simpan file sebagai `Go.run` di folder `/.c9/runners`.
4. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Go.
5. Pilih tab hello.go untuk mengaktifkannya.

3. Untuk Perintah, ketik `hello.go 5 9`. Dalam kode, 5 mewakili `os.Args[1]`, dan 9 mewakili `os.Args[2]`.

4. Pilih tombol Jalankan, dan bandingkan output Anda.

```
Hello, World!
The sum of 2 and 3 is 5.
The sum of 5 and 9 is 14.
```

# Langkah 4: Instal dan konfigurasikan AWS SDK untuk Go

Anda dapat menyempurnakan sampel ini AWS SDK untuk Go untuk menggunakan bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja Anda buat.

Pada langkah ini, Anda menginstal dan mengonfigurasi AWS SDK untuk Go, yang menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari kode Go Anda. Sebelum Anda memasang AWS SDK untuk Go, Anda harus mengatur variabel lingkungan `GOPATH` Anda. Setelah Anda memasang AWS SDK untuk Go dan mengatur variabel lingkungan `GOPATH`, Anda harus menyiapkan pengelolaan kredensial di lingkungan Anda. AWS SDK untuk Go Kebutuhan kredensil ini untuk berinteraksi dengan AWS layanan.

## Untuk mengatur variabel lingkungan GOPATH Anda

1. Buka file `~/.bashrc` Anda untuk mengedit.
2. Setelah baris terakhir dalam file, ketik kode ini.

   ```
   GOPATH=~/environment/go

   export GOPATH
   ```

3. Simpan file tersebut.
4. Cari file `~/.bashrc` sehingga terminal sekarang dapat menemukan variabel lingkungan `GOPATH` yang baru saja Anda referensikan.

   ```
   . ~/.bashrc
   ```

5. Konfirmasikan bahwa variabel lingkungan `GOPATH` berhasil diatur dengan menjalankan perintah **`echo $GOPATH`** . Jika berhasil, `/home/ec2-user/environment/go` atau `/home/ubuntu/environment/go` harus menjadi output.

## Untuk menginstal AWS SDK untuk Go

Jalankan **`go get`**perintah, tentukan lokasi AWS SDK untuk Go sumber.

```
go get -u github.com/aws/aws-sdk-go/...
```

Go menginstal AWS SDK untuk Go sumber ke lokasi yang ditentukan oleh variabel `GOPATH` lingkungan Anda, yang merupakan `go` folder di lingkungan Anda.

### Untuk mengatur pengelolaan kredensial di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK untuk Go untuk memanggil AWS layanan, Anda harus memberikan satu set kredensil dengan panggilan tersebut. Kredensil ini menentukan apakah AWS SDK untuk Go memiliki izin yang sesuai untuk melakukan panggilan itu. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat Menentukan Kredensial di Panduan Developer AWS SDK untuk Go .

## Langkah 5: Tambahkan kode AWS SDK

Pada langkah ini, Anda menambahkan beberapa kode lagi, kali ini untuk berinteraksi dengan Amazon S3 untuk membuat bucket, daftarkan bucket Anda yang tersedia, lalu hapus bucket yang baru saja Anda buat. Anda akan menjalankan kode ini nanti.

Di AWS Cloud9 IDE, buat file dengan konten ini, dan simpan file dengan namanya`s3.go`.

```
package main

import (
 "fmt"
 "os"

 "github.com/aws/aws-sdk-go/aws"
 "github.com/aws/aws-sdk-go/aws/session"
 "github.com/aws/aws-sdk-go/service/s3"
)

func main() {

 if len(os.Args) < 3 {
  fmt.Printf("Usage: go run s3.go <the bucket name> <the AWS Region to use>\n" +
   "Example: go run s3.go my-test-bucket us-east-2\n")
  os.Exit(1)
 }

 sess := session.Must(session.NewSessionWithOptions(session.Options{
  SharedConfigState: session.SharedConfigEnable,
 }))
 svc := s3.New(sess, &aws.Config{
```

```
  Region: aws.String(os.Args[2]),
 })

 listMyBuckets(svc)
 createMyBucket(svc, os.Args[1], os.Args[2])
 listMyBuckets(svc)
 deleteMyBucket(svc, os.Args[1])
 listMyBuckets(svc)
}

// List all of your available buckets in this AWS Region.
func listMyBuckets(svc *s3.S3) {
 result, err := svc.ListBuckets(nil)

 if err != nil {
  exitErrorf("Unable to list buckets, %v", err)
 }

 fmt.Println("My buckets now are:\n")

 for _, b := range result.Buckets {
  fmt.Printf(aws.StringValue(b.Name) + "\n")
 }

 fmt.Printf("\n")
}

// Create a bucket in this AWS Region.
func createMyBucket(svc *s3.S3, bucketName string, region string) {
 fmt.Printf("\nCreating a new bucket named '" + bucketName + "'...\n\n")

 _, err := svc.CreateBucket(&s3.CreateBucketInput{
  Bucket: aws.String(bucketName),
  CreateBucketConfiguration: &s3.CreateBucketConfiguration{
   LocationConstraint: aws.String(region),
  },
 })

 if err != nil {
  exitErrorf("Unable to create bucket, %v", err)
 }

 // Wait until bucket is created before finishing
 fmt.Printf("Waiting for bucket %q to be created...\n", bucketName)
```

```
 err = svc.WaitUntilBucketExists(&s3.HeadBucketInput{
  Bucket: aws.String(bucketName),
 })
}

// Delete the bucket you just created.
func deleteMyBucket(svc *s3.S3, bucketName string) {
 fmt.Printf("\nDeleting the bucket named '" + bucketName + "'...\n\n")

 _, err := svc.DeleteBucket(&s3.DeleteBucketInput{
  Bucket: aws.String(bucketName),
 })

 if err != nil {
  exitErrorf("Unable to delete bucket, %v", err)
 }

 // Wait until bucket is deleted before finishing
 fmt.Printf("Waiting for bucket %q to be deleted...\n", bucketName)

 err = svc.WaitUntilBucketNotExists(&s3.HeadBucketInput{
  Bucket: aws.String(bucketName),
 })
}

// If there's an error, display it.
func exitErrorf(msg string, args ...interface{}) {
 fmt.Fprintf(os.Stderr, msg+"\n", args...)
 os.Exit(1)
}
```

## Langkah 6: Jalankan kode AWS SDK

1. Di AWS Cloud9 IDE, pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Konfigurasi Jalankan Baru.
2. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Go.
3. Untuk Commands`s3.go` *`YOUR_BUCKET_NAME`* *`THE_AWS_REGION`* , ketik, di mana *`YOUR_BUCKET_NAME`* nama bucket yang ingin Anda buat lalu hapus, dan *`THE_AWS_REGION`* merupakan ID AWS Wilayah tempat Anda ingin membuat bucket. Sebagai contoh, untuk Wilayah

US East (Ohio)., gunakan `us-east-2`. Untuk selengkapnya IDs, lihat Amazon Simple Storage Service (Amazon S3) di halaman. Referensi Umum Amazon Web Services

> **Note**
>
> Nama bucket Amazon S3 harus unik di seluruh AWS—bukan hanya akun Anda. AWS

4. Pilih tombol Jalankan, dan bandingkan output Anda.

```
My buckets now are:

Creating a new bucket named 'my-test-bucket'...

My buckets now are:

my-test-bucket

Deleting the bucket named 'my-test-bucket'...

My buckets now are:
```

### Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

## TypeScript tutorial untuk AWS Cloud9

Tutorial ini menunjukkan cara bekerja dengan TypeScript dalam lingkungan AWS Cloud9 pengembangan.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2 dan Amazon S3. Untuk informasi selengkapnya, lihat EC2 Harga Amazon dan Harga Amazon S3.

Topik

- Prasyarat

- Langkah 1: Pasang alat yang diperlukan
- Langkah 2: Menambahkan kode
- Langkah 3: Jalankan kode
- Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js
- Langkah 5: Tambahkan kode AWS SDK
- Langkah 6: Jalankan kode AWS SDK
- Langkah 7: Bersihkan


## Prasyarat

Sebelum Anda menggunakan sampel ini, pastikan pengaturan Anda memenuhi persyaratan berikut:

- Anda harus memiliki lingkungan AWS Cloud9 EC2 pengembangan yang ada. Contoh ini mengasumsikan bahwa Anda sudah memiliki EC2 lingkungan yang terhubung ke EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Peladen. Jika Anda memiliki jenis lingkungan atau sistem operasi yang berbeda, Anda mungkin perlu menyesuaikan petunjuk sampel ini untuk menyiapkan alat terkait. Untuk informasi selengkapnya, lihat Menciptakan lingkungan di AWS Cloud9.
- Anda memiliki AWS Cloud9 IDE untuk lingkungan yang ada sudah terbuka. Saat Anda membuka lingkungan, AWS Cloud9 buka IDE untuk lingkungan itu di browser web Anda. Untuk informasi selengkapnya, lihat Membuka lingkungan di AWS Cloud9.

## Langkah 1: Pasang alat yang diperlukan

Pada langkah ini, Anda menginstal TypeScript dengan menggunakan Node Package Manager (**`npm`**). Untuk memasang **`npm`** , Anda menggunakan Node Version Manager ( **`nvm`** ). Jika Anda tidak memiliki **`nvm`** , Anda memasangnya pada langkah ini terlebih dulu.

1. Dalam sesi terminal di AWS Cloud9 IDE, konfirmasikan TypeScript apakah sudah diinstal dengan menjalankan TypeScript compiler baris perintah dengan **`--version`**opsi. (Untuk memulai sesi terminal baru, pada bilah menu, pilih Jendela, Terminal Baru.) Jika berhasil, output berisi nomor TypeScript versi. Jika TypeScript diinstal, lewati ke depanLangkah 2: Menambahkan kode.

   ```
   tsc --version
   ```

2. Konfirmasi apakah **npm** sudah terpasang dengan menjalankan **npm** dengan opsi **--version**. Jika berhasil, output berisi nomor versi **npm**. Jika **npm**diinstal, lewati ke langkah 10 dalam prosedur ini untuk digunakan **npm**untuk menginstal TypeScript.

```
npm --version
```

3. Jalankan perintah **yum update** untuk (Amazon Linux) atau **apt update** untuk (Ubuntu Server) untuk membantu memastikan pembaruan keamanan terbaru dan perbaikan bug terpasang.

   Untuk Amazon Linux:

```
sudo yum -y update
```

   Untuk Ubuntu Server:

```
sudo apt update
```

4. Untuk menginstal **npm**, mulailah dengan menjalankan perintah berikut untuk men-download Node Version Manager (**nvm**). (**nvm**adalah skrip shell Bash sederhana yang berguna untuk menginstal dan mengelola versi Node.js. Untuk informasi selengkapnya, lihat Pengelola Versi Node di GitHub situs web.)

```
curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.0/install.sh | bash
```

5. Untuk mulai menggunakan **nvm**, tutup sesi terminal dan mulai lagi, atau cari file `~/.bashrc` yang berisi perintah untuk memuat **nvm**.

```
. ~/.bashrc
```

6. Konfirmasi apakah **nvm** sudah terpasang dengan menjalankan **nvm** dengan opsi **--version**.

```
nvm --version
```

7. Instal versi terbaru 16 dari Node.js dengan menjalankan **nvm**. (**npm**termasuk dalam Node.js.)

```
nvm install v16
```

8. Konfirmasi bahwa Node.js terpasang dengan menjalankan versi baris perintah dari Node.js dengan opsi **`--version`** .

```
node --version
```

9. Konfirmasi apakah **`npm`** sudah terpasang dengan menjalankan **`npm`** dengan opsi **`--version`** .

```
npm --version
```

10. Instal TypeScript **`npm`**dengan menjalankan dengan **`-g`**opsi. Ini diinstal TypeScript sebagai paket global di lingkungan.

```
npm install -g typescript
```

11. Konfirmasikan yang TypeScript diinstal dengan menjalankan TypeScript kompiler baris perintah dengan **`--version`**opsi.

```
tsc --version
```

## Langkah 2: Menambahkan kode

1. Di AWS Cloud9 IDE, buat file bernama`hello.ts`. (Untuk membuat file, pada bilah menu, pilih File, File Baru. Untuk menyimpan file, pilih File, Simpan.)
2. Di sebuah terminal di IDE, dari direktori yang sama seperti file `hello.ts`, jalankan **`npm`** untuk memasang pustaka `@types/node`.

```
npm install @types/node
```

   Hal ini menambahkan folder `node_modules/@types/node` dalam direktori yang sama dengan file `hello.ts`. Folder baru ini berisi definisi tipe Node.js yang TypeScript perlu nanti dalam prosedur ini untuk `console.log` dan `process.argv` properti yang akan Anda tambahkan ke `hello.ts` file.

3. Tambahkan kode berikut ke file `hello.ts`:

```
console.log('Hello, World!');
```

```
console.log('The sum of 2 and 3 is 5.');

const sum: number = parseInt(process.argv[2], 10) + parseInt(process.argv[3], 10);

console.log('The sum of ' + process.argv[2] + ' and ' +
  process.argv[3] + ' is ' + sum + '.');
```

## Langkah 3: Jalankan kode

1. Di terminal, dari direktori yang sama dengan `hello.ts` file, jalankan TypeScript kompiler. Menentukan file `hello.ts` dan pustaka tambahan untuk disertakan.

   ```
   tsc hello.ts --lib es6
   ```

   TypeScript menggunakan `hello.ts` file dan satu set file pustaka ECMAScript 6 (ES6) untuk mentranspile TypeScript kode dalam `hello.ts` file menjadi JavaScript kode yang setara dalam file bernama`hello.js`.
2. Di jendela Lingkungan, buka file `hello.js`.
3. Pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.
4. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Node.js.
5. Untuk Perintah, ketik `hello.js 5 9`. Dalam kode, `5` mewakili `process.argv[2]`, dan `9` mewakili `process.argv[3]`. (`process.argv[0]`mewakili nama runtime (`node`), dan `process.argv[1]` mewakili nama file (`hello.js`).)
6. Pilih Jalankan, dan bandingkan output Anda. Setelah Anda selesai, pilih Berhenti.

   ```
   Hello, World!
   The sum of 2 and 3 is 5.
   The sum of 5 and 9 is 14.
   ```

> **Note**
>
> Alih-alih membuat konfigurasi run baru di IDE, Anda juga dapat menjalankan kode ini dengan menjalankan perintah **`node hello.js 5 9`** dari terminal.

## Langkah 4: Instal dan konfigurasikan AWS SDK untuk JavaScript di Node.js

Anda dapat menyempurnakan sampel ini untuk menggunakan AWS SDK for JavaScript di Node.js untuk membuat bucket Amazon S3, mencantumkan bucket yang tersedia, lalu menghapus bucket yang baru saja dibuat.

Pada langkah ini, Anda menginstal dan mengkonfigurasi AWS SDK untuk JavaScript di Node.js. SDK menyediakan cara mudah untuk berinteraksi dengan AWS layanan seperti Amazon S3, dari JavaScript kode Anda. Setelah Anda menginstal AWS SDK untuk JavaScript di Node.js, Anda harus menyiapkan manajemen kredensional di lingkungan Anda. SDK membutuhkan kredensil ini untuk berinteraksi dengan layanan. AWS

### Untuk menginstal AWS SDK untuk JavaScript di Node.js

Dalam sesi terminal di AWS Cloud9 IDE, dari direktori yang sama dengan `hello.js` file dariLangkah 3: Jalankan kode, jalankan **`npm`**untuk menginstal AWS SDK untuk JavaScript di Node.js.

```
npm install aws-sdk
```

Perintah ini menambahkan beberapa folder ke folder `node_modules` dari Langkah 3: Jalankan kode. Folder ini berisi kode sumber dan dependensi untuk AWS SDK untuk JavaScript di Node.js. Untuk informasi selengkapnya, lihat Menginstal SDK untuk JavaScript di Panduan AWS SDK untuk JavaScript Pengembang.

### Untuk mengatur pengelolaan kredensial di lingkungan Anda

Setiap kali Anda menggunakan AWS SDK for JavaScript di Node.js untuk memanggil AWS layanan, Anda harus menyediakan satu set kredensional dengan panggilan tersebut. Kredensi ini menentukan apakah AWS SDK untuk JavaScript di Node.js memiliki izin yang sesuai untuk melakukan panggilan tersebut. Jika kredensial tidak mencakup izin yang sesuai, panggilan akan gagal.

Pada langkah ini, Anda menyimpan kredensial Anda di dalam lingkungan. Untuk melakukannya, ikuti petunjuk di Menelepon Layanan AWS dari lingkungan di AWS Cloud9, lalu kembali ke topik ini.

Untuk informasi tambahan, lihat Mengatur Kredensial di Node.js di Panduan Developer AWS SDK untuk JavaScript .

## Langkah 5: Tambahkan kode AWS SDK

Pada langkah ini, Anda menambahkan beberapa kode lagi, kali ini untuk berinteraksi dengan Amazon S3 untuk membuat bucket, daftarkan bucket Anda yang tersedia, lalu hapus bucket yang baru saja Anda buat. Anda akan menjalankan kode ini nanti.

1. Di AWS Cloud9 IDE, di direktori yang sama dengan `hello.js` file di langkah sebelumnya, buat file bernama`s3.ts`.
2. Dari terminal di AWS Cloud9 IDE, di direktori yang sama dengan `s3.ts` file, aktifkan kode untuk memanggil operasi Amazon S3 secara asinkron dengan menjalankan **`npm`**dua kali untuk menginstal pustaka asinkron untuk dan lagi untuk. TypeScript JavaScript

   ```
   npm install @types/async # For TypeScript.
   npm install async        # For JavaScript.
   ```

3. Tambahkan kode berikut ke file `s3.ts`:

   ```
   import * as async from 'async';
   import * as AWS from 'aws-sdk';

   if (process.argv.length < 4) {
     console.log('Usage: node s3.js <the bucket name> <the AWS Region to use>\n' +
   ```

```
    'Example: node s3.js my-test-bucket us-east-2');
  process.exit(1);
}

const AWS = require('aws-sdk'); // To set the AWS credentials and AWS Region.
const async = require('async'); // To call AWS operations asynchronously.

const s3: AWS.S3 = new AWS.S3({apiVersion: '2006-03-01'});
const bucket_name: string = process.argv[2];
const region: string = process.argv[3];

AWS.config.update({
  region: region
});

const create_bucket_params: any = {
  Bucket: bucket_name,
  CreateBucketConfiguration: {
    LocationConstraint: region
  }
};

const delete_bucket_params: any = {
  Bucket: bucket_name
};

// List all of your available buckets in this AWS Region.
function listMyBuckets(callback): void {
  s3.listBuckets(function(err, data) {
    if (err) {

    } else {
      console.log("My buckets now are:\n");

      for (let i: number = 0; i < data.Buckets.length; i++) {
        console.log(data.Buckets[i].Name);
      }
    }

    callback(err);
  });
}

// Create a bucket in this AWS Region.
```

```
function createMyBucket(callback): void {
  console.log("\nCreating a bucket named '" + bucket_name + "'...\n");

  s3.createBucket(create_bucket_params, function(err, data) {
    if (err) {
      console.log(err.code + ": " + err.message);
    }

    callback(err);
  });
}

// Delete the bucket you just created.
function deleteMyBucket(callback): void {
  console.log("\nDeleting the bucket named '" + bucket_name + "'...\n");

  s3.deleteBucket(delete_bucket_params, function(err, data) {
    if (err) {
      console.log(err.code + ": " + err.message);
    }

    callback(err);
  });
}

// Call the AWS operations in the following order.
async.series([
  listMyBuckets,
  createMyBucket,
  listMyBuckets,
  deleteMyBucket,
  listMyBuckets
]);
```

## Langkah 6: Jalankan kode AWS SDK

1. Di terminal, dari direktori yang sama dengan `s3.ts` file, jalankan TypeScript kompiler. Menentukan file `s3.ts` dan pustaka tambahan untuk disertakan.

```
tsc s3.ts --lib es6
```

TypeScript menggunakan `s3.ts` file, AWS SDK untuk JavaScript di Node.js, pustaka async, dan satu set file pustaka ECMAScript 6 (ES6) untuk mengubah kode dalam `s3.ts` file menjadi TypeScript kode yang setara dalam file JavaScript bernama. `s3.js`

2. Di jendela Lingkungan, buka file `s3.js`.
3. Pada bilah menu, pilih Jalankan, Jalankan Konfigurasi, Jalankan Konfigurasi Baru.
4. Di tab [Baru] - Diam, pilih Runner: Otomatis, lalu pilih Node.js.
5. Untuk Commands`s3.js` *`YOUR_BUCKET_NAME THE_AWS_REGION`* , ketik, di mana *`YOUR_BUCKET_NAME`* nama bucket yang ingin Anda buat lalu hapus, dan *`THE_AWS_REGION`* merupakan ID AWS Region untuk membuat bucket. Sebagai contoh, untuk Wilayah US East (Ohio)., gunakan `us-east-2`. Untuk selengkapnya IDs, lihat Amazon Simple Storage Service (Amazon S3) di halaman. Referensi Umum Amazon Web Services

   > **Note**
   >
   > Nama bucket Amazon S3 harus unik di seluruh AWS—bukan hanya akun Anda. AWS

6. Pilih Jalankan, dan bandingkan output Anda. Setelah Anda selesai, pilih Berhenti.

```
My buckets now are:

Creating a new bucket named 'my-test-bucket'...

My buckets now are:

my-test-bucket

Deleting the bucket named 'my-test-bucket'...

My buckets now are:
```

## Langkah 7: Bersihkan

Untuk mencegah tagihan yang sedang berlangsung ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghapus lingkungan. Untuk instruksi, lihat Menghapus lingkungan di AWS Cloud9.

# Tutorial Docker untuk AWS Cloud9

Tutorial ini menunjukkan cara menghubungkan lingkungan pengembangan AWS Cloud9 SSH ke wadah Docker yang sedang berjalan di dalam instance Amazon Linux di Amazon. EC2 Ini memungkinkan Anda menggunakan AWS Cloud9 IDE untuk bekerja dengan kode dan file di dalam wadah Docker dan menjalankan perintah pada wadah itu. Untuk informasi tentang Docker, lihat Apa itu Docker di situs web Docker.

Mengikuti tutorial ini dan membuat sampel ini dapat mengakibatkan biaya ke AWS akun Anda. Ini termasuk kemungkinan biaya untuk layanan seperti Amazon EC2. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

Topik


- Prasyarat
- Langkah 1: Pasang dan jalankan Docker
- Langkah 2: Bangun citra
- Langkah 3: Jalankan kontainer
- Langkah 4: Buat lingkungan
- Langkah 5: Jalankan kode
- Langkah 6: Membersihkan


## Prasyarat

- Anda harus memiliki EC2 instance Amazon yang menjalankan Amazon Linux atau Ubuntu Server. Contoh ini mengasumsikan Anda sudah memiliki EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server di AWS akun Anda. Untuk meluncurkan EC2 instans Amazon, lihat Meluncurkan Mesin Virtual Linux. Di halaman Pilih Amazon Machine Image (AMI) dari wizard, pilih AMI yang nama tampilannya dimulai dengan Amazon Linux AMI atau Ubuntu Server.
- Jika EC2 instans Amazon berjalan dalam VPC Amazon, ada persyaratan tambahan. Lihat Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9.
- EC2 Instans Amazon harus memiliki setidaknya 8 hingga 16 GB ruang disk kosong yang tersedia. Sampel ini menggunakan citra Docker yang berukuran lebih dari 3 GB dan dapat menggunakan penambahan ruang disk sebesar 3 GB atau lebih untuk membangun citra. Jika Anda mencoba menjalankan sampel ini pada disk yang memiliki ruang kosong 8 GB atau kurang,

kami menemukan bahwa citra Docker mungkin tidak dibangun atau kontainer Docker mungkin tidak berjalan. Untuk memeriksa ruang disk kosong instans, Anda dapat menjalankan perintah seperti **`df -h`** (untuk "informasi sistem file disk dalam format yang dapat dibaca manusia") pada instans. Untuk meningkatkan ukuran disk instans yang ada, lihat Memodifikasi Volume di Panduan EC2 Pengguna Amazon.

## Langkah 1: Pasang dan jalankan Docker

Pada langkah ini, Anda memeriksa apakah Docker diinstal pada EC2 instance Amazon, dan menginstal Docker jika belum diinstal. Setelah Anda memasang Docker, Anda menjalankannya pada instans.

1. Connect ke EC2 instans Amazon yang sedang berjalan dengan menggunakan klien SSH seperti **`ssh`**utilitas atau PuTTY. Untuk melakukannya, lihat "Langkah 3: Hubungkan ke Instans Anda" di Meluncurkan Mesin Virtual Linux.
2. Periksa apakah Docker terpasang pada instans. Untuk melakukan ini, jalankan perintah **`docker`** pada instans dengan opsi **`--version`** .

   ```
   docker --version
   ```

   Jika Docker terpasang, versi dan nomor build Docker akan ditampilkan. Dalam hal ini, lewati ke langkah 5 nanti dalam prosedur ini.
3. Pasang Docker. Untuk melakukannya, jalankan perintah **`yum`** atau **`apt`** dengan tindakan **`install`** , menentukan paket **`docker`** atau **`docker.io`** yang akan dipasang.

   Untuk Amazon Linux:

   ```
   sudo yum install -y docker
   ```

   Untuk Ubuntu Server:

   ```
   sudo apt install -y docker.io
   ```
4. Konfirmasikan bahwa Docker terpasang. Untuk melakukan ini, jalankan perintah **`docker --version`** lagi. Versi dan nomor build Docker ditampilkan.
5. Jalankan Docker. Untuk melakukan ini, jalankan perintah **`service`** dengan layanan **`docker`** dan tindakan **`start`** .

```
sudo service docker start
```

6. Konfirmasi Docker sedang berjalan. Untuk melakukan ini, jalankan perintah **`docker`** dengan tindakan **`info`** .

   ```
   sudo docker info
   ```

   Jika Docker sedang berjalan, informasi tentang Docker akan ditampilkan.

## Langkah 2: Bangun citra

Pada langkah ini, Anda menggunakan Dockerfile untuk membangun citra Docker ke dalam instans. Sampel ini menggunakan citra yang menyertakan Node.js dan sampel aplikasi server obrolan.

1. Pada instans, buat Dockerfile. Untuk melakukan ini, dengan klien SSH masih terhubung ke instans, di direktori `/tmp` pada instans, buat file dengan nama `Dockerfile`. Sebagai contoh, jalankan perintah **`touch`** seperti berikut.

   ```
   sudo touch /tmp/Dockerfile
   ```

2. Tambahkan konten berikut ini ke file `Dockerfile`.

   ```
   # Build a Docker image based on the Amazon Linux 2 Docker image.
   FROM amazonlinux:2

   # install common tools
   RUN yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-
   latest-7.noarch.rpm
   RUN yum update -y
   RUN yum install -y sudo bash curl wget git man-db nano vim bash-completion tmux
    gcc gcc-c++ make tar

   # Enable the Docker container to communicate with AWS Cloud9 by
   # installing SSH.
   RUN yum install -y openssh-server

   # Ensure that Node.js is installed.
   RUN yum install -y nodejs
   ```

```
# Create user and enable root access
RUN useradd --uid 1000 --shell /bin/bash -m --home-dir /home/ubuntu ubuntu && \
    sed -i 's/%wheel\s.*/%wheel ALL=NOPASSWD:ALL/' /etc/sudoers && \
    usermod -a -G wheel ubuntu

# Add the AWS Cloud9 SSH public key to the Docker container.
# This assumes a file named authorized_keys containing the
# AWS Cloud9 SSH public key already exists in the same
# directory as the Dockerfile.
RUN mkdir -p /home/ubuntu/.ssh
ADD ./authorized_keys /home/ubuntu/.ssh/authorized_keys
RUN chown -R ubuntu /home/ubuntu/.ssh /home/ubuntu/.ssh/authorized_keys && \
chmod 700 /home/ubuntu/.ssh && \
chmod 600 /home/ubuntu/.ssh/authorized_keys

# Update the password to a random one for the user ubuntu.
RUN echo "ubuntu:$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 32 | head -n 1)"
 | chpasswd

# pre-install Cloud9 dependencies
USER ubuntu
RUN curl https://d2j6vhu5uywtq3.cloudfront.net/static/c9-install.sh | bash

USER root
# Start SSH in the Docker container.
CMD ssh-keygen -A && /usr/sbin/sshd -D
```

Untuk menambahkan konten sebelumnya ke file `Dockerfile`, Anda dapat menggunakan utilitas **`vi`** pada instans sebagai berikut.

a. Gunakan AWS Cloud9 untuk membuka dan mengedit `/tmp/Dockerfile` file.

```
sudo vi /tmp/Dockerfile
```

b. Tempel konten sebelumnya ke dalam file `Dockerfile`. Jika Anda tidak yakin bagaimana melakukannya, lihat dokumentasi klien SSH Anda.

c. Beralih ke mode perintah. Untuk melakukan ini, tekan tombol `Esc`. (`-- INSERT --` menghilang dari bagian bawah jendela.)

d. Ketik `:wq` (untuk menulis ke file `/tmp/Dockerfile`, simpan file, lalu keluar dari **`vi`** ), lalu tekan `Enter`.

> **Note**
>
> Anda dapat mengakses daftar gambar Docker yang sering diperbarui dari AWS CodeBuild. Untuk informasi selengkapnya, lihat gambar Docker yang disediakan oleh CodeBuild di Panduan AWS CodeBuild Pengguna.

3. Pada contoh, buat file yang berisi kunci publik AWS Cloud9 SSH untuk wadah Docker untuk digunakan. Untuk melakukan ini, di direktori yang sama dengan file `Dockerfile`, buat file dengan nama `authorized_keys`, misalnya, dengan menjalankan perintah **`touch`** .

```
sudo touch /tmp/authorized_keys
```

4. Tambahkan kunci publik AWS Cloud9 SSH ke `authorized_keys` file. Untuk mendapatkan kunci publik AWS Cloud9 SSH, lakukan hal berikut:

   a. Buka AWS Cloud9 konsol di https://console.aws.amazon.com/cloud9/.

   b. Di bilah AWS navigasi, di pemilih AWS Wilayah, pilih AWS Wilayah tempat Anda ingin membuat lingkungan AWS Cloud9 pengembangan nanti dalam topik ini.

   c. Jika halaman selamat datang ditampilkan, untuk AWS Cloud9 Lingkungan baru, pilih Buat lingkungan. Jika tidak, pilih Buat lingkungan.

   d. Pada halaman Nama lingkungan, untuk Nama, ketikkan nama untuk lingkungan. (Nama tidak penting di sini. Anda akan memilih nama lain nanti.)

   e. Pilih Langkah selanjutnya.

   f. Untuk Tipe Lingkungan, pilih Hubungkan dan jalankan di server jarak jauh (SSH).

   g. Luaskan Lihat kunci SSH publik.

   h. Pilih Tempel kunci ke clipboard. (Ini antara Lihat kunci SSH publik dan Pengaturan lanjutan.)

   i. Pilih Batalkan.

   j. Tempelkan konten clipboard ke dalam file `authorized_keys`, lalu simpan file tersebut. Misalnya, Anda dapat menggunakan utilitas **`vi`** , seperti yang dijelaskan sebelumnya dalam langkah ini.

5. Bangun citra dengan menjalankan perintah **`docker`** dengan tindakan **`build`** , tambahkan tanda `cloud9-image:latest` ke citra dan tentukan jalur ke file `Dockerfile` yang akan digunakan.

```
sudo docker build -t cloud9-image:latest /tmp
```

Jika berhasil, dua baris terakhir dari output bangunan menampilkan `Successfully built` dan `Successfully tagged`.

Untuk mengonfirmasi bahwa Docker berhasil membuat citra, jalankan perintah **`docker`** dengan tindakan `image ls`.

```
sudo docker image ls
```

Jika berhasil, output menampilkan entri di mana bidang `REPOSITORY` diatur ke `cloud9-image` dan bidang `TAG` diatur ke `latest`.

6. Catat alamat IP publik EC2 instans Amazon. Anda akan membutuhkannya untuk Langkah 4: Buat lingkungan . Jika Anda tidak yakin apa alamat IP publik dari instans, Anda dapat menjalankan perintah berikut pada instans untuk mendapatkannya.

```
curl http://169.254.169.254/latest/meta-data/public-ipv4
```

## Langkah 3: Jalankan kontainer

Pada langkah ini, Anda menjalankan kontainer Docker pada instans. Kontainer ini didasarkan pada citra yang Anda bangun pada langkah sebelumnya.

1. Untuk menjalankan kontainer Docker, jalankan perintah **`docker`** pada instans dengan tindakan **`run`** dan opsi berikut.

```
sudo docker run -d -it --expose 9090 -p 0.0.0.0:9090:22 --name cloud9 cloud9-
image:latest
```

- `-d` menjalankan kontainer dalam mode terlepas, keluar setiap kali proses root yang digunakan untuk menjalankan kontainer (dalam sampel ini, klien SSH) keluar.
- `-it` menjalankan kontainer dengan pseudo-TTY yang dialokasikan dan membuat STDIN tetap terbuka, meskipun kontainer tidak terlampir.
- `--expose` membuat port yang ditentukan (dalam sampel ini, port `9090`) tersedia dari kontainer.

- `-p`membuat port yang ditentukan tersedia secara internal ke EC2 instans Amazon melalui alamat IP dan port yang ditentukan. Dalam contoh ini, port `9090` pada wadah dapat diakses secara internal melalui port `22` pada EC2 instance Amazon.
- `--name` adalah nama kontainer yang dapat dibaca manusia (dalam sampel ini, `cloud9`).
- `cloud9-image:latest` adalah nama yang dapat dibaca manusia dari citra yang dibuat untuk digunakan untuk menjalankan kontainer.

Untuk mengonfirmasi bahwa Docker berhasil menjalankan kontainer jalankan perintah **`docker`** dengan tindakan `container ls`.

```
sudo docker container ls
```

Jika berhasil, output menampilkan entri di mana bidang `IMAGE` diatur ke `cloud9-image:latest` dan bidang `NAMES` diatur ke `cloud9`.

2. Masuk ke kontainer yang sedang berjalan. Untuk melakukannya, jalankan perintah **`docker`** dengan tindakan **`exec`** dan opsi berikut.

```
sudo docker exec -it cloud9 bash
```

- `-it` menjalankan kontainer dengan pseudo-TTY yang dialokasikan dan membuat STDIN tetap terbuka, meskipun kontainer tidak terlampir.
- `cloud9` adalah nama kontainer yang sedang berjalan yang dapat dibaca manusia.
- `bash` memulai shell standar di kontainer yang sedang berjalan.

Jika berhasil, permintaan terminal berubah untuk menampilkan nama pengguna yang masuk untuk kontainer dan ID kontainer.

> **Note**
>
> Jika Anda ingin keluar dari kontainer yang sedang berjalan, jalankan perintah **`exit`** . Prompt terminal berubah kembali untuk menampilkan nama pengguna yang masuk untuk instans dan DNS privat instans. Kontainer tetap harus berjalan.

3. Untuk direktori pada wadah yang sedang berjalan yang AWS Cloud9 ingin Anda mulai setelah masuk, atur izin aksesnya ke **`rwxr-xr-x`**. Ini berarti read-write-execute izin untuk pemilik, izin

baca-eksekusi untuk grup, dan izin baca-eksekusi untuk orang lain. Misalnya, jika jalur direktori adalah ~, Anda dapat mengatur izin ini pada direktori dengan menjalankan perintah **chmod** di kontainer yang sedang berjalan sebagai berikut.

```
sudo chmod u=rwx,g=rx,o=rx ~
```

4. Catat jalur ke direktori pada kontainer berjalan yang berisi biner Node.js, karena Anda akan membutuhkannya untuk Langkah 4: Buat lingkungan . Jika Anda tidak yakin jalur apa ini, jalankan perintah berikut pada kontainer yang sedang berjalan untuk mendapatkannya.

```
which node
```

## Langkah 4: Buat lingkungan

Pada langkah ini, Anda gunakan AWS Cloud9 untuk membuat lingkungan pengembangan AWS Cloud9 SSH dan menghubungkannya ke wadah Docker yang sedang berjalan. Setelah AWS Cloud9 menciptakan lingkungan, ini menampilkan AWS Cloud9 IDE sehingga Anda dapat mulai bekerja dengan file dan kode dalam wadah.

Anda membuat lingkungan pengembangan AWS Cloud9 SSH dengan AWS Cloud9 konsol. Anda tidak dapat membuat lingkungan SSH menggunakan CLI.

### Prasyarat

- Pastikan Anda menyelesaikan langkah-langkahnya terlebih Menyiapkan AWS Cloud9 dahulu. Dengan begitu, Anda dapat masuk ke AWS Cloud9 konsol dan membuat lingkungan.
- Identifikasi instance komputasi cloud yang ada (misalnya, EC2 instans Amazon di server Anda Akun AWS) atau server Anda sendiri yang ingin Anda sambungkan AWS Cloud9 ke lingkungan.
- Pastikan instans yang ada atau server Anda sendiri memenuhi semua Persyaratan hosting SSH. Ini termasuk memiliki versi tertentu dari Python, Node.js dan komponen lain yang sudah diinstal, mengatur izin khusus pada direktori yang Anda inginkan untuk memulai AWS Cloud9 setelah login, dan menyiapkan Amazon Virtual Private Cloud terkait.

### Membuat Lingkungan SSH

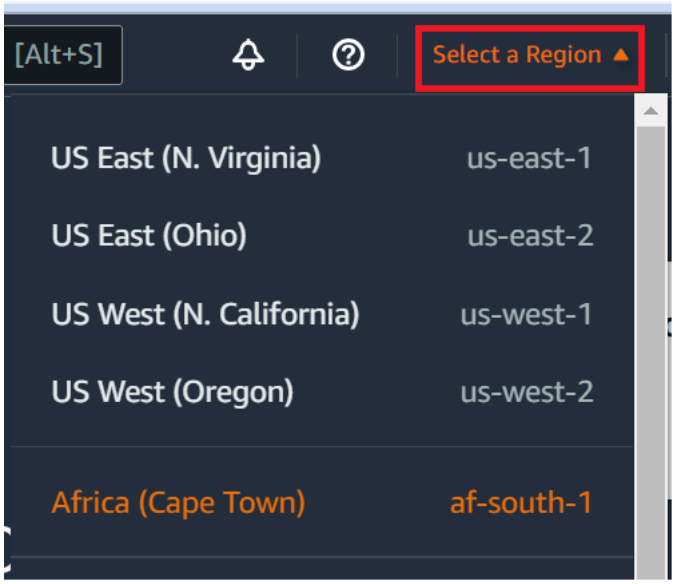
1. Pastikan Anda menyelesaikan prasyarat sebelumnya.

2. Hubungkan ke instans yang ada atau server Anda sendiri dengan menggunakan klien SSH, jika Anda belum terhubung dengannya. Ini memastikan bahwa Anda dapat menambahkan nilai kunci SSH publik yang diperlukan ke instance atau server. Ini dijelaskan kemudian dalam prosedur ini.

   > **Note**
   >
   > Untuk menyambung ke instans AWS Cloud komputasi yang ada, lihat satu atau beberapa sumber daya berikut:
   >
   > - Untuk Amazon EC2, lihat Connect to Your Linux Instance di Panduan EC2 Pengguna Amazon.
   > - Untuk Amazon Lightsail, lihat Menghubungkan ke instans Lightsail berbasis Linux/Unix Anda di Dokumentasi Amazon Lightsail.
   > - Untuk AWS Elastic Beanstalk, lihat Daftar dan Menghubungkan ke Instans Server di Panduan AWS Elastic Beanstalk Pengembang.
   > - Untuk AWS OpsWorks, lihat Menggunakan SSH untuk Masuk ke Instance Linux di Panduan AWS OpsWorks Pengguna.
   > - Untuk yang lain Layanan AWS, lihat dokumentasi untuk layanan khusus itu.
   >
   > Untuk terhubung ke server Anda sendiri, gunakan SSH. SSH sudah diinstal pada sistem operasi macOS dan Linux. Untuk terhubung ke server Anda dengan menggunakan SSH pada Windows, Anda harus menginstal PuTTY.

3. Masuk ke AWS Cloud9 konsol, di https://console.aws.amazon.com/cloud9/.

4. Setelah Anda masuk ke AWS Cloud9 konsol, di bilah navigasi atas pilih Wilayah AWS untuk membuat lingkungan. Untuk daftar yang tersedia Wilayah AWS, lihat AWS Cloud9di Referensi Umum AWS.




5. Jika ini adalah pertama kalinya Anda membuat lingkungan pengembangan, halaman selamat datang akan ditampilkan. Di panel  AWS Cloud9 lingkungan baru, pilih Buat lingkungan.

   Jika sebelumnya Anda telah membuat lingkungan pengembangan, Anda juga dapat memperluas panel di sebelah kiri layar. Pilih lingkungan Anda, lalu pilih Buat lingkungan.

   Di halaman selamat datang:

   Atau di halaman Lingkungan Anda:

6. Pada halaman Buat lingkungan, masukkan nama untuk lingkungan Anda.
7. Untuk Deskripsi, memasukkan sesuatu tentang lingkungan Anda. Untuk tutorial ini, gunakan `This environment is for the AWS Cloud9 tutorial.`
8. Untuk jenis Lingkungan, pilih Komputasi yang Ada dari opsi berikut:

   - EC2 Instans baru — Meluncurkan EC2 instans Amazon yang AWS Cloud9 dapat terhubung langsung melalui SSH.

- Komputasi yang ada — Meluncurkan EC2 instans Amazon yang tidak memerlukan port masuk terbuka. AWS Cloud9 terhubung ke instance melalui AWS Systems Manager.
  - Jika Anda memilih opsi Komputasi yang ada, peran layanan dan profil instans IAM akan dibuat untuk memungkinkan Systems Manager berinteraksi dengan EC2 instans atas nama Anda. Anda dapat melihat nama-nama keduanya di bagian Peran layanan dan profil instans untuk akses Systems Manager lebih jauh ke bawah antarmuka. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager.

> **Warning**
>
> Membuat EC2 instance untuk lingkungan Anda dapat mengakibatkan biaya yang mungkin Akun AWS untuk Amazon Anda EC2. Tidak ada biaya tambahan untuk menggunakan Systems Manager untuk mengelola koneksi ke EC2 instans Anda.

> **Warning**
>
> AWS Cloud9 menggunakan kunci publik SSH untuk terhubung dengan aman ke server Anda. Untuk membuat koneksi aman, tambahkan kunci publik kami ke `~/.ssh/authorized_keys` file Anda dan berikan kredensi login Anda dalam langkah-langkah berikut. Pilih Salin kunci ke clipboard untuk menyalin kunci SSH, atau Lihat kunci SSH publik untuk menampilkannya.

9. Pada panel komputasi yang ada, untuk Pengguna, masukkan nama login yang Anda gunakan untuk terhubung ke instance atau server sebelumnya dalam prosedur ini. Misalnya, untuk contoh AWS Cloud komputasi, mungkin`ec2-user`,`ubuntu`, atau`root`.

> **Note**
>
> Kami menyarankan agar nama login dikaitkan dengan izin administratif atau pengguna administrator pada instance atau server. Lebih khusus lagi, kami menyarankan agar nama login ini memiliki instalasi Node.js pada instance atau server. Untuk memeriksa ini, dari terminal instance atau server Anda, jalankan perintah **`ls -l $(which node)`**(atau **`ls -l $(nvm which node)`**jika Anda menggunakan`nvm`). Perintah ini

menampilkan nama pemilik instalasi Node.js. Ini juga menampilkan izin instalasi, nama grup, dan lokasi.

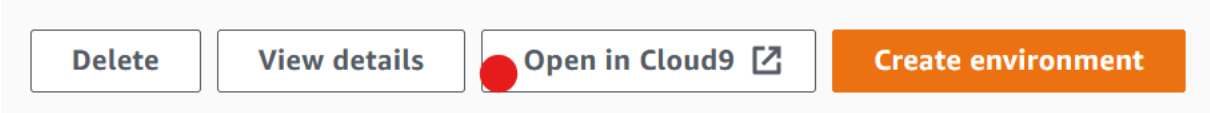
10. Untuk Host, masukkan alamat IP publik (lebih disukai) atau nama host dari instance atau server.
11. Untuk Port, masukkan port yang AWS Cloud9 ingin Anda gunakan untuk mencoba terhubung ke instance atau server. Atau, pertahankan port default.
12. Pilih Detail tambahan - opsional untuk menampilkan jalur lingkungan, jalur ke biner node.js dan informasi host lompat SSH.
13. Untuk jalur Lingkungan, masukkan jalur ke direktori pada instance atau server yang AWS Cloud9 ingin Anda mulai. Anda mengidentifikasi ini sebelumnya di prasyarat untuk prosedur ini. Jika Anda membiarkannya kosong, AWS Cloud9 menggunakan direktori tempat instans atau server Anda biasanya memulai setelah login. Ini biasanya merupakan direktori beranda atau default.
14. Untuk jalur biner Path to Node.js, masukkan informasi jalur untuk menentukan jalur ke biner Node.js pada instance atau server. Untuk mendapatkan jalur, Anda dapat menjalankan perintah **`which node`**(atau **`nvm which node`**jika Anda menggunakan`nvm`) pada instance atau server Anda. Misalnya, jalurnya mungkin `/usr/bin/node`. Jika Anda membiarkannya kosong, AWS Cloud9 mencoba menebak di mana biner Node.js saat mencoba terhubung.
15. Untuk host lompat SSH, masukkan informasi tentang host lompat yang digunakan instance atau server. Gunakan format `USER_NAME@HOSTNAME:PORT_NUMBER` (misalnya,`ec2-user@:ip-192-0-2-0:22`).

    Host lompat harus memenuhi persyaratan berikut:

    - Jump host harus dapat dijangkau melalui internet publik menggunakan SSH.
    - Jump host harus memungkinkan akses masuk oleh alamat IP apa pun melalui port tertentu.
    - Nilai kunci SSH publik yang disalin ke file `~/.ssh/authorized_keys` pada instans atau server yang ada juga harus disalin ke file `~/.ssh/authorized_keys` pada jump host.
    - Netcat harus diinstal.
16. Menambahkan hingga 50 tanda dengan menyediakan Kunci dan sebuah Nilai untuk setiap tanda. Lakukan dengan memilih Tambahkan tag baru. Tag dilampirkan ke AWS Cloud9 lingkungan sebagai tag sumber daya, dan disebarkan ke sumber daya dasar berikut: AWS CloudFormation tumpukan, EC2 instans Amazon, dan grup EC2 keamanan Amazon. Untuk mempelajari lebih lanjut tentang tag, lihat Mengontrol Akses Menggunakan Tag AWS Sumber Daya di Panduan Pengguna IAM dan informasi lanjutan tentang tag dalam panduan ini.

> **Warning**
>
> Jika Anda memperbarui tag ini setelah Anda membuatnya, perubahan tidak disebarkan ke sumber daya yang mendasarinya. Untuk informasi selengkapnya, lihat Menyebarkan pembaruan tanda ke sumber daya yang mendasari dalam informasi lanjutan tentang tanda.

17. Pilih Buat untuk membuat lingkungan Anda, dan Anda kemudian diarahkan ke halaman beranda. Ketika akun berhasil dibuat, bilah flash hijau muncul di bagian atas AWS Cloud9 konsol. Anda dapat memilih lingkungan baru dan memilih Buka di Cloud9 untuk meluncurkan IDE.




Jika akun gagal dibuat, bilah flash merah muncul di bagian atas AWS Cloud9 konsol. Akun Anda mungkin gagal dibuat karena masalah dengan browser web Anda, izin AWS akses Anda, instans, atau jaringan terkait. Anda dapat menemukan informasi tentang kemungkinan perbaikan pada masalah yang mungkin menyebabkan akun gagal di bagian AWS Cloud9 Pemecahan Masalah.

> **Note**
>
> Jika lingkungan Anda menggunakan proxy untuk mengakses internet, Anda harus memberikan detail proxy AWS Cloud9 agar dapat menginstal dependensi. Untuk informasi selengkapnya, lihat Gagal menginstal dependensi.

## Langkah 5: Jalankan kode

Pada langkah ini, Anda menggunakan AWS Cloud9 IDE untuk menjalankan aplikasi sampel di dalam wadah Docker yang sedang berjalan.

1. Dengan AWS Cloud9 IDE ditampilkan untuk wadah yang sedang berjalan, mulai server obrolan sampel. Untuk melakukannya, di jendela Lingkungan, klik kanan file sampel `workspace/server.js`, lalu pilih Jalankan.

2. Pratinjau aplikasi sampel. Untuk melakukan ini, di jendela Lingkungan, buka file `workspace/client/index.html`. Kemudian, pada bilah menu, pilih Alat, Pratinjau, Pratinjau Aplikasi yang Berjalan.
3. Pada tab pratinjau aplikasi, untuk Nama Anda, ketikkan nama Anda. Untuk Pesan, ketik pesan. Lalu, pilih Kirim. Server obrolan menambahkan nama dan pesan Anda ke daftar.

## Langkah 6: Membersihkan

Pada langkah ini, Anda menghapus lingkungan AWS Cloud9 dan menghapus serta file dukungan Docker dari EC2 instance Amazon. Selain itu, untuk mencegah tagihan berkelanjutan ke AWS akun Anda setelah selesai menggunakan sampel ini, Anda harus menghentikan EC2 instance Amazon yang menjalankan Docker.

### Langkah 6.1: Hapus lingkungan

Untuk menghapus lingkungan, lihat Menghapus lingkungan di AWS Cloud9.

### Langkah 6.2: Hapus file dukungan AWS Cloud9 dari kontainer

Setelah Anda menghapus lingkungan, beberapa file AWS Cloud9 dukungan masih tetap berada di wadah. Jika Anda ingin tetap menggunakan wadah tetapi tidak lagi membutuhkan file dukungan ini, hapus `.c9` folder dari direktori pada wadah yang Anda tentukan AWS Cloud9 untuk memulai setelah masuk. Misalnya, jika direktorinya adalah ~, jalankan perintah **`rm`** dengan opsi **`-r`** sebagai berikut.

```
sudo rm -r ~/.c9
```

### Langkah 6.3: Hapus file dukungan Docker dari instans

Jika Anda tidak lagi ingin menyimpan wadah Docker, image Docker, dan Docker di EC2 instance Amazon, tetapi Anda ingin menyimpan instance, Anda dapat menghapus file dukungan Docker ini sebagai berikut.

1. Hapus kontainer Docker dari instans. Untuk melakukannya, jalankan perintah **`docker`** pada instans dengan tindakan penghentian **`stop`** dan **`rm`** dan nama kontainer yang dapat dibaca manusia.

   ```
   sudo docker stop cloud9
   sudo docker rm cloud9
   ```

2. Hapus citra Docker dari instans. Untuk melakukannya, jalankan perintah **`docker`** pada instans dengan tindakan **`image rm`** dan tanda citra.

```
sudo docker image rm cloud9-image:latest
```

3. Hapus file dukungan Docker tambahan yang mungkin masih keluar. Untuk melakukan ini, jalankan perintah **`docker`** pada instans dengan tindakan **`system prune`** .

```
sudo docker system prune -a
```

4. Copot pemasangan Docker. Untuk melakukannya, jalankan perintah **`yum`** pada instans dengan tindakan **`remove`** , menentukan paket **`docker`** yang akan dihapus pemasangannya.

   Untuk Amazon Linux:

```
sudo yum -y remove docker
```

   Untuk Ubuntu Server:

```
sudo apt -y remove docker
```

   Anda juga dapat menghapus file `Dockerfile` dan `authorized_keys` yang Anda buat sebelumnya. Sebagai contoh, jalankan perintah **`rm`** pada instans.

```
sudo rm /tmp/Dockerfile
sudo rm /tmp/authorized_keys
```

### Langkah 6.4: Akhiri instans

Untuk menghentikan EC2 instans Amazon, lihat Menghentikan Instans Anda di  EC2 Panduan Pengguna Amazon.

## Tutorial Terkait

- Memulai dengan AWS RoboMaker di Panduan AWS RoboMaker Pengembang. Tutorial ini digunakan AWS Cloud9 untuk memodifikasi, membangun, dan bundel contoh aplikasi robot.

# Topik lanjutan untuk AWS Cloud9

Topik ini berisi informasi berikut:

- Informasi yang digunakan untuk konfigurasi lanjutan dan pengambilan keputusan.
- Informasi yang terkait dengan tugas tertentu dan dapat memberi Anda pemahaman yang lebih baik AWS Cloud9 tetapi tidak penting untuk menyelesaikan tugas itu.

Topik


- EC2 lingkungan dibandingkan dengan lingkungan SSH di AWS Cloud9
- Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9
- Persyaratan host lingkungan SSH
- Menggunakan AWS Cloud9 Installer untuk lingkungan AWS Cloud9 SSH
- Rentang alamat IP SSH masuk untuk AWS Cloud9
- Konten AMI untuk Lingkungan AWS Cloud9 EC2 Pengembangan
- Menggunakan peran tertaut layanan untuk AWS Cloud9
- Pencatatan panggilan AWS Cloud9 API dengan AWS CloudTrail
- Tanda


## EC2 lingkungan dibandingkan dengan lingkungan SSH di AWS Cloud9

Seperti yang dibahas dalam pendahuluan untuk lingkungan dan sumber daya komputasi dan bekerja dengan AWS Cloud9 lingkungan, lingkungan Anda dapat diatur sebagai EC2 lingkungan atau lingkungan SSH.

Tabel berikut menyoroti persamaan dan perbedaan antara menggunakan EC2 lingkungan dan lingkungan SSH di. AWS Cloud9

| EC2 lingkungan | Lingkungan SSH |
| --- | --- |
| AWS Cloud9 membuat EC2 instance Amazon terkait dan mengelola siklus hidup instance. | Anda menggunakan instans komputasi cloud yang ada atau server Anda sendiri. Anda |

| EC2 lingkungan | Lingkungan SSH |
| --- | --- |
| Ini termasuk memulai, menghentikan, dan menghentikan operasi. | bertanggung jawab untuk mengelola siklus hidupnya. |
| Instans berjalan di Amazon Linux atau Ubuntu Peladen. | Anda dapat menggunakan instans komputasi cloud apa pun yang menjalankan Linux, atau Anda dapat menggunakan server Anda sendiri yang menjalankan Linux. |
| AWS Cloud9 secara otomatis mengatur instance untuk mulai bekerja dengan AWS Cloud9. | Anda harus secara manual mengonfigurasi instans atau server Anda sendiri untuk bekerja dengan AWS Cloud9. |
| AWS Cloud9 secara otomatis mengatur AWS Command Line Interface (AWS CLI) pada instance. | Jika Anda ingin menggunakan AWS CLI pada instance atau server Anda sendiri, Anda bertanggung jawab untuk mengaturnya sendiri. |
| Instance ini memiliki akses ke ratusan paket yang berguna, dengan beberapa paket umum sudah diinstal dan dikonfigurasi. Contohnya adalah Git, Docker, Node.js, dan Python. | Anda mungkin perlu mengunduh, menginstal, dan mengonfigurasi paket tambahan untuk menyelesaikan tugas umum. |
| Anda mempertahankan contoh, misalnya, dengan menerapkan pembaruan sistem secara berkala. | Anda memelihara instans atau server Anda sendiri. |
| Saat Anda menghapus lingkungan, AWS Cloud9 secara otomatis menghentikan instance terkait. | Saat Anda menghapus lingkungan, instans atau server Anda sendiri tetap ada. |

| EC2 lingkungan | Lingkungan SSH |
| --- | --- |
| AWS kredensial sementara terkelola tersedia di EC2 lingkungan. Dengan kredensi ini, Anda dapat, dengan beberapa batasan, mengaktif kan atau menonaktifkan semua AWS tindakan untuk semua AWS sumber daya di pemanggil . Akun AWS Anda tidak perlu mengonfigurasi profil instans untuk EC2 instans Amazon lingkungan Anda atau menyimpan kredenal AWS akses permanen AWS entitas, seperti pengguna IAM. Jika EC2 instans Amazon untuk lingkungan Anda diluncurkan ke subnet pribadi, Anda tidak dapat menggunakan kredensil sementara AWS terkelola untuk mengizinkan EC2 lingkungan Amazon mengakses AWS layanan atas nama AWS entitas, misalnya pengguna IAM. | AWS kredensial sementara terkelola tidak tersedia di lingkungan SSH. Anda harus menggunakan AWS Identity and Access Managementuntuk mengelola izin yang memungkinkan Anda untuk bekerja dengan keduanya AWS Cloud9 Layanan AWS dan sumber daya lainnya. |
| AWS Toolkit, panel Git, dan dukungan Java yang disempurnakan tersedia untuk digunakan. | AWS Toolkit, panel Git, dan disempurnakan Java dukungan tidak tersedia. |

# Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9

Setiap lingkungan AWS Cloud9 pengembangan yang terkait dengan Amazon Virtual Private Cloud (Amazon VPC) harus memenuhi persyaratan VPC tertentu. Lingkungan ini mencakup EC2 lingkungan, dan lingkungan SSH yang terkait dengan instance AWS Cloud komputasi yang berjalan dalam VPC. Contohnya termasuk instans Amazon EC2 dan Amazon Lightsail.

## Persyaratan Amazon VPC untuk AWS Cloud9

VPC Amazon yang AWS Cloud9 menggunakan memerlukan pengaturan berikut. Jika Anda sudah terbiasa dengan persyaratan ini dan hanya ingin membuat VPC yang kompatibel, lewati ke Buat VPC plus sumber daya VPC lainnya.

Gunakan daftar periksa berikut untuk mengonfirmasi bahwa VPC memenuhi semua persyaratan berikut:

- VPC bisa sama Akun AWS dan Wilayah AWS sebagai lingkungan AWS Cloud9 pengembangan atau VPC dapat menjadi VPC bersama yang berbeda dari lingkungan. Akun AWS Namun, VPC harus Wilayah AWS sama dengan lingkungan. Untuk informasi selengkapnya tentang Amazon VPCs Wilayah AWS, lihatLihat daftar VPCs untuk Wilayah AWS. Untuk petunjuk selengkapnya tentang cara membuat VPC Amazon AWS Cloud9, lihat. Buat VPC plus sumber daya VPC lainnya Untuk informasi tentang bekerja dengan Amazon bersama VPCs, lihat Bekerja dengan dibagikan VPCs di Panduan Pengguna Amazon VPC.
- VPC harus memiliki subnet publik. Subnet bersifat publik jika lalu lintasnya diarahkan ke gateway internet. Untuk daftar subnet untuk VPC Amazon, lihat. Melihat daftar subnet untuk VPC
- Jika lingkungan Anda mengakses EC2 instance-nya secara langsung melalui SSH, instance dapat diluncurkan ke subnet publik saja. Untuk informasi tentang mengonfirmasi apakah subnet bersifat publik, lihatMengonfirmasi apakah subnet bersifat publik.
- Jika Anda mengakses instans EC2 Amazon tanpa masuk menggunakan Systems Manager, instans dapat diluncurkan ke subnet publik atau pribadi.
- Jika Anda menggunakan subnet publik, lampirkan gateway internet ke VPC. Ini begitu AWS Systems Manager Agent (SSM Agent) misalnya dapat terhubung ke Systems Manager.
- Jika Anda menggunakan subnet privat, izinkan instans untuk subnet berkomunikasi dengan internet dengan meng-host gateway NAT di subnet publik. Untuk informasi selengkapnya tentang melihat atau mengubah setelan untuk gateway internet, lihat Melihat atau mengubah pengaturan untuk gateway internet
- Subnet publik harus memiliki tabel rute dengan seperangkat rute minimum. Untuk mempelajari cara mengonfirmasi apakah subnet memiliki tabel rute, lihatMengonfirmasi apakah subnet memiliki tabel rute. Untuk informasi tentang cara membuat tabel rute, lihatMembuat tabel rute.
- Grup keamanan terkait untuk VPC (atau untuk instance AWS Cloud komputasi, tergantung pada arsitektur Anda) harus mengizinkan set minimum lalu lintas masuk dan keluar. Untuk daftar grup keamanan untuk VPC Amazon, lihat. Melihat daftar grup keamanan untuk VPC Untuk informasi selengkapnya tentang membuat grup keamanan di VPC Amazon, lihat. Buat grup keamanan di VPC
- Untuk lapisan tambahan keamanan, jika VPC memiliki ACL jaringan, ACL jaringan harus mengizinkan set minimal masuk dan keluar lalu lintas. Untuk mengonfirmasi apakah VPC Amazon memiliki setidaknya satu jaringan ACL, lihat. Mengonfirmasi apakah VPC memiliki setidaknya satu ACL jaringan Untuk informasi tentang membuat ACL jaringan, lihatBuat ACL jaringan.
- Jika lingkungan pengembangan Anda menggunakan SSM untuk mengakses EC2 instance, pastikan instans tersebut diberi alamat IP publik oleh subnet publik yang diluncurkan. Untuk melakukannya, Anda harus mengaktifkan penetapan otomatis opsi alamat IP publik untuk subnet

publik, dan mengaturnya ke. `Yes` Anda dapat mengaktifkan ini di subnet publik sebelum membuat AWS Cloud9 lingkungan dalam halaman pengaturan subnet. Untuk langkah-langkah yang terlibat dalam memodifikasi pengaturan IP penetapan otomatis di subnet publik, lihat Memodifikasi atribut IPv4 pengalamatan publik untuk subnet Anda di Panduan Pengguna Amazon VPC. Untuk informasi selengkapnya tentang mengonfigurasi subnet publik dan pribadi, lihat Mengonfigurasi subnet sebagai publik atau subnet privat

> **Note**
>
> Untuk prosedur berikut, masuk ke AWS Management Console dan gunakan kredensil administrator untuk membuka konsol VPC Amazon (/vpc) atau konsol Amazon EC2 (https://console.aws.amazon.com/ec2). https://console.aws.amazon.com
> Jika Anda menggunakan AWS CLI atau AWS CloudShell, kami sarankan Anda mengonfigurasi AWS CLI atau AWS CloudShell dengan kredensi untuk administrator di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

## Lihat daftar VPCs untuk Wilayah AWS

Untuk menggunakan konsol VPC Amazon, di bilah AWS navigasi, pilih Wilayah AWS yang AWS Cloud9 menciptakan lingkungan di dalamnya. Kemudian, pilih Your VPCs di panel navigasi.

Untuk menggunakan AWS CLI atau AWS CloudShell, jalankan EC2 **`describe-vpcs`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-vpcs --output table --query 'Vpcs[*].VpcId' --region us-east-2
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang AWS Cloud9 menciptakan lingkungan di. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Outputnya berisi daftar VPC IDs.

## Melihat daftar subnet untuk VPC

Untuk menggunakan konsol Amazon VPC, pilih Your VPCs di panel navigasi. Catat ID VPC di kolom ID VPC. Kemudian memilih Subnet di panel navigasi, dan mencari subnet yang berisi ID di kolom VPC.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`describe-subnets`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-subnets --output table --query 'Subnets[*].[SubnetId,VpcId]' --region
 us-east-2
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi subnet. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Dalam output, cari subnet yang sesuai dengan ID VPC.

## Mengonfirmasi apakah subnet bersifat publik

> **Important**
>
> Misalkan Anda meluncurkan EC2 instance lingkungan Anda ke subnet pribadi. Pastikan lalu lintas keluar diizinkan untuk contoh itu sehingga dapat terhubung ke layanan SSM. Untuk subnet privat, lalu lintas keluar biasanya dikonfigurasi melalui gateway terjemahan alamat jaringan (NAT) atau VPC endpoint. (Gateway NAT membutuhkan subnet publik.) Misalkan Anda memilih titik akhir VPC alih-alih gateway NAT untuk mengakses SSM. Pembaruan otomatis dan patch keamanan untuk instans Anda mungkin tidak berfungsi jika bergantung pada akses internet. Anda dapat menggunakan aplikasi lain, seperti AWS Systems Manager Patch Manager, untuk mengelola pembaruan perangkat lunak apa pun yang mungkin diperlukan oleh lingkungan Anda. AWS Cloud9 Perangkat lunak akan diperbarui seperti biasa.

Untuk menggunakan konsol Amazon VPC, memilih Subnet di panel navigasi. Pilih kotak di sebelah subnet yang AWS Cloud9 ingin Anda gunakan. Pada tab Tabel Rute, jika ada entri di kolom Target yang dimulai dengan igw-, subnet adalah publik.

Untuk menggunakan AWS CLI atauaws-shell, jalankan EC2 **describe-route-tables**perintah Amazon.

```
aws ec2 describe-route-tables --output table --query 'RouteTables[*].Routes[*].
{GatewayIds:GatewayId}' --region us-east-2 --filters Name=association.subnet-
id,Values=subnet-12a3456b
```

Pada perintah sebelumnya, ganti us-east-2 dengan Wilayah AWS yang berisi subnet, dan ganti subnet-12a3456b dengan subnet ID. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya denganaws-shell, hilangkan. aws

Dalam output, jika ada setidaknya satu hasil yang dimulai dengan igw-, subnet adalah publik.

Dalam output, jika tidak ada hasil, tabel rute mungkin terkait dengan VPC alih-alih dengan subnet. Untuk mengonfirmasi ini, jalankan EC2 **describe-route-tables**perintah Amazon untuk VPC yang terkait dengan subnet alih-alih subnet itu sendiri, misalnya, sebagai berikut.

```
aws ec2 describe-route-tables --output table --query 'RouteTables[*].Routes[*].
{GatewayIds:GatewayId}' --region us-east-1 --filters Name=vpc-id,Values=vpc-1234ab56
```

Pada perintah sebelumnya, ganti us-east-2 dengan Wilayah AWS yang berisi VPC, dan ganti dengan vpc-1234ab56 ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya denganaws-shell, hilangkan. aws

Dalam output, jika ada setidaknya satu hasil yang dimulai dengan igw-, VPC berisi gateway internet.

### Melihat atau mengubah pengaturan untuk gateway internet

Untuk menggunakan konsol Amazon VPC, memilih Gateway Internet di panel navigasi. Pilih kotak di samping gateway internet. Untuk melihat pengaturan, lihat setiap tab. Untuk mengubah pengaturan pada tab, memilih Edit jika berlaku, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau aws-shell untuk melihat pengaturan, jalankan EC2 **describe-internet-gateways**perintah Amazon.

```
aws ec2 describe-internet-gateways --output table --region us-east-2 --internet-
gateway-id igw-1234ab5c
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi gateway internet, dan ganti `igw-1234ab5c` dengan ID gateway internet. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

### Membuat gateway internet baru

Untuk menggunakan konsol Amazon VPC, memilih Gateway Internet di panel navigasi. Memilih Buat gateway internet, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`create-internet-gateway`**perintah Amazon.

```
aws ec2 create-internet-gateway --output text --query
 'InternetGateway.InternetGatewayId' --region us-east-2
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi gateway internet baru. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Output berisi ID dari gateway internet baru.

### Pasangkan gateway internet ke VPC

Untuk menggunakan konsol Amazon VPC, memilih Gateway Internet di panel navigasi. Pilih kotak di samping gateway internet. Memilih Tindakan, Lampirkan ke VPC jika tersedia, dan kemudian ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`attach-internet-gateway`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 attach-internet-gateway --region us-east-2 --internet-gateway-id igw-a1b2cdef
 --vpc-id vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi gateway internet. Ganti `igw-a1b2cdef` dengan ID gateway internet. Dan ganti `vpc-1234ab56` dengan ID VPC. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Mengonfirmasi apakah subnet memiliki tabel rute

Untuk menggunakan konsol Amazon VPC, memilih Subnet di panel navigasi. Pilih kotak di sebelah subnet publik untuk VPC yang AWS Cloud9 ingin Anda gunakan. Pada tab Tabel rute, jika ada nilai untuk Tabel Rute, subnet publik memiliki tabel rute.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`describe-route-tables`**perintah Amazon.

```
aws ec2 describe-route-tables --output table --query 'RouteTables[*].Associations[*].
{RouteTableIds:RouteTableId}' --region us-east-2 --filters Name=association.subnet-
id,Values=subnet-12a3456b
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi subnet publik, dan ganti `subnet-12a3456b` dengan ID subnet publik. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Jika ada nilai-nilai dalam output, subnet publik memiliki setidaknya satu tabel rute.

Dalam output, jika tidak ada hasil, tabel rute mungkin terkait dengan VPC alih-alih dengan subnet. Untuk mengonfirmasi hal ini, jalankan EC2 **`describe-route-tables`**perintah Amazon untuk VPC terkait subnet alih-alih subnet itu sendiri, misalnya, sebagai berikut.

```
aws ec2 describe-route-tables --output table --query 'RouteTables[*].Associations[*].
{RouteTableIds:RouteTableId}' --region us-east-2 --filters Name=vpc-
id,Values=vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi VPC, dan ganti dengan `vpc-1234ab56` ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Dalam output, jika ada setidaknya satu hasil, VPC memiliki setidaknya satu tabel rute.

## Melampirkan tabel rute ke subnet

Untuk menggunakan konsol Amazon VPC, memilih Tabel Rute di panel navigasi. Pilih kotak di samping tabel rute yang ingin Anda lampirkan. Pada tab Asosiasi Subnet, memilih Edit, pilih kotak di samping subnet yang ingin Anda lampirkan, lalu memilih Simpan.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`associate-route-table`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 associate-route-table --region us-east-2 --subnet-id subnet-12a3456b --route-table-id rtb-ab12cde3
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi tabel rute. Ganti `subnet-12a3456b` dengan subnet ID. Dan ganti `rtb-ab12cde3` dengan ID tabel rute. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

### Membuat tabel rute

Untuk menggunakan konsol Amazon VPC, memilih Tabel Rute di panel navigasi. Memilih Buat Tabel Rute, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`create-route-table`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 create-route-table --output text --query 'RouteTable.RouteTableId' --region us-east-2 --vpc-id vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi tabel rute baru, dan ganti `vpc-1234ab56` dengan ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Output berisi ID dari tabel rute baru.

### Melihat atau mengubah pengaturan untuk tabel rute

Untuk menggunakan konsol Amazon VPC, memilih Tabel Rute di panel navigasi. Pilih kotak di samping tabel rute. Untuk melihat pengaturan, lihat setiap tab. Untuk mengubah pengaturan pada tab, memilih Edit, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau `aws-shell` untuk melihat pengaturan, jalankan EC2 **`describe-route-tables`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-route-tables --output table --region us-east-2 --route-table-ids rtb-ab12cde3
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi tabel rute, dan ganti `rtb-ab12cde3` dengan ID tabel rute. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Pengaturan tabel rute minimum yang disarankan untuk AWS Cloud9

| Tujuan | Target | Status | Disebarkan |
|---|---|---|---|
| BLOK-CIDR | lokal | Aktif | Tidak |
| 0.0.0.0/0 | `igw-INTERNET-GATEWAY-ID` | Aktif | Tidak |

Dalam pengaturan ini, *`CIDR-BLOCK`* adalah blok CIDR untuk subnet, dan `igw-`*`INTERNET-GATEWAY-ID`* adalah ID dari gateway internet yang kompatibel.

## Melihat daftar grup keamanan untuk VPC

Untuk menggunakan konsol Amazon VPC, memilih Grup Keamanan di panel navigasi. Di kotak Cari Grup Keamanan, memasukkan VPC ID atau nama, lalu tekan `Enter`. Grup keamanan untuk VPC tersebut muncul dalam daftar hasil pencarian.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`describe-security-groups`**perintah Amazon.

```
aws ec2 describe-security-groups --output table --query 'SecurityGroups[*].GroupId' --
region us-east-2 --filters Name=vpc-id,Values=vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi VPC, dan ganti dengan `vpc-1234ab56` ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Outputnya berisi daftar grup keamanan IDs untuk VPC itu.

## Melihat daftar grup keamanan untuk instance AWS Cloud komputasi

Untuk menggunakan EC2 konsol Amazon, perluas Instans di panel navigasi, lalu pilih Instans. Dalam daftar instans, memilih kotak di samping instans. Grup keamanan untuk instans itu muncul di tab Deskripsi di sebelah Grup keamanan.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`describe-security-groups`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-instances --output table --query
 'Reservations[*].Instances[*].NetworkInterfaces[*].Groups[*].GroupId' --region us-
east-2 --instance-ids i-12a3c456d789e0123
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi instance, dan ganti `i-12a3c456d789e0123` dengan ID instance. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

Output berisi daftar grup keamanan IDs untuk contoh itu.

## Melihat atau mengubah pengaturan untuk grup keamanan di VPC

Untuk menggunakan konsol Amazon VPC, memilih Grup Keamanan di panel navigasi. Pilih kotak centang di samping grup keamanan. Untuk melihat pengaturan, lihat setiap tab. Untuk mengubah pengaturan pada tab, memilih Edit jika berlaku, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau `aws-shell` untuk melihat pengaturan, jalankan EC2 **`describe-security-groups`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-security-groups --output table --region us-east-2 --group-ids
 sg-12a3b456
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi instance, dan ganti `sg-12a3b456` dengan ID grup keamanan. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Melihat atau mengubah setelan untuk grup keamanan instans AWS Cloud komputasi

Untuk menggunakan EC2 konsol Amazon, perluas Instans di panel navigasi, lalu pilih Instans. Dalam daftar instans, pilih kotak di samping instans. Di tab Deskripsi untuk, untuk Grup keamanan, memilih

grup keamanan. Lihat masing-masing tab. Untuk mengubah pengaturan pada tab, memilih Edit jika berlaku, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau `aws-shell` untuk melihat pengaturan, jalankan EC2 **`describe-security-groups`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 describe-security-groups --output table --region us-east-2 --group-ids
 sg-12a3b456
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi instance, dan ganti `sg-12a3b456` dengan ID grup keamanan. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Pengaturan lalu lintas masuk dan keluar minimum untuk AWS Cloud9

> **Important**
>
> Grup keamanan IA untuk sebuah instance mungkin tidak memiliki aturan masuk. Jika ini terjadi, ini berarti tidak ada lalu lintas masuk yang berasal dari host lain ke instance yang diizinkan. Untuk informasi tentang menggunakan EC2 instance no-ingress, lihat. Mengakses instance no-ingress EC2 dengan AWS Systems Manager

- Masuk: Semua alamat IP menggunakan SSH melalui port 22. Namun, Anda dapat membatasi alamat IP ini hanya untuk yang AWS Cloud9 menggunakan. Untuk informasi selengkapnya, lihat Rentang alamat IP SSH masuk untuk AWS Cloud9.

  > **Note**
  >
  > Untuk EC2 lingkungan yang dibuat pada atau setelah 31 Juli 2018, AWS Cloud9 menggunakan grup keamanan untuk membatasi alamat IP masuk menggunakan SSH melalui port 22. Alamat IP masuk ini secara khusus hanya alamat yang AWS Cloud9 digunakan. Untuk informasi selengkapnya, lihat Rentang alamat IP SSH masuk untuk AWS Cloud9.

- Inbound ( ACLs hanya jaringan): Untuk EC2 lingkungan dan lingkungan SSH yang terkait dengan EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server, semua alamat IP menggunakan TCP melalui port 32768-61000. Untuk informasi selengkapnya, dan untuk rentang

port untuk jenis EC2 instans Amazon lainnya, lihat Port Ephemeral di Panduan Pengguna Amazon VPC.

- Keluar: Semua sumber lalu lintas menggunakan protokol dan port.

Anda dapat mengatur perilaku ini pada tingkat grup keamanan. Untuk tingkat keamanan tambahan, Anda juga dapat menggunakan ACL jaringan. Untuk informasi selengkapnya, lihat Perbandingan grup keamanan dan jaringan ACLs di Panduan Pengguna Amazon VPC.

Misalnya, untuk menambahkan aturan masuk dan keluar untuk grup keamanan, Anda dapat mengatur aturan-aturan tersebut sebagai berikut.

Aturan-aturan ke dalam

| Jenis | Protokol | Rentang pelabuhan | Sumber |
|---|---|---|---|
| SSH (22) | TCP (6) | 22 | 0.0.0.0 (Tapi lihat catatan berikut dan Rentang alamat IP SSH masuk untuk AWS Cloud9.) |

> **Note**
>
> Untuk EC2 lingkungan yang dibuat pada atau setelah 31 Juli 2018, AWS Cloud9 menambahkan aturan masuk untuk membatasi alamat IP masuk menggunakan SSH melalui port 22. Ini membatasi secara khusus hanya alamat yang AWS Cloud9 menggunakan. Untuk informasi selengkapnya, lihat Rentang alamat IP SSH masuk untuk AWS Cloud9.

Aturan-aturan ke luar

| Jenis | Protokol | Rentang pelabuhan | Sumber |
|---|---|---|---|
| Semua Lalu lintas | SEMUA | SEMUA | 0.0.0.0/0 |

Jika Anda juga memilih untuk menambahkan aturan masuk dan keluar ke ACL jaringan, Anda dapat mengatur aturan tersebut sebagai berikut.

Aturan-aturan ke dalam

| Aturan # | Jenis | Protokol | Rentang pelabuhan | Sumber | Izinkan/Tolak |
|---|---|---|---|---|---|
| 100 | SSH (22) | TCP (6) | 22 | 0.0.0.0 (Tapi lihat Rentang alamat IP SSH masuk untuk AWS Cloud9.) | IZINKAN |
| 200 | Aturan TCP kustom | TCP (6) | 32768-61000 (Untuk instans Amazon Linux dan Ubuntu Server. Untuk tipe instans lainnya, lihat Port Ephemeral.) | 0.0.0.0/0 | IZINKAN |
| * | Semua Lalu lintas | SEMUA | SEMUA | 0.0.0.0/0 | MENOLAK |

Aturan-aturan ke luar

| Aturan # | Jenis | Protokol | Rentang pelabuhan | Sumber | Izinkan/Tolak |
|---|---|---|---|---|---|
| 100 | Semua Lalu lintas | SEMUA | SEMUA | 0.0.0.0/0 | IZINKAN |

| Aturan # | Jenis | Protokol | Rentang pelabuhan | Sumber | Izinkan/Tolak |
|---|---|---|---|---|---|
| * | Semua Lalu lintas | SEMUA | SEMUA | 0.0.0.0/0 | MENOLAK |

Untuk informasi selengkapnya tentang grup keamanan dan jaringan ACLs, lihat berikut ini di Panduan Pengguna Amazon VPC.

- Keamanan
- Grup keamanan untuk VPC Anda
- Jaringan ACLs

## Buat grup keamanan di VPC

Untuk menggunakan EC2 konsol Amazon VPC atau Amazon, lakukan salah satu tindakan berikut:

- Di konsol Amazon VPC, memilih Grup Keamanan di panel navigasi. Memilih Buat Grup Keamanan, lalu ikuti petunjuk di layar.
- Di EC2 konsol Amazon, perluas Jaringan & Keamanan di panel navigasi, lalu pilih Grup Keamanan. Memilih Buat Grup Keamanan, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`create-security-group`**perintah Amazon, misalnya, sebagai berikut.

```
aws ec2 create-security-group --region us-east-2 --vpc-id vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi VPC, dan ganti dengan `vpc-1234ab56` ID VPC. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Mengonfirmasi apakah VPC memiliki setidaknya satu ACL jaringan

Untuk menggunakan konsol Amazon VPC, pilih Your VPCs di panel navigasi. Pilih kotak di sebelah VPC yang AWS Cloud9 ingin Anda gunakan. Pada tab Ringkasan, jika ada nilai untuk ACL Jaringan, VPC memiliki setidaknya satu ACL jaringan.

Untuk menggunakan AWS CLI atauaws-shell, jalankan EC2 **describe-network-acls**perintah Amazon.

```
aws ec2 describe-network-acls --output table --query
 'NetworkAcls[*].Associations[*].NetworkAclId' --region us-east-2 --filters Name=vpc-
id,Values=vpc-1234ab56
```

Pada perintah sebelumnya, ganti us-east-2 dengan Wilayah AWS yang berisi VPC, dan ganti dengan vpc-1234ab56 ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya denganaws-shell, hilangkan. aws

Jika output berisi setidaknya satu entri dalam daftar, VPC memiliki setidaknya satu ACL jaringan.

## Lihat daftar jaringan ACLs untuk VPC

Untuk menggunakan konsol VPC Amazon, pilih Jaringan ACLs di panel navigasi. Di ACLs kotak Jaringan Pencarian, masukkan ID atau nama VPC, lalu tekan. Enter Jaringan ACLs untuk VPC itu muncul dalam daftar hasil pencarian.

Untuk menggunakan AWS CLI atauaws-shell, jalankan EC2 **describe-network-acls**perintah Amazon.

```
aws ec2 describe-network-acls --output table --query
 'NetworkAcls[*].Associations[*].NetworkAclId' --region us-east-2 --filters Name=vpc-
id,Values=vpc-1234ab56
```

Pada perintah sebelumnya, ganti us-east-2 dengan Wilayah AWS yang berisi VPC, dan ganti dengan vpc-1234ab56 ID VPC. Untuk menjalankan perintah sebelumnya di Windows, ganti tanda kutip tunggal (') dengan tanda kutip ganda (“”). Untuk menjalankan perintah sebelumnya denganaws-shell, hilangkan. aws

Outputnya berisi daftar jaringan ACLs untuk VPC itu.

## Melihat atau mengubah pengaturan untuk ACL jaringan

Untuk menggunakan konsol VPC Amazon, pilih Jaringan ACLs di panel navigasi. Memilih kotak di samping ACL jaringan. Untuk melihat pengaturan, lihat setiap tab. Untuk mengubah pengaturan pada tab, memilih Edit, jika berlaku, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau `aws-shell` untuk melihat pengaturan, jalankan EC2 **`describe-network-acls`**perintah Amazon.

```
aws ec2 describe-network-acls --output table --region us-east-2 --network-acl-ids
 acl-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi ACL jaringan, dan ganti `acl-1234ab56` dengan ID ACL jaringan. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

### Buat ACL jaringan

Untuk menggunakan konsol VPC Amazon, pilih Jaringan ACLs di panel navigasi. Memilih Buat ACL Jaringan, lalu ikuti petunjuk di layar.

Untuk menggunakan AWS CLI atau`aws-shell`, jalankan EC2 **`create-network-acl`**perintah Amazon.

```
aws ec2 create-network-acl --region us-east-2 --vpc-id vpc-1234ab56
```

Pada perintah sebelumnya, ganti `us-east-2` dengan Wilayah AWS yang berisi VPC yang ingin Anda pasang ACL jaringan baru. Juga, ganti `vpc-1234ab56` dengan ID VPC. Untuk menjalankan perintah sebelumnya dengan`aws-shell`, hilangkan. `aws`

## Buat VPC plus sumber daya VPC lainnya

Gunakan prosedur berikut untuk membuat VPC dan sumber daya VPC tambahan yang Anda perlukan untuk menjalankan aplikasi Anda. Sumber daya VPC termasuk subnet, tabel rute, gateway internet, dan gateway NAT.

Untuk membuat VPC, subnet, dan sumber daya VPC lainnya menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di dasbor VPC, pilih Buat VPC.
3. Agar Sumber Daya dapat dibuat, pilih VPC dan lainnya.
4. Untuk membuat tag nama untuk sumber daya VPC, pertahankan pembuatan otomatis tag Nama tetap dipilih. Untuk memberikan tag nama Anda sendiri untuk sumber daya VPC, kosongkan.

5. Untuk blok IPv4 CIDR, Anda harus memasukkan rentang IPv4 alamat untuk VPC. IPv4 Rentang yang direkomendasikan untuk AWS Cloud9 adalah`10.0.0.0/16`.
6. (Opsional) Untuk mendukung IPv6 lalu lintas, pilih blok IPv6 CIDR, blok CIDR yang disediakan Amazon IPv6 .
7. Pilih opsi Sewa. Opsi ini menentukan apakah EC2 instance yang Anda luncurkan ke VPC akan berjalan pada perangkat keras yang dibagikan dengan perangkat keras lain Akun AWS atau pada perangkat keras yang didedikasikan untuk penggunaan Anda saja. Jika Anda memilih penyewaan VPC yang `Default` akan menjadi EC2 , instance yang diluncurkan ke VPC ini akan menggunakan atribut penyewaan yang ditentukan saat Anda meluncurkan instance. Untuk informasi selengkapnya, lihat Meluncurkan instance menggunakan parameter yang ditentukan di Panduan EC2 Pengguna Amazon.

   Jika Anda memilih penyewaan VPC, instans akan selalu berjalan sebagai Instans Khusus pada perangkat keras yang didedikasikan untuk Anda gunakan. `Dedicated` Jika Anda menggunakan AWS Outposts, Anda Outpost membutuhkan konektivitas pribadi, dan Anda harus menggunakan `Default` penyewaan.
8. Untuk Jumlah Availability Zones (AZs), kami sarankan Anda menyediakan subnet setidaknya dalam dua Availability Zones untuk lingkungan produksi. Untuk memilih subnet Anda, perluas Kustomisasi AZs. AZs Jika tidak, Anda dapat membiarkan AWS memilih AZs untuk Anda.
9. Untuk mengkonfigurasi subnet Anda, pilih nilai untuk Jumlah subnet publik dan Jumlah subnet pribadi. Untuk memilih rentang alamat IP untuk subnet Anda, perluas Sesuaikan subnet blok CIDR. Jika tidak, biarkan AWS memilihnya untuk Anda.
10. (Opsional) Jika sumber daya dalam subnet pribadi memerlukan akses ke internet publik melalui IPv4: Untuk gateway NAT, pilih jumlah AZs di mana untuk membuat gateway NAT. Dalam produksi, kami menyarankan Anda menerapkan gateway NAT di setiap AZ dengan sumber daya yang memerlukan akses ke internet publik.
11. (Opsional) Jika sumber daya di subnet pribadi memerlukan akses ke internet publik melalui IPv6: Untuk gateway internet khusus Egress, pilih Ya.
12. (Opsional) Untuk mengakses Amazon S3 langsung dari VPC Anda, pilih titik akhir VPC, S3 Gateway. Ini menciptakan titik akhir VPC gateway untuk Amazon S3. Untuk informasi selengkapnya, lihat titik akhir VPC Gateway di Panduan.AWS PrivateLink
13. (Opsional) Untuk opsi DNS, kedua opsi untuk resolusi nama domain diaktifkan secara default. Jika default tidak memenuhi kebutuhan Anda, Anda dapat menonaktifkan opsi ini.
14. (Opsional) Untuk menambahkan tag ke VPC Anda, perluas Tag tambahan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.

15. Di panel Pratinjau, Anda dapat memvisualisasikan hubungan antara sumber daya VPC yang dikonfigurasi. Garis padat mewakili hubungan antar sumber daya. Garis putus-putus mewakili lalu lintas jaringan ke gateway NAT, gateway internet, dan titik akhir gateway. Setelah Anda membuat VPC, Anda dapat memvisualisasikan sumber daya di VPC Anda dalam format ini kapan saja menggunakan tab Peta sumber daya.
16. Setelah Anda selesai mengonfigurasi VPC Anda, pilih Buat VPC.

## Buat VPC saja

Gunakan prosedur berikut untuk membuat VPC tanpa sumber daya VPC tambahan dengan menggunakan konsol VPC Amazon.

Untuk membuat VPC tanpa sumber daya VPC tambahan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di dasbor VPC, pilih Buat VPC.
3. Agar Sumber Daya dapat dibuat, pilih VPC saja.
4. (Opsional) Untuk tag Nama, masukkan nama untuk VPC Anda. Melakukan hal itu akan menciptakan tag dengan kunci `Name` dan nilai yang Anda tentukan.
5. Untuk blok IPv4 CIDR, lakukan salah satu hal berikut:
   - Pilih input manual IPv4 CIDR dan masukkan rentang IPv4 alamat untuk VPC Anda. IPv4 Rentang yang direkomendasikan untuk AWS Cloud9 adalah`10.0.0.0/16`.
   - Pilih blok IPv4 CIDR yang dialokasikan IPAM, pilih kumpulan alamat Amazon VPC IP Address Manager (IPAM) dan netmask. IPv4 Ukuran blok CIDR dibatasi oleh aturan alokasi pada kolam IPAM. IPAM adalah fitur VPC yang membantu Anda merencanakan, melacak, dan memantau alamat IP untuk AWS beban kerja Anda. Untuk informasi lebih lanjut, lihat Apa itu IPAM? di Panduan Administrator Amazon Virtual Private Cloud.

     Jika Anda menggunakan IPAM untuk mengelola alamat IP Anda, kami sarankan Anda memilih opsi ini. Jika tidak, blok CIDR yang Anda tentukan untuk VPC Anda mungkin tumpang tindih dengan alokasi CIDR IPAM.
6. (Opsional) Untuk membuat VPC tumpukan ganda, tentukan rentang IPv6 alamat untuk VPC Anda. Untuk blok IPv6 CIDR, lakukan salah satu hal berikut:
   - Pilih blok IPv6 CIDR yang dialokasikan IPAM dan pilih kumpulan alamat IPAM Anda. IPv6 Ukuran blok CIDR dibatasi oleh aturan alokasi pada kolam IPAM.

- Untuk meminta blok IPv6 CIDR dari kumpulan IPv6 alamat Amazon, pilih blok CIDR yang disediakan Amazon IPv6 . Untuk Grup Perbatasan Jaringan, pilih grup tempat AWS mengiklankan alamat IP. Amazon menyediakan ukuran blok IPv6 CIDR tetap /56.
- Pilih IPv6 CIDR yang dimiliki oleh saya untuk menggunakan blok IPv6 CIDR yang Anda bawa AWS menggunakan bawa alamat IP Anda sendiri (BYOIP). Untuk Pool, pilih kumpulan IPv6 alamat untuk mengalokasikan blok IPv6 CIDR.

7. (Opsional) Pilih opsi Sewa. Opsi ini menentukan apakah EC2 instance yang Anda luncurkan ke VPC akan berjalan pada perangkat keras yang dibagikan dengan perangkat keras lain Akun AWS atau pada perangkat keras yang didedikasikan untuk penggunaan Anda saja. Jika Anda memilih penyewaan VPC yang `Default` akan menjadi EC2 , instance yang diluncurkan ke VPC ini akan menggunakan atribut penyewaan yang ditentukan saat Anda meluncurkan instance. Untuk informasi selengkapnya, lihat Meluncurkan instance menggunakan parameter yang ditentukan di Panduan EC2 Pengguna Amazon.

   Jika Anda memilih penyewaan VPC, instans akan selalu berjalan sebagai Instans Khusus pada perangkat keras yang didedikasikan untuk Anda gunakan. `Dedicated` Jika Anda menggunakan AWS Outposts, Anda Outpost membutuhkan konektivitas pribadi, dan Anda harus menggunakan `Default` penyewaan.
8. (Opsional) Untuk menambahkan tag ke VPC Anda, pilih Tambahkan tag baru dan masukkan kunci tag dan nilai tag.
9. Pilih Buat VPC.
10. Setelah Anda membuat VPC, Anda dapat menambahkan subnet.

## Buat subnet untuk AWS Cloud9

Anda dapat menggunakan konsol VPC Amazon untuk membuat subnet untuk VPC yang kompatibel dengannya. AWS Cloud9 Apakah Anda dapat membuat subnet pribadi atau publik untuk EC2 instans Anda tergantung pada bagaimana lingkungan Anda terhubung dengannya:

- Akses langsung melalui SSH: hanya subnet publik
- Akses melalui Systems Manager: subnet publik atau subnet privat

Opsi untuk meluncurkan lingkungan Anda EC2 ke subnet pribadi hanya tersedia jika Anda membuat EC2 lingkungan “no-ingress” menggunakan konsol, baris perintah, atau. AWS CloudFormation

Anda mengikuti Langkah yang sama untuk membuat subnet yang dapat dibuat publik atau privat. Jika subnet dikaitkan dengan tabel rute yang memiliki rute ke gateway internet, itu menjadi subnet publik. Jika subnet dikaitkan dengan tabel rute yang tidak memiliki rute ke gateway internet, itu menjadi subnet publik. Untuk informasi selengkapnya, silakan lihat Mengonfigurasi subnet sebagai publik atau subnet privat

Jika Anda mengikuti prosedur sebelumnya untuk membuat VPC AWS Cloud9, Anda juga tidak perlu mengikuti prosedur ini. Hal ini karena wizard Buat VPC baru menciptakan subnet untuk Anda secara otomatis.

⚠ Important

- Akun AWS Harus sudah memiliki VPC yang kompatibel dalam hal yang sama Wilayah AWS untuk lingkungan. Untuk informasi lebih lanjut, lihat Persyaratan VPC di Persyaratan Amazon VPC untuk AWS Cloud9.
- Untuk prosedur ini, kami menyarankan Anda masuk ke AWS Management Console dan membuka konsol VPC Amazon menggunakan kredensi untuk administrator IAM di Anda. Akun AWS Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.
- Beberapa organisasi mungkin tidak mengizinkan Anda membuat subnet sendiri. Jika Anda tidak dapat membuat subnet, periksa dengan Akun AWS administrator atau administrator jaringan Anda.

Untuk membuat subnet

1. Jika konsol VPC Amazon belum terbuka, masuk ke AWS Management Console dan buka konsol VPC Amazon di /vpc. https://console.aws.amazon.com
2. Di bilah navigasi, jika Wilayah AWS tidak sama dengan Wilayah untuk lingkungan, pilih Wilayah yang benar.
3. Memilih Subnet di panel navigasi, jika halaman Subnet belum ditampilkan.
4. Pilih Buat Subnet
5. Di kotak dialog Buat Subnet, untuk Tanda nama, memasukkan nama untuk subnet.
6. Untuk VPC, memilih VPC untuk mengaitkan subnet.
7. Untuk Availability Zone, pilih Availability Zone dalam subnet yang akan digunakan, atau pilih No Preference untuk membiarkan AWS memilih Availability Zone untuk Anda. Wilayah AWS

8. Untuk blok IPv4 CIDR, masukkan rentang alamat IP untuk subnet yang akan digunakan, dalam format CIDR. Kisaran alamat IP ini harus menjadi bagian dari alamat IP di VPC.

   Untuk informasi tentang blok CIDR, lihat Ukuran VPC dan subnet dalam Panduan Pengguna Amazon VPC. Lihat juga 3.1. Konsep Dasar dan Notasi Awalan di RFC 4632 atau blok IPv4 CIDR di Wikipedia.

Setelah Anda membuat subnet, mengonfigurasi itu sebagai subnet publik atau subnet privat.

## Mengonfigurasi subnet sebagai publik atau subnet privat

Setelah Anda membuat subnet, Anda dapat membuatnya publik atau privat dengan menentukan bagaimana ia berkomunikasi dengan internet.

Subnet publik memiliki alamat IP publik dan gateway internet (IGW) melekat padanya yang memungkinkan komunikasi antara instance untuk subnet dan internet dan lainnya. Layanan AWS

Sebuah contoh dalam subnet pribadi memiliki alamat IP pribadi dan gateway terjemahan alamat jaringan (NAT) digunakan untuk mengirim lalu lintas bolak-balik antara instance untuk subnet dan internet dan lainnya. Layanan AWS NAT gateway harus di-host di subnet publik.

Public subnets

> **Note**
>
> Bahkan jika instans untuk lingkungan Anda diluncurkan di subnet privat, VPC Anda harus memiliki fitur setidaknya satu subnet publik. Hal ini karena NAT gateway yang meneruskan lalu lintas ke dan dari instans harus di-host di subnet publik.

Mengonfigurasi subnet sebagai publik melibatkan melampirkan gateway internet (IGW) untuk itu, mengonfigurasi tabel rute untuk menentukan rute ke IGW itu, dan menentukan pengaturan dalam grup keamanan untuk mengontrol lalu lintas masuk dan keluar.

Bimbingan untuk melaksanakan tugas-tugas ini tersedia di Buat VPC plus sumber daya VPC lainnya.

> **Important**
>
> Jika lingkungan pengembangan Anda menggunakan SSM untuk mengakses EC2 instance, pastikan instans tersebut diberi alamat IP publik oleh subnet publik yang diluncurkan. Untuk melakukannya, Anda harus mengaktifkan penetapan otomatis opsi alamat IP publik untuk subnet publik, dan mengaturnya ke. `Yes` Anda dapat mengaktifkan ini di subnet publik sebelum membuat AWS Cloud9 lingkungan dalam halaman pengaturan subnet. Untuk langkah-langkah yang terlibat dalam memodifikasi pengaturan IP penetapan otomatis di subnet publik, lihat Memodifikasi atribut IPv4 pengalamatan publik untuk subnet Anda di Panduan Pengguna Amazon VPC. Untuk informasi selengkapnya tentang mengonfigurasi subnet publik dan pribadi, lihat. Mengonfigurasi subnet sebagai publik atau subnet privat

Private subnets

Jika Anda membuat instans no-ingress yang diakses melalui Systems Manager, Anda dapat meluncurkannya ke subnet privat. Subnet pribadi tidak memiliki alamat IP publik. Jadi, Anda memerlukan gateway NAT untuk memetakan alamat IP pribadi ke alamat publik untuk permintaan, dan Anda juga perlu memetakan alamat IP publik kembali ke alamat pribadi untuk respons.

> **Warning**
>
> Anda dikenakan biaya untuk membuat dan menggunakan gateway NAT di akun Anda. Tarif penggunaan dan pemrosesan data gateway NAT per jam berlaku. EC2 Biaya Amazon untuk transfer data juga berlaku. Untuk informasi lebih lanjut, lihat Harga Amazon VPC.

Sebelum membuat dan mengonfigurasi gateway NAT, Anda harus melakukan hal berikut:

- Buat subnet VPC publik untuk host gateway NAT.
- Penyediaan Alamat IP elastis yang dapat ditugaskan ke gateway NAT.
- Untuk subnet pribadi, kosongkan kotak centang Enable auto-assign public IPv4 address sehingga instance yang diluncurkan ke dalamnya diberi alamat IP pribadi. Untuk informasi selengkapnya, lihat Penentuan Alamat IP di VPC Anda di Panduan Pengguna Amazon VPC.

Untuk langkah-langkah dalam tugas ini, lihat Bekerja dengan gateway NAT di Panduan Pengguna Amazon VPC.

> **Important**
>
> Saat ini, jika EC2 instance lingkungan Anda diluncurkan ke subnet pribadi, Anda tidak dapat menggunakan kredensil sementara AWS terkelola untuk memungkinkan EC2 lingkungan mengakses Layanan AWS atas nama AWS entitas seperti pengguna IAM.

# Persyaratan host lingkungan SSH

Untuk menginstruksikan AWS Cloud9 agar menghubungkan lingkungan ke instans komputasi cloud yang ada atau server Anda sendiri, Anda membuat lingkungan pengembangan AWS Cloud9 SSH. Namun, sebelum Anda membuat lingkungan SSH, pertimbangkan manfaat menciptakan EC2 lingkungan sebagai gantinya.

Saat Anda membuat EC2 lingkungan, AWS Cloud9 membuat lingkungan baru, meminta Amazon EC2 untuk meluncurkan instance baru, dan kemudian menghubungkan instance yang baru diluncurkan ke lingkungan baru. Menciptakan EC2 lingkungan memiliki manfaat sebagai berikut:

- Peluncuran instans otomatis. Saat Anda membuat EC2 lingkungan, AWS Cloud9 meminta Amazon EC2 untuk membuat instance baru secara bersamaan. Di lingkungan SSH, Anda harus menyediakan instans komputasi awan yang ada (misalnya, EC2 instans Amazon) atau server Anda sendiri.
- Shutdown instance otomatis. Secara default, AWS Cloud9 secara otomatis mematikan EC2 lingkungan 30 menit setelah semua instance browser web yang terhubung ke IDE untuk EC2 lingkungan ditutup. Anda dapat mengubah perilaku ini kapan saja. Ini membantu mengurangi kemungkinan memiliki biaya tambahan yang diterapkan pada Anda Akun AWS untuk menggunakan Amazon EC2.
- Pembersihan instans otomatis. Saat Anda menghapus EC2 lingkungan, EC2 instans Amazon yang terhubung akan dihapus secara otomatis. Ini juga membantu mengurangi kemungkinan biaya tambahan yang diterapkan pada Anda Akun AWS untuk menggunakan Amazon EC2. Dalam lingkungan SSH yang terhubung ke instans komputasi cloud, Anda harus ingat untuk menghapus instans sendiri.
- AWS kredensi sementara yang dikelola. Untuk EC2 lingkungan, Anda dapat dengan mudah mengaktifkan atau menonaktifkan semua AWS tindakan untuk semua AWS sumber daya di

pemanggil Akun AWS (dengan beberapa batasan). Anda tidak perlu mengonfigurasi profil instans untuk EC2 instans Amazon lingkungan Anda atau menyimpan kredenal AWS akses permanen AWS entitas (misalnya, pengguna IAM).

Untuk informasi selengkapnya, lihat AWS kredensi sementara yang dikelola.

- AWS Toolkit dan panel Git. Alat ini untuk berinteraksi dengan Layanan AWS dan menggunakan kontrol sumber visual hanya tersedia di AWS Cloud9 lingkungan yang dibuat dengan EC2 instans Amazon.

Jika Anda ingin membuat EC2 lingkungan sebagai gantinya, lihatMenciptakan EC2 Lingkungan. Jika tidak, lanjutkan membaca untuk informasi tentang menciptakan lingkungan SSH.

## Waktu dan cara membuat Lingkungan SSH

Anda harus membuat lingkungan SSH alih-alih EC2 lingkungan setiap kali Anda memiliki salah satu persyaratan berikut:

| Persyaratan | Petunjuk arah |
| --- | --- |
| Anda tidak ingin dikenakan biaya tambahan Akun AWS untuk menggunakan instans AWS Cloud komputasi. Jadi, Anda memutuskan untuk terhubung AWS Cloud9 ke instance komputasi cloud yang ada di luar AWS atau server Anda sendiri. | 1. Pastikan instance atau server Anda memenuhi persyaratan yang dijelaskan nanti dalam topik ini.<br>2. Buat lingkungan SSH agar AWS Cloud9 dapat menghubungkan instans atau server Anda. |
| Anda ingin menggunakan instance komputasi AWS cloud yang ada (misalnya, EC2 instans Amazon) di dalam instans Anda Akun AWS alih-alih AWS Cloud9 harus meluncurkan instance baru pada saat yang sama lingkungan dibuat. | 1. Pastikan instance memenuhi persyaratan yang dijelaskan nanti dalam topik ini.<br>2. Menciptakan lingkungan SSH pada AWS Cloud9 untuk menghubungkan instans. |
| Anda ingin menggunakan jenis EC2 instans Amazon yang AWS Cloud9 saat ini tidak mendukung EC2 lingkungan (misalnya, R4). | 1. Luncurkan EC2 instans Amazon berdasarkan jenis instans yang Anda inginkan. Atau, identifikasi instance yang ada di |

| Persyaratan | Petunjuk arah |
| --- | --- |
| | Anda Akun AWS yang menjalankan jenis instance yang diinginkan.<br>2. Pastikan instance memenuhi persyaratan yang dijelaskan nanti dalam topik ini.<br>3. Menciptakan lingkungan SSH pada AWS Cloud9 untuk menghubungkan instans. |
| Anda ingin menggunakan EC2 instance Amazon yang didasarkan pada Amazon Machine Image (AMI) selain Amazon Linux atau Ubuntu Peladen. | 1. Luncurkan EC2 instans Amazon berdasarkan AMI yang Anda inginkan. Atau, identifikasi instance yang ada di Anda Akun AWS yang didasarkan pada AMI yang Anda inginkan.<br>2. Pastikan instance memenuhi persyaratan yang dijelaskan nanti dalam topik ini.<br>3. Menciptakan lingkungan SSH pada AWS Cloud9 untuk menghubungkan instans. |
| Anda ingin menghubungkan beberapa lingkungan untuk instans komputasi cloud tunggal yang ada atau server Anda sendiri. | 1. Pastikan instance atau server memenuhi persyaratan yang dijelaskan nanti dalam topik ini.<br>2. Buat lingkungan SSH untuk setiap lingkungan yang AWS Cloud9 ingin Anda hubungkan dengan instance atau server. |

> **Note**
>
> Meluncurkan EC2 instans Amazon dapat mengakibatkan kemungkinan biaya Akun AWS untuk Amazon Anda EC2. Untuk informasi lebih lanjut, lihat Amazon EC2 Harga.

## Persyaratan hosting SSH

Instans komputasi awan yang ada atau server Anda sendiri harus memenuhi persyaratan berikut AWS Cloud9 untuk menghubungkannya ke lingkungan SSH.

- Itu harus menjalankan Linux. (AWS Cloud9 tidak mendukung Windows.)
- Itu tidak harus menggunakan Armarsitektur berbasis. (Support untuk sistem yang dibangun di sekitar Arm prosesor sedang ditinjau.)
- Instans ini harus dicapai melalui internet publik dengan menggunakan SSH. Jika Instans ini hanya dapat dijangkau melalui virtual private cloud (VPC) atau jaringan pribadi virtual (VPN), VPC atau VPN tersebut harus memiliki akses ke internet publik.
- Jika host adalah instance komputasi AWS Cloud yang sudah ada yang merupakan bagian dari Amazon Virtual Private Cloud (Amazon VPC), ada persyaratan tambahan. Untuk informasi selengkapnya, lihat Pengaturan VPC Amazon.
- Itu harus memiliki Python3 diinstal dan ditetapkan sebagai default Python versi dan pip3 saat menginstal AWS Cloud9. Untuk memeriksa versi, dari terminal instance yang ada atau server Anda, jalankan perintah **`python --version`**. Untuk menginstal Python pada instance atau server, lihat salah satu sumber daya berikut:
  - Langkah 1: Instal Alat yang Diperlukan di Python Sampel.
  - Unduh Pythondari Python situs web.

  > **Note**
  >
  > Untuk menyambung ke instans AWS Cloud komputasi yang ada untuk memverifikasi dan memenuhi persyaratan, lihat satu atau beberapa sumber daya berikut:
  >
  > - Untuk Amazon EC2, lihat Connect to Your Linux Instance di Panduan EC2 Pengguna Amazon.
  > - Untuk Amazon Lightsail, lihat Menghubungkan ke instans Lightsail berbasis Linux/Unix Anda di Dokumentasi Amazon Lightsail.
  > - Untuk AWS Elastic Beanstalk, lihat Daftar dan Menghubungkan ke Instans Server di Panduan AWS Elastic Beanstalk Pengembang.
  > - Untuk AWS OpsWorks, lihat Menggunakan SSH untuk Masuk ke Instans Linux di Panduan AWS OpsWorks Pengguna.
  > - Untuk yang lain Layanan AWS, lihat dokumentasi layanan.
  >
  > Untuk terhubung ke server Anda sendiri untuk memverifikasi dan memenuhi persyaratan, cari di internet menggunakan frasa seperti “sambungkan ke server dengan menggunakan perintah SSH” (dari macOS atau Linux) atau “sambungkan ke server dengan menggunakan PuTTY“(dari Windows).

- Jalankan perintah berikut untuk menginstal semua paket yang diperlukan.

Untuk Amazon Linux:

```
sudo yum install -y make glibc-devel gcc gcc-c++
```

Untuk Ubuntu Server:

```
sudo apt install build-essential
```

- Node.js harus diinstal. Kami merekomendasikan menginstal yang terbaru Node.js versi yang didukung oleh sistem operasi host.

> **Warning**
>
> AWS Cloud9 Masalah instalasi mungkin terjadi saat membuat lingkungan SSH jika Anda menggunakan versi Node.js yang tidak didukung oleh AWS Cloud9.

Untuk memeriksa versi Anda, dari terminal instans yang ada atau server Anda, jalankan perintah **`node --version`** . Untuk menginstal Node.js pada instans atau server, lihat salah satu sumber daya berikut:

- Langkah 1: Pasang alat yang diperlukan di Contoh Node.js.
- Menginstal Node.js melalui manajer paket pada situs web Node.js.
- Pengelola Versi Node aktif GitHub.

- Jalur ke direktori pada instans atau server yang ada yang menurut Anda perlu AWS Cloud9 mulai setelah login harus memiliki izin akses yang disetel ke `rwxr-xr-x`. Ini berarti bahwa read-write-run izin untuk pemilik yang sesuai dengan nama login yang Anda tentukan di wizard buat lingkungan untuk Pengguna di halaman Konfigurasi pengaturan, izin baca-jalankan untuk grup yang dimiliki pemilik ini, dan izin baca-jalankan untuk orang lain.

  Misalnya, jika jalur direktori adalah ~ (di mana ~ mewakili direktori home untuk nama login yang Anda tentukan untuk Pengguna di halaman Konfigurasi pengaturan), Anda dapat mengatur izin ini pada direktori dengan menjalankan **`chmod`**perintah pada instance atau server menggunakan perintah berikut dan instruksi yang mengikuti.

  ```
  sudo chmod u=rwx,g=rx,o=rx ~
  ```

- Unduh dan jalankan Penginstal AWS Cloud9 pada instans atau server yang ada.

- Secara opsional, Anda dapat membatasi lalu lintas masuk melalui SSH hanya untuk alamat IP yang menggunakan. AWS Cloud9 Untuk melakukannya, atur lalu lintas SSH masuk ke rentang IP seperti yang dijelaskan dalam Rentang alamat IP SSH masuk untuk AWS Cloud9.

Setelah Anda yakin instans atau server Anda memenuhi persyaratan sebelumnya, buat lingkungan SSH AWS Cloud9 untuk menghubungkannya.

# Menggunakan AWS Cloud9 Installer untuk lingkungan AWS Cloud9 SSH

Sebelum Anda membuat lingkungan pengembangan AWS Cloud9 SSH, instance komputasi awan (misalnya instans Amazon EC2) atau server Anda sendiri yang ingin Anda sambungkan ke lingkungan harus memenuhi Persyaratan Host SSH. Salah satu persyaratan ini adalah Anda harus mengunduh dan menjalankan AWS Cloud9 Installer pada instance atau server. AWS Cloud9 Installer adalah skrip shell Linux yang memeriksa apakah instance atau server berjalan pada platform sistem operasi dan arsitektur yang AWS Cloud9 mendukung. Jika pemeriksaan ini berhasil, skrip kemudian mencoba menginstal komponen dan dependensinya yang AWS Cloud9 harus ada di instance atau server.

Topik ini menjelaskan cara mengunduh dan menjalankan skrip penginstal ini pada instans atau server target.

- Unduh dan Jalankan AWS Cloud9 Installer
- Memecahkan Masalah Installer AWS Cloud9

## Unduh dan Jalankan AWS Cloud9 Installer

1. Pastikan instans komputasi cloud atau server Anda sendiri yang ingin Anda hubungkan ke lingkungan memenuhi Persyaratan Host SSH. Ini termasuk memiliki versi Python dan Node.js tertentu yang sudah diinstal, mengatur izin khusus pada direktori yang Anda inginkan untuk memulai AWS Cloud9 setelah login, dan menyiapkan Amazon Virtual Private Cloud terkait.
2. Ketika Anda terhubung ke instans atau server, jalankan salah satu perintah berikut pada instans atau server tersebut. Anda harus menginstal `gcc` sebelum menjalankan salah satu perintah.

```
curl -L https://d3kgj69l4ph6w4.cloudfront.net/static/c9-install-2.0.0.sh | bash
wget -O - https://d3kgj69l4ph6w4.cloudfront.net/static/c9-install-2.0.0.sh | bash
```

3. Jika pesan Selesai ditampilkan tanpa kesalahan, Anda dapat membuat lingkungan SSH.

   Jika pesan kesalahan ditampilkan, lihat bagian berikutnya untuk informasi pemecahan masalah.

## Memecahkan Masalah Installer AWS Cloud9

Bagian ini menjelaskan masalah umum, kemungkinan penyebab, dan solusi yang disarankan untuk memecahkan masalah kesalahan AWS Cloud9 Penginstal.

Jika masalah Anda tidak terdaftar, atau jika Anda memerlukan bantuan tambahan, lihat Forum Diskusi AWS Cloud9. (Saat Anda memasuki forum ini, AWS mungkin mengharuskan Anda untuk masuk.) Anda juga dapat kontak kami secara langsung.

- -bash: wget: perintah tidak ditemukan
- Kesalahan: silakan instal make untuk melanjutkan
- Kesalahan: silakan instal gcc untuk melanjutkan
- konfigurasi: kesalahan: kutukan tidak ditemukan

### -bash: wget: perintah tidak ditemukan

Masalah: Ketika Anda menjalankan skrip penginstal, pesan berikut akan ditampilkan: `-bash: wget: command not found`.

Kemungkinan penyebab: Utilitas **`wget`** tidak diinstal pada instans atau server.

Solusi yang Direkomendasikan: Jalankan skrip penginstal pada instans atau server dengan utilitas **`curl`** sebagai gantinya.

### Kesalahan: silakan instal make untuk melanjutkan

Masalah: Ketika Anda menjalankan skrip penginstal, pesan berikut akan ditampilkan: `Error: please install make to proceed`.

Kemungkinan penyebab: Utilitas **`make`** tidak diinstal pada instans atau server.

Solusi yang Direkomendasikan: Instal utilitas **`make`** , lalu coba jalankan skrip penginstal pada instans atau server lagi.

Untuk menginstal utilitas **`make`** , jalankan salah satu perintah berikut di instans atau server Anda.

- Untuk Amazon Linux, Amazon Linux 2, dan Red Hat Enterprise Linux (RHEL) yang berjalan di Amazon EC2: **`sudo yum -y groupinstall "Development Tools"`**
- Untuk Server Ubuntu yang berjalan di Amazon EC2: **`sudo apt install -y build-essential`**
- Untuk SUSE: **`sudo zypper install -y make`**

### Kesalahan: silakan instal gcc untuk melanjutkan

Masalah: Ketika Anda menjalankan skrip penginstal, pesan berikut akan ditampilkan: `Error: please install gcc to proceed`.

Kemungkinan penyebab: Utilitas **`gcc`** tidak diinstal pada instans atau server.

Solusi yang Direkomendasikan: Instal utilitas **`gcc`** , lalu coba jalankan skrip penginstal pada instans atau server lagi.

Untuk menginstal utilitas **`gcc`** , jalankan salah satu perintah berikut di instans atau server Anda.

- Untuk Amazon Linux, Amazon Linux 2, dan Red Hat Enterprise Linux (RHEL) yang berjalan di Amazon EC2: **`sudo yum -y groupinstall "Development Tools"`**
- Untuk Server Ubuntu yang berjalan di Amazon EC2: **`sudo apt install -y build-essential`**
- Untuk SUSE: **`sudo zypper install -y gcc`**
- Untuk sistem operasi lainnya, lihat Menginstal GCC.

### configure: error: curses tidak ditemukan

Masalah: Ketika Anda menjalankan skrip penginstal, pesan berikut akan ditampilkan: `configure: error: curses not found`.

Kemungkinan penyebab: Pustaka kontrol terminal **`ncurses`** tidak diinstal pada instans atau server.

Solusi yang Direkomendasikan: Instal pustaka kontrol terminal **`ncurses`** (dan, pada beberapa sistem operasi, pustaka **`glibc-static`** ), lalu coba jalankan skrip penginstal pada instans atau server lagi.

Untuk menginstal pustaka kontrol terminal **`ncurses`** (dan, pada beberapa sistem operasi, pustaka **`glibc-static`** ), jalankan salah satu perintah berikut pada instans atau server Anda:

- Untuk Amazon Linux, Amazon Linux 2, dan Red Hat Enterprise Linux (RHEL) yang berjalan di Amazon EC2: **`sudo yum -y install ncurses-devel`**
- Untuk SUSE: **`sudo zypper install -y ncurses-devel`** dan **`sudo zypper install -y glibc-static`**

## Rentang alamat IP SSH masuk untuk AWS Cloud9

Anda dapat membatasi lalu lintas masuk hanya pada rentang alamat IP yang AWS Cloud9 digunakan untuk menghubungkan melalui SSH ke EC2 instance komputasi AWS awan (misalnya instans Amazon) di VPC Amazon atau server Anda sendiri di jaringan Anda.

> **Note**
>
> Anda dapat membatasi lalu lintas masuk ke hanya rentang alamat IP yang AWS Cloud9 digunakan untuk terhubung melalui SSH. Untuk EC2 lingkungan yang dibuat pada atau setelah 31 Juli 2018, Anda dapat melewati topik ini. Ini karena AWS Cloud9 secara otomatis membatasi lalu lintas SSH masuk untuk lingkungan itu hanya untuk alamat IP yang dijelaskan nanti dalam topik ini. AWS Cloud9 melakukan ini dengan secara otomatis menambahkan aturan ke grup keamanan yang terkait dengan EC2 instans Amazon untuk lingkungan. Aturan ini membatasi lalu lintas SSH masuk melalui port 22 hanya untuk alamat IP untuk Wilayah terkait. AWS Untuk server Anda sendiri di jaringan Anda, Anda masih harus mengikuti langkah-langkah yang dijelaskan nanti dalam topik ini.

Rentang alamat IP untuk sebagian besar AWS Wilayah ada dalam `ip-ranges.json` file, seperti yang dijelaskan dalam Rentang Alamat AWS IP di file Referensi Umum AWS.

> **Note**
>
> Lihat di bawah untuk rentang alamat IP untuk Wilayah Asia Pacific (Hong Kong), Europe (Milan), dan Middle East (Bahrain) yang saat ini tidak disertakan dalam file `ip-ranges.json`.

Untuk menemukan rentang IP dalam file `ip-ranges.json`:

- Untuk Windows, menggunakan AWS Tools for Windows PowerShell, jalankan perintah berikut.

```
Get-AWSPublicIpAddressRange -ServiceKey CLOUD9
```

- Untuk Linux, unduh file ip-ranges.json. Kemudian, Anda dapat membuat kueri dengan menggunakan alat seperti **`jq`** dengan menjalankan perintah berikut.

```
jq '.prefixes[] | select(.service=="CLOUD9")' < ip-ranges.json
```

Rentang IP ini mungkin berubah sesekali. Setiap kali ada perubahan, kami mengirimkan notifikasi kepada pelanggan topik `AmazonIpSpaceChanged`. Untuk mendapatkan notifikasi ini, lihat Pemberitahuan Rentang Alamat AWS IP di Referensi Umum AWS.

Untuk menggunakan rentang alamat IP ini saat mengonfigurasi lingkungan yang menggunakan instance komputasi AWS awan, lihat. Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9 Juga, jika Anda memilih untuk membatasi lalu lintas masuk untuk EC2 lingkungan, atau untuk lingkungan SSH yang terkait dengan EC2 instans Amazon yang menjalankan Amazon Linux atau Ubuntu Server, pastikan untuk juga mengizinkan setidaknya semua alamat IP menggunakan TCP melalui port 32768-61000. Untuk informasi selengkapnya, dan rentang port untuk jenis instans komputasi cloud AWS lainnya, lihat Port ephemeral di Panduan Pengguna Amazon VPC.

Untuk menggunakan rentang alamat IP ini ketika mengonfigurasi lingkungan SSH yang menggunakan jaringan Anda sendiri, lihat dokumentasi untuk jaringan atau administrator jaringan Anda.

## Alamat IP tidak ada di **`ip-ranges.json`**

AWS Cloud9 Rentang alamat IP untuk AWS Wilayah berikut saat ini tidak disediakan dalam `ip-ranges.json` file: Asia Pasifik (Hong Kong), Eropa (Milan), dan Timur Tengah (Bahrain). Tabel berikut mencantumkan rentang IP untuk Wilayah tersebut.

> **Note**
>
> Setiap Wilayah memiliki dua rentang alamat IP untuk mendukung layanan bidang AWS Cloud9 kontrol (routing informasi) dan bidang data (pemrosesan informasi).

| AWS Wilayah | Kode | Rentang IP (notasi CIDR) |
| --- | --- | --- |
| Asia Pasifik (Hong Kong) | `ap-east1` | `18.163.201.96/27`<br>`18.163.139.32/27` |
| Europe (Milan) | `eu-south-1` | `15.161.135.64/27`<br>`15.161.135.96/27` |
| Timur Tengah (Bahrain) | `me-south-1` | `15.185.141.160/27`<br>`15.185.91.32/27` |

# Konten AMI untuk Lingkungan AWS Cloud9 EC2 Pengembangan

Gunakan informasi berikut untuk mendapatkan detail tentang Amazon Machine Images (AMIs) yang AWS Cloud9 digunakan untuk EC2 lingkungan.

> **Important**
>
> Jika EC2 instans Amazon lingkungan Anda didasarkan pada template AMI Amazon Linux 2023 atau Amazon Linux 2 AMI, pembaruan keamanan diinstal pada instans segera setelah diluncurkan. Dan patch keamanan kemudian secara otomatis diterapkan ke instans setiap jam. Pembaruan ini diterapkan oleh proses latar belakang dan tidak memengaruhi penggunaan instans Anda.
> Untuk EC2 lingkungan Ubuntu, pembaruan keamanan juga diinstal pada instance segera setelah diluncurkan. Kemudian paket `unattended-upgrades` secara otomatis menginstal pembaruan yang tersedia setiap hari.

Topik

- Amazon Linux 2023/Amazon Linux 2
- Ubuntu Server

# Amazon Linux 2023/Amazon Linux 2

> **Important**
>
> Kami menyarankan Anda memilih opsi Amazon Linux 2023 saat membuat EC2 lingkungan Amazon menggunakan konsol. Selain menyediakan lingkungan runtime yang aman, stabil, dan berkinerja tinggi, Amazon Linux 2023 AMI menyertakan dukungan jangka panjang hingga 2024.

Untuk menampilkan versi instance Amazon Linux, jalankan perintah berikut dari AWS Cloud9 IDE untuk lingkungan yang terhubung atau dari utilitas SSH seperti ssh perintah atau PuTTY.

```
cat /etc/system-release
```

Untuk menampilkan daftar paket yang diinstal pada instans Amazon Linux, jalankan satu atau beberapa perintah berikut.

Untuk menampilkan semua paket yang terinstal sebagai satu daftar:

```
sudo yum list installed
```

Untuk menampilkan daftar paket yang terinstal dengan nama paket yang berisi teks yang ditentukan:

```
sudo yum list installed | grep YOUR_SEARCH_TERM
```

Pada perintah sebelumnya, ganti `YOUR_SEARCH_TERM` dengan beberapa bagian dari nama paket. Misalnya, untuk menampilkan daftar semua paket yang terinstal dengan nama yang mengandung `sql`:

```
sudo yum list installed | grep sql
```

Untuk menampilkan daftar semua paket yang terinstal, ditampilkan satu halaman pada satu waktu:

```
sudo yum list installed | less
```

Untuk menggulir halaman yang ditampilkan:

- Untuk turun satu baris, tekan **`j`**.

- Untuk naik satu baris, tekan **k**.
- Untuk turun satu halaman, tekan **Ctrl-F**.
- Untuk naik satu halaman, tekan **Ctrl-B**.
- Untuk berhenti, tekan **q**.

> **Note**
>
> Dengan Amazon Linux 2, Anda dapat menggunakan Pustaka Extras untuk menginstal pembaruan aplikasi dan perangkat lunak pada instans Anda. Pembaruan perangkat lunak ini dikenal sebagai topik. Untuk informasi selengkapnya, lihat Pustaka ekstra (Amazon Linux 2) di Panduan EC2 Pengguna Amazon.

Untuk opsi tambahan, jalankan perintah man yum. Lihat juga sumber daya berikut:

- Amazon Linux 2023: Halaman AMI.
- Amazon Linux: Paket Amazon Linux AMI 2018.03.

## Ubuntu Server

Untuk menampilkan versi instans Ubuntu Server, jalankan perintah berikut dari AWS Cloud9 IDE untuk lingkungan yang terhubung atau dari utilitas SSH seperti perintah ssh atau PuTTY.

```
lsb_release -a
```

Versi akan ditampilkan di sebelah bidang Deskripsi.

Untuk menampilkan daftar paket yang diinstal pada Ubuntu Server, jalankan satu atau beberapa perintah berikut.

Untuk menampilkan semua paket yang terinstal sebagai satu daftar:

```
sudo apt list --installed
```

Untuk menampilkan daftar paket yang terinstal dengan nama paket yang berisi teks yang ditentukan:

```
sudo apt list --installed | grep YOUR_SEARCH_TERM
```

Pada perintah sebelumnya, ganti `YOUR_SEARCH_TERM` dengan beberapa bagian dari nama paket. Misalnya, untuk menampilkan daftar semua paket yang terinstal dengan nama yang mengandung `sql`:

```
sudo apt list --installed grep sql
```

Untuk menampilkan daftar semua paket yang terinstal, satu halaman pada satu waktu:

```
sudo apt list --installed | less
```

Untuk menggulir halaman yang ditampilkan:

- Untuk turun satu baris, tekan **`j`**.
- Untuk naik satu baris, tekan **`k`**.
- Untuk turun satu halaman, tekan **`Ctrl-F`**.
- Untuk naik satu halaman, tekan **`Ctrl-B`**.
- Untuk berhenti, tekan **`q`**.

Untuk opsi tambahan, jalankan perintah man apt. Lihat juga Pencarian Paket Ubuntu di situs web Ubuntu.

## Menggunakan peran tertaut layanan untuk AWS Cloud9

AWS Cloud9 menggunakan AWS Identity and Access Management peran terkait layanan (IAM). Peran terkait layanan adalah jenis unik peran IAM yang ditautkan langsung ke. AWS Cloud9 Peran yang terkait dengan layanan ditentukan sebelumnya oleh AWS Cloud9 dan meliputi semua izin yang diperlukan layanan untuk menghubungi layanan AWS lainnya atas nama Anda.

Peran terkait layanan membuat pengaturan AWS Cloud9 lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan. AWS Cloud9 mendefinisikan izin peran terkait layanan, dan hanya AWS Cloud9 dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran hanya setelah terlebih dahulu menghapus sumber daya terkaitnya. Ini melindungi AWS Cloud9 sumber daya Anda karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat Layanan AWS yang bisa digunakan dengan IAM dan carilah layanan yang memiliki opsi Ya di kolom Peran Terkait Layanan. Pilih Ya dengan tautan untuk melihat dokumentasi peran tertaut layanan untuk layanan tersebut.

- Izin peran terkait layanan untuk AWS Cloud9
- Membuat peran terkait layanan untuk AWS Cloud9
- Mengedit peran terkait layanan untuk AWS Cloud9
- Menghapus peran terkait layanan untuk AWS Cloud9
- Wilayah yang Didukung untuk AWS Cloud9 peran terkait layanan

## Izin peran terkait layanan untuk AWS Cloud9

AWS Cloud9 menggunakan peran terkait layanan bernama. AWSService RoleFor AWSCloud9 Peran terkait layanan ini mempercayai layanan `cloud9.amazonaws.com` untuk menjalankan peran.

Kebijakan izin untuk peran terkait layanan ini diberi nama AWSCloud9ServiceRolePolicy, dan memungkinkan AWS Cloud9 untuk menyelesaikan tindakan yang tercantum dalam kebijakan pada sumber daya yang ditentukan.

> **Important**
>
> Jika Anda menggunakan License Manager dan menerima `unable to access your environment` kesalahan, Anda perlu mengganti peran terkait layanan lama dengan versi yang mendukung License Manager. Anda dapat mengganti peran lama hanya dengan menghapusnya. Peran yang diperbarui kemudian dibuat secara otomatis.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ec2:RunInstances",
    "ec2:CreateSecurityGroup",
    "ec2:DescribeVpcs",
    "ec2:DescribeSubnets",
```

```
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeInstances",
    "ec2:DescribeInstanceStatus",
    "cloudformation:CreateStack",
    "cloudformation:DescribeStacks",
    "cloudformation:DescribeStackEvents",
    "cloudformation:DescribeStackResources"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:TerminateInstances",
    "ec2:DeleteSecurityGroup",
    "ec2:AuthorizeSecurityGroupIngress"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "cloudformation:DeleteStack"
   ],
   "Resource": "arn:aws:cloudformation:*:*:stack/aws-cloud9-*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:CreateTags"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ec2:*:*:security-group/*"
   ],
   "Condition": {
    "StringLike": {
     "aws:RequestTag/Name": "aws-cloud9-*"
    }
   }
  },
  {
   "Effect": "Allow",
   "Action": [
```

```
    "ec2:StartInstances",
    "ec2:StopInstances"
   ],
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "ec2:ResourceTag/aws:cloudformation:stack-name": "aws-cloud9-*"
    }
   }
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:StartInstances",
    "ec2:StopInstances"
   ],
   "Resource": [
    "arn:aws:license-manager:*:*:license-configuration:*"
   ]
  },
  {
   "Effect": "Allow",
   "Action": [
    "iam:ListInstanceProfiles",
    "iam:GetInstanceProfile"
   ],
   "Resource": [
    "arn:aws:iam::*:instance-profile/cloud9/*"
   ]
  },
  {
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/service-role/AWSCloud9SSMAccessRole"
   ],
   "Condition": {
    "StringLike": {
     "iam:PassedToService": "ec2.amazonaws.com"
    }
   }
  }
```

```
 ]
}
```

Anda harus mengonfigurasi izin agar memungkinkan AWS Cloud9 untuk membuat peran terkait layanan atas nama entitas IAM (seperti pengguna, grup, atau peran).

AWS Cloud9 Untuk memungkinkan membuat peran AWSService RoleFor AWSCloud9 terkait layanan, tambahkan pernyataan berikut ke kebijakan izin untuk entitas IAM yang namanya AWS Cloud9 perlu membuat peran terkait layanan.

```
{
  "Effect": "Allow",
  "Action": [
    "iam:CreateServiceLinkedRole"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": "cloud9.amazonaws.com"
    }
  }
}
```

Atau, Anda dapat menambahkan kebijakan yang terkelola AWS `AWSCloud9User` atau `AWSCloud9Administrator` untuk entitas IAM.

Untuk mengizinkan entitas IAM menghapus peran AWSService RoleFor AWSCloud9 terkait layanan, tambahkan pernyataan berikut ke kebijakan izin untuk entitas IAM yang perlu menghapus peran terkait layanan.

```
{
  "Effect": "Allow",
  "Action": [
    "iam:DeleteServiceLinkedRole",
    "iam:GetServiceLinkedRoleDeletionStatus"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": "cloud9.amazonaws.com"
    }
```

```
    }
}
```

## Membuat peran terkait layanan untuk AWS Cloud9

Anda tidak perlu membuat peran terkait layanan. Saat Anda membuat lingkungan AWS Cloud9 pengembangan, AWS Cloud9 ciptakan peran terkait layanan untuk Anda.

## Mengedit peran terkait layanan untuk AWS Cloud9

Anda tidak dapat mengedit peran AWSService RoleFor AWSCloud9 terkait layanan di. AWS Cloud9 Misalnya, setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit Peran Tertaut Layanan dalam Panduan Pengguna IAM.

## Menghapus peran terkait layanan untuk AWS Cloud9

Jika Anda tidak lagi memerlukan penggunaan fitur atau layanan yang memerlukan peran terkait layanan, kami menyarankan Anda untuk menghapus peran tersebut. Dengan begitu Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif.

### Menghapus peran terkait layanan di IAM

Sebelum dapat menggunakan IAM untuk menghapus peran terkait layanan, Anda harus menghapus semua sumber daya AWS Cloud9 yang digunakan oleh peran tersebut. Untuk menghapus AWS Cloud9 sumber daya, lihat Menghapus Lingkungan.

Anda dapat menggunakan konsol IAM untuk menghapus peran AWSService RoleFor AWSCloud9 terkait layanan. Untuk informasi selengkapnya, silakan lihat Menghapus Peran Terkait Layanan di Panduan Pengguna IAM.

## Wilayah yang Didukung untuk AWS Cloud9 peran terkait layanan

AWS Cloud9 mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia. Untuk informasi selengkapnya, lihat AWS Cloud9 di Referensi Umum Amazon Web Services.

# Pencatatan panggilan AWS Cloud9 API dengan AWS CloudTrail

AWS Cloud9 terintegrasi dengan CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS dalam AWS Cloud9. CloudTrail menangkap semua panggilan API untuk AWS Cloud9 sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari AWS Cloud9 konsol dan dari panggilan kode ke AWS Cloud9 APIs. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon Simple Storage Service (Amazon S3), termasuk acara untuk. AWS Cloud9 Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat AWS Cloud9, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

## AWS Cloud9 informasi di CloudTrail

CloudTrail diaktifkan pada Akun AWS saat Anda membuat akun. Ketika aktivitas terjadi di AWS Cloud9, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di akun AWS Anda. Untuk informasi lain, lihat Melihat Peristiwa dengan Riwayat Peristiwa CloudTrail.

Untuk catatan acara yang sedang berlangsung di Anda Akun AWS, termasuk acara untuk AWS Cloud9, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di AWS partisi dan mengirimkan file log ke bucket S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- Gambaran Umum untuk Membuat Jejak
- CloudTrail Layanan dan Integrasi yang Didukung
- Mengkonfigurasi Notifikasi Amazon SNS untuk CloudTrail
- Menerima File CloudTrail Log dari Beberapa Wilayah dan Menerima File CloudTrail Log dari Beberapa Akun

AWS Cloud9 mendukung pencatatan tindakan berikut sebagai peristiwa dalam file CloudTrail log:

- `CreateEnvironmentEC2`
- `CreateEnvironmentSSH`
- `CreateEnvironmentMembership`
- `DeleteEnvironment`
- `DeleteEnvironmentMembership`
- `DescribeEnvironmentMemberships`
- `DescribeEnvironments`
- `DescribeEnvironmentStatus`
- `ListEnvironments`
- `ListTagsForResource`
- `TagResource`
- `UntagResource`
- `UpdateEnvironment`
- `UpdateEnvironmentMembership`

> **Note**
>
> Beberapa CloudTrail peristiwa untuk AWS Cloud9 tidak disebabkan oleh operasi API publik. Sebaliknya, peristiwa berikut dimulai oleh pembaruan internal yang memengaruhi autentikasi pengguna dan dikelola kredensial sementara:
>
> - `DisableManagedCredentialsByCollaborator`
> - `EnvironmentTokenSuccessfullyCreated`
> - `ManagedCredentialsUpdatedOnEnvironment`

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut ini:

- Apakah permintaan dibuat dengan kredenal pengguna root atau AWS Identity and Access Management IAM.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.

- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

Untuk informasi lain, lihat Elemen userIdentity CloudTrail .

## Memahami entri file AWS Cloud9 log

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, dan parameter permintaan. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

- CreateEnvironmentEC2
- CreateEnvironmentSSH
- CreateEnvironmentMembership
- DeleteEnvironment
- DeleteEnvironmentMembership
- DescribeEnvironmentMemberships
- DescribeEnvironments
- DescribeEnvironmentStatus
- ListEnvironments
- ListTagsForResource
- TagResource
- UntagResource
- UpdateEnvironment
- UpdateEnvironmentMembership

### CreateEnvironmentEC2

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `CreateEnvironmentEC2` tindakan.

```
{
  "Records": [
    {
```

```
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "CreateEnvironmentEC2",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "instanceType": "t2.small",
        "subnetId": "subnet-1d4a9eEX",
        "description": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "dryRun": true,
        "automaticStopTimeMinutes": 30,
        "name": "my-test-environment",
        "clientRequestToken": "cloud9-console-f8e37272-e541-435d-a567-5c684EXAMPLE"
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## CreateEnvironmentSSH

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `CreateEnvironmentSSH` tindakan.


```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "CreateEnvironmentSSH",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "host": "198.51.100.0",
        "port": 22,
        "name": "my-ssh-environment",
        "description": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "clientRequestToken": "cloud9-console-b015a0e9-469e-43e3-be90-6f432EXAMPLE",
        "loginName": "ec2-user"
      },
      "responseElements": {
        "environmentId": "5c39cc4a85d74a8bbb6e23ed6EXAMPLE"
      },
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
```

```
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## CreateEnvironmentMembership

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `CreateEnvironmentMembership` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "CreateEnvironmentMembership",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
        "userArn": "arn:aws:iam::111122223333:user/MyUser",
        "permissions": "read-write"
      },
      "responseElements": {
        "membership": {
```

```
          "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
          "permissions": "read-write",
          "userId": "AIDACKCEVSQ6C2EXAMPLE",
          "userArn": "arn:aws:iam::111122223333:user/MyUser"
        }
      },
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## DeleteEnvironment

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `DeleteEnvironment` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "DeleteEnvironment",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
```

```
      "requestParameters": {
        "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE"
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## DeleteEnvironmentMembership

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `DeleteEnvironmentMembership` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "DeleteEnvironmentMembership",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
```

```
        "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
        "userArn": "arn:aws:iam::111122223333:user/MyUser",
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## DescribeEnvironmentMemberships

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `DescribeEnvironmentMemberships` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "DescribeEnvironmentMemberships",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
```

```
        "nextToken": "NEXT_TOKEN_EXAMPLE",
        "permissions": [ "owner" ],
        "maxResults": 15
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## DescribeEnvironments

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `DescribeEnvironments` tindakan.

```
{
   "Records": [
     {
       "eventVersion": "1.05",
       "userIdentity": {
         "type": "IAMUser",
         "principalId": "AIDACKCEVSQ6C2EXAMPLE",
         "arn": "arn:aws:iam::111122223333:user/MyUser",
         "accountId": "111122223333",
         "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
         "userName": "MyUser",
         "sessionContext": {
           "attributes": {
             "mfaAuthenticated": "false",
             "creationDate": "2019-01-14T11:29:47Z"
           }
         },
         "invokedBy": "signin.amazonaws.com"
       },
       "eventTime": "2019-01-14T11:33:27Z",
       "eventSource": "cloud9.amazonaws.com",
       "eventName": "DescribeEnvironments",
       "awsRegion": "us-west-2",
       "sourceIPAddress": "192.0.2.0",
```

```
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "environmentIds": [
          "2f5ff70a640f49398f67e3bdeb811ab2"
        ]
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

## DescribeEnvironmentStatus

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `DescribeEnvironmentStatus` tindakan.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::123456789012:myuser_role",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:sts::123456789012:myuser_role",
                "accountId": "123456789012",
                "userName": "barshane_role"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-03-12T15:10:54Z"
            }
        }
```

```
    },
    "eventTime": "2021-03-12T15:13:31Z",
    "eventSource": "cloud9.amazonaws.com",
    "eventName": "DescribeEnvironmentStatus",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "XX.XX.XXX.XX",
    "userAgent": "aws-internal/3 aws-sdk-java/1.11.951
 Linux/4.9.230-0.1.ac.223.84.332.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.282-b08
 java/1.8.0_282 vendor/Oracle_Corporation",
    "requestParameters": {
        "environmentId": "31ea8a12746a4221b7d8e07d9ef6ee21"
    },
    "responseElements": null,
    "requestID": "68b163fb-aa88-4f40-bafd-4a18bf24cbd5",
    "eventID": "c0fc52a9-7331-4ad0-a8ee-157995dfb5e6",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

## ListEnvironments

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ListEnvironments` tindakan.

```
{
   "Records": [
     {
       "eventVersion": "1.05",
       "userIdentity": {
         "type": "IAMUser",
         "principalId": "AIDACKCEVSQ6C2EXAMPLE",
         "arn": "arn:aws:iam::111122223333:user/MyUser",
         "accountId": "111122223333",
         "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
         "userName": "MyUser",
         "sessionContext": {
           "attributes": {
             "mfaAuthenticated": "false",
             "creationDate": "2019-01-14T11:29:47Z"
           }
         },
```

```
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "ListEnvironments",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "nextToken": "NEXT_TOKEN_EXAMPLE",
        "maxResults": 15
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    }
  ]
}
```

## ListTagsForResource

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ListTagsForResource` tindakan.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::123456789012:myuser_role",
        "accountId": "123456789012",
        "accessKeyId": "AIDACKCEVSQ6C2EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "123456789012:myuser_role",
                "accountId": "123456789012",
                "userName": "barshane_role"
            },
```

```
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-03-23T16:41:51Z"
            }
        }
    },
    "eventTime": "2021-03-23T16:42:58Z",
    "eventSource": "cloud9.amazonaws.com",
    "eventName": "ListTagsForResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "XX.XX.XXX.XX",
    "userAgent": "aws-internal/3 aws-sdk-java/1.11.976
 Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.282-b08
 java/1.8.0_282 vendor/Oracle_Corporation cfg/retry-mode/legacy",
    "requestParameters": {
        "resourceARN": "arn:aws:cloud9:us-
east-1:123456789012:environment:3XXXXXXXXX6a4221b7d8e07d9ef6ee21"
    },
    "responseElements": {
        "tags": "HIDDEN_DUE_TO_SECURITY_REASONS"
    },
    "requestID": "5750a344-8462-4020-82f9-f1d500a75162",
    "eventID": "188d572d-9a14-4082-b98b-0389964c7c30",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```


## TagResource

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `TagResource` tindakan.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts:: 123456789012:myuser_role",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
```

```
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/myuser_role",
                "accountId": "123456789012",
                "userName": "MyUser"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-03-23T15:03:57Z"
            }
        }
    },
    "eventTime": "2021-03-23T15:08:16Z",
    "eventSource": "cloud9.amazonaws.com",
    "eventName": "TagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "54.XXX.XXX.XXX",
    "userAgent": "aws-internal/3 aws-sdk-java/1.11.976
 Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.282-b08
 java/1.8.0_282 vendor/Oracle_Corporation cfg/retry-mode/legacy",
    "requestParameters": {
        "resourceARN": "arn:aws:cloud9:us-
east-1:123456789012:environment:3XXXXXXXXX6a4221b7d8e07d9ef6ee21",
        "tags": "HIDDEN_DUE_TO_SECURITY_REASONS"
    },
    "responseElements": null,
    "requestID": "658e9d70-91c2-41b8-9a69-c6b4cc6a9456",
    "eventID": "022b2893-73d1-44cb-be6f-d3faa68e83b1",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

## UntagResource

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `UntagResource` tindakan.

```
{
```

```
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:sts::123456789012/MyUser",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:MyUser",
                "accountId": "123456789012",
                "userName": "MyUser"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-03-23T15:58:36Z"
            }
        }
    },
    "eventTime": "2021-03-23T16:05:08Z",
    "eventSource": "cloud9.amazonaws.com",
    "eventName": "UntagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "3.XX.XX.XXX",
    "userAgent": "aws-internal/3 aws-sdk-java/1.11.976
 Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.282-b08
 java/1.8.0_282 vendor/Oracle_Corporation cfg/retry-mode/legacy",
    "requestParameters": {
        "resourceARN": "arn:aws:cloud9:us-
east-1:123456789012:environment:3XXXXXXXXX6a4221b7d8e07d9ef6ee21",
        "tagKeys": "HIDDEN_DUE_TO_SECURITY_REASONS"
    },
    "responseElements": null,
    "requestID": "0eadaef3-dc0a-4cd7-85f6-135b8529f75f",
    "eventID": "41f2f2e2-4b17-43d4-96fc-9857981ca1de",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
```

```
}
```

## UpdateEnvironment

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `UpdateEnvironment` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "UpdateEnvironment",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
        "description": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "name": "my-test-environment-renamed"
      },
      "responseElements": null,
      "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
      "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
```

```
}
```

## UpdateEnvironmentMembership

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `UpdateEnvironmentMembership` tindakan.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/MyUser",
        "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "MyUser",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-01-14T11:29:47Z"
          }
        },
        "invokedBy": "signin.amazonaws.com"
      },
      "eventTime": "2019-01-14T11:33:27Z",
      "eventSource": "cloud9.amazonaws.com",
      "eventName": "UpdateEnvironmentMembership",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "signin.amazonaws.com",
      "requestParameters": {
        "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
        "userArn": "arn:aws:iam::111122223333:user/MyUser",
        "permissions": "read-only"
      },
      "responseElements": {
        "membership": {
          "environmentId": "2f5ff70a640f49398f67e3bdeEXAMPLE",
          "permissions": "read-only",
          "userId": "AIDACKCEVSQ6C2EXAMPLE",
          "userArn": "arn:aws:iam::111122223333:user/MyUser"
```

```
            }
        },
        "requestID": "f0e629fb-fd37-49bd-b2cc-e9822EXAMPLE",
        "eventID": "8a906445-1b2a-47e9-8d7c-5b242EXAMPLE",
        "eventType": "AwsApiCall",
        "recipientAccountId": "111122223333"
      }
    ]}
```

# Tanda

Tag adalah label atau atribut yang Anda atau AWS lampirkan ke AWS sumber daya. Setiap tanda terdiri atas kunci dan nilai pasangan. Anda dapat menggunakan tag untuk mengontrol akses ke AWS Cloud9 sumber daya Anda, seperti yang dijelaskan dalam Akses Kontrol Menggunakan Tag AWS Sumber Daya di Panduan Pengguna IAM. Tanda juga dapat membantu Anda mengelola informasi penagihan, seperti yang dijelaskan dalam Tanda Alokasi Biaya Buatan Pengguna.

Saat Anda membuat lingkungan AWS Cloud9 EC2 pengembangan, AWS Cloud9 sertakan tag sistem tertentu yang dibutuhkan untuk mengelola lingkungan. Tanda sistem dimulai dengan "aws:". Selama proses pembuatan itu, Anda juga dapat menambahkan tag sumber daya Anda sendiri.

Setelah lingkungan dibuat, Anda dapat melihat tanda yang dilampirkan ke lingkungan, menambahkan tanda sumber daya baru ke lingkungan, atau mengubah atau menghapus tanda yang Anda tambahkan sebelumnya. Anda dapat melampirkan hingga 50 tag yang ditentukan pengguna ke lingkungan AWS Cloud9 .

Tampilkan atau perbarui tanda dengan menggunakan satu atau beberapa metode berikut.

- Di konsol AWS Cloud9, pilih lingkungan yang Anda minati, lalu pilih Lihat Detail.

- Gunakan perintah AWS Cloud9 CLI berikut: `list-tags-for-resource`, `tag-resource`, dan. `untag-resource`
- Gunakan tindakan AWS Cloud9 API berikut: ListTagsForResource, TagResource, dan UntagResource.

> **Warning**
>
> Tag yang Anda buat atau perbarui AWS Cloud9 dengan menggunakan metode sebelumnya tidak secara otomatis disebarkan ke sumber daya yang mendasarinya. Untuk informasi tentang cara melakukannya, lihat bagian berikutnya, Menyebarkan pembaruan tanda ke sumber daya yang mendasari.

## Menyebarkan pembaruan tanda ke sumber daya yang mendasari

Saat Anda menggunakan perintah AWS Cloud9 CLI atau tindakan API untuk menambahkan, memodifikasi, atau menghapus tag yang dilampirkan ke AWS Cloud9 lingkungan, perubahan tersebut tidak secara otomatis disebarkan ke sumber daya dasar seperti AWS CloudFormation tumpukan, EC2 instans Amazon, dan grup keamanan Amazon EC2 . Anda harus secara manual menyebarkan perubahan tersebut.

Agar lebih mudah menggunakan prosedur berikut, Anda dapat memperoleh ID lingkungan untuk lingkungan yang Anda minati. Jika Anda ingin menggunakanyan, ikuti langkah-langkah ini:

1. Di konsol AWS Cloud9, pilih lingkungan yang Anda minati, lalu pilih Lihat Detail.
2. Cari properti ARN Lingkungan dan catat ID lingkungan, yang merupakan bagian dari ARN lingkungan setelah "lingkungan:".

Anda perlu menyebarkan pembaruan tanda ke satu atau beberapa lokasi berikut, tergantung pada apa yang akan Anda gunakan untuk tanda tersebut.

### Menyebarkan pembaruan tag ke tumpukan AWS CloudFormation

> **Note**
>
> Saat Anda memperbarui tag ke AWS CloudFormation tumpukan, pembaruan tersebut secara otomatis disebarkan ke EC2 instans Amazon dan grup EC2 keamanan Amazon yang terkait dengan tumpukan.

1. Navigasikan ke konsol AWS CloudFormation tersebut.
2. Temukan dan pilih tumpukan yang sesuai dengan AWS Cloud9 lingkungan yang Anda minati. Jika Anda mencatat ID lingkungan, Anda dapat menggunakannya untuk memfilter lingkungan.

3. Pada tab Info Tumpukan, di bagian Tag, tinjau daftar tanda.
4. Jika Anda perlu memperbarui tanda, pilih Perbarui di dekat bagian atas halaman, dan ikuti petunjuknya. Untuk informasi selengkapnya, lihat Memperbarui Tumpukan Secara Langsung di Panduan Pengguna AWS CloudFormation.

Anda juga dapat memperbarui tanda dengan menggunakan Perintah CLI `describe-stacks` dan `update-stack`.

### Menyebarkan pembaruan tag ke instans Amazon EC2

1. Arahkan ke konsol EC2Instans Amazon.
2. Temukan dan pilih EC2 instans Amazon yang sesuai dengan AWS Cloud9 lingkungan yang Anda minati. Jika Anda mencatat ID lingkungan sebelumnya, Anda dapat menggunakannya untuk memfilter lingkungan.
3. Pada tab Tanda, lihat dan perbarui tanda seperlunya.

Anda juga dapat memperbarui tag menggunakan perintah CLI deskripsi-tag, buat-tag, dan hapus-tag.

### Menyebarkan pembaruan tag ke grup EC2 keamanan Amazon

1. Arahkan ke konsol Grup EC2 Keamanan Amazon.
2. Temukan dan pilih grup keamanan yang sesuai dengan AWS Cloud9 lingkungan yang Anda minati. Jika Anda mencatat ID lingkungan sebelumnya, Anda dapat menggunakannya untuk memfilter lingkungan.
3. Buka tab Tanda untuk melihat dan memperbarui tanda seperlunya.

Anda juga dapat memperbarui tag menggunakan perintah CLI deskripsi-tag, buat-tag, dan hapus-tag.

# Keamanan untuk AWS Cloud9

Keamanan cloud di Amazon Web Services (AWS) merupakan prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan. Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model Tanggung Jawab Bersama menggambarkan ini sebagai Keamanan dari Cloud dan Keamanan dalam Cloud.

Security of the Cloud - AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan semua layanan yang ditawarkan di AWS Cloud dan memberi Anda layanan yang dapat Anda gunakan dengan aman. Tanggung jawab keamanan kami adalah prioritas tertinggi di AWS, dan efektivitas keamanan kami secara teratur diuji dan diverifikasi oleh auditor pihak ketiga sebagai bagian dari Program AWS Kepatuhan.

Keamanan di Cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan, dan faktor-faktor lain termasuk sensitivitas data Anda, persyaratan organisasi Anda, serta undang-undang dan peraturan yang berlaku.

AWS Cloud9 mengikuti model tanggung jawab bersama melalui AWS layanan khusus yang didukungnya. Untuk informasi keamanan AWS layanan, lihat halaman dokumentasi keamanan AWS layanan dan AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan.

Topik berikut menunjukkan cara mengonfigurasi AWS Cloud9 untuk memenuhi tujuan keamanan dan kepatuhan Anda.

Topik


- Perlindungan data di AWS Cloud9
- Identity and Access Management untuk AWS Cloud9
- Penebangan dan pemantauan di AWS Cloud9
- Validasi kepatuhan untuk AWS Cloud9
- Ketahanan di AWS Cloud9
- Keamanan infrastruktur di AWS Cloud9
- Pembaruan perangkat lunak dan patch
- Praktik terbaik keamanan untuk AWS Cloud9

# Perlindungan data di AWS Cloud9

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di AWS Cloud9. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensil dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AWS Cloud9 atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau AWS SDKs. Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan

atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

# Enkripsi data

Enkripsi data mengacu pada perlindungan data saat dalam perjalanan, karena melakukan perjalanan antara AWS Cloud9 dan AWS akun Anda dan saat diam saat disimpan di penyimpanan AWS Cloud9 konfigurasi dan instance AWS komputasi awan.

Dalam konteks AWS Cloud9, jenis data berikut mungkin memerlukan perlindungan melalui enkripsi:

Konten dan data Anda

Informasi yang Anda manipulasi, kumpulkan, dan simpan. Berikut ini adalah contoh dari tipe data ini:

- File kode Anda
- Konfigurasi, aplikasi, dan data untuk lingkungan terlampir atau EC2 lingkungan SSH

AWS Cloud9 metadata

Data yang AWS Cloud9 memanipulasi, mengumpulkan, dan menyimpan. Berikut ini adalah contoh dari tipe data ini:

- Pengaturan IDE seperti status tab, file terbuka, dan preferensi IDE
- AWS Cloud9 metadata lingkungan pengembangan seperti nama dan deskripsi lingkungan
- AWS Cloud9 API layanan, dan log konsol
- Log layanan seperti permintaan HTTP

AWS Cloud9 juga mentransmisikan beberapa konten dan data Anda melalui layanan pesawat datanya. Ini termasuk file Anda, input terminal, teks output, dan beberapa perintah IDE (misalnya, untuk menyimpan file).

## Enkripsi diam

Enkripsi saat istirahat didefinisikan sebagai perlindungan data dari akses tidak sah dengan mengenkripsi data saat disimpan. Setiap data pelanggan yang disimpan dalam AWS Cloud9

lingkungan seperti file kode, paket, atau dependensi selalu disimpan dalam sumber daya pelanggan. Jika pelanggan menggunakan EC2 lingkungan Amazon, data disimpan dalam volume Amazon Elastic Block Store (Amazon EBS) terkait yang ada di akun mereka. AWS Jika pelanggan menggunakan lingkungan SSH, data disimpan di penyimpanan lokal di server Linux mereka.

Saat EC2 instans Amazon dibuat untuk lingkungan AWS Cloud9 pengembangan, volume Amazon EBS yang tidak terenkripsi dibuat dan dilampirkan ke instance tersebut. Pelanggan yang ingin mengenkripsi data mereka perlu membuat volume EBS terenkripsi dan melampirkannya ke instance. EC2 AWS Cloud9 dan volume Amazon EBS terlampir mendukung enkripsi default Amazon EBS, yang merupakan pengaturan khusus Wilayah secara default. Untuk informasi selengkapnya, lihat Enkripsi secara default di Panduan Pengguna Cloud Komputasi AWS Elastis.

Metadata tentang lingkungan AWS Cloud9 pengembangan, seperti nama lingkungan, anggota lingkungan, dan pengaturan IDE, disimpan oleh AWS, bukan dalam sumber daya pelanggan. Informasi khusus pelanggan, seperti deskripsi lingkungan dan pengaturan IDE, dienkripsi.

## Enkripsi bergerak

Enkripsi dalam transit didefinisikan sebagai perlindungan data dari intersepsi saat data ditransfer antar-endpoint komunikasi. Semua data yang dikirimkan antara klien pelanggan dan AWS Cloud9 layanan dienkripsi melalui HTTPS, WSS, dan SSH terenkripsi.

- HTTPS — Memastikan permintaan aman antara browser web pelanggan dan AWS Cloud9 layanan. AWS Cloud9 juga memuat aset dari Amazon yang CloudFront dikirim melalui HTTPS dari browser pelanggan.
- WSS (WebSocket Aman) — Memungkinkan komunikasi dua arah yang aman melalui WebSockets antara browser web pelanggan dan layanan. AWS Cloud9
- SSH Terenkripsi (Secure Shell): Memungkinkan transmisi data yang aman antara browser web klien dan layanan. AWS Cloud9

Penggunaan protokol HTTPS, WSS, dan SSH tergantung pada penggunaan browser yang didukung oleh. AWS Cloud9 Lihat Browser yang didukung untuk AWS Cloud9.

> **Note**
>
> Protokol enkripsi diimplementasikan secara default di AWS Cloud9. Pelanggan tidak dapat mengubah encryption-in-transit pengaturan.

### Manajemen kunci

AWS Key Management Service (AWS KMS) adalah layanan terkelola untuk membuat dan mengendalikan AWS KMS keys, kunci enkripsi yang digunakan untuk mengenkripsi data pelanggan. AWS Cloud9 menghasilkan dan mengelola kunci kriptografi untuk mengenkripsi data atas nama pelanggan.

### Privasi lalu lintas antarjaringan

Lingkungan SSH terhubung ke komputasi dan penyimpanan lokal milik pelanggan. Koneksi SSH, HTTPS, dan WSS yang dienkripsi mendukung transit data antara layanan dan lingkungan SSH.

Anda dapat mengonfigurasi lingkungan AWS Cloud9 EC2 pengembangan (didukung oleh EC2 instans Amazon) untuk diluncurkan dalam subnet VPCs dan spesifik. Untuk informasi selengkapnya tentang pengaturan Amazon Virtual Private Cloud, lihat Pengaturan VPC untuk Lingkungan Pengembangan AWS Cloud9.

# Identity and Access Management untuk AWS Cloud9

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS Cloud9 IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik


- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Bagaimana AWS Cloud9 bekerja dengan IAM
- Contoh kebijakan berbasis identitas untuk AWS Cloud9
- Memecahkan masalah AWS Cloud9 identitas dan akses
- Bagaimana AWS Cloud9 bekerja dengan sumber daya dan operasi IAM
- AWS kebijakan terkelola untuk AWS Cloud9
- Membuat kebijakan terkelola pelanggan untuk AWS Cloud9

- AWS Cloud9 referensi izin
- AWS kredensi sementara yang dikelola

## Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan. AWS Cloud9

Pengguna layanan — Jika Anda menggunakan AWS Cloud9 layanan untuk melakukan pekerjaan Anda, maka administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak AWS Cloud9 fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara mengelola akses dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di AWS Cloud9, lihat Memecahkan masalah AWS Cloud9 identitas dan akses.

Administrator layanan — Jika Anda bertanggung jawab atas AWS Cloud9 sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke AWS Cloud9. Tugas Anda adalah menentukan AWS Cloud9 fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM AWS Cloud9, lihatBagaimana AWS Cloud9 bekerja dengan IAM.

Administrator IAM – Jika Anda adalah administrator IAM, Anda mungkin ingin belajar dengan lebih detail tentang cara Anda menulis kebijakan untuk mengelola akses ke AWS Cloud9. Untuk melihat contoh kebijakan AWS Cloud9 berbasis identitas yang dapat Anda gunakan di IAM, lihat. Contoh kebijakan berbasis identitas untuk AWS Cloud9

## Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensi yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (IAM Identity Center), autentikasi masuk tunggal perusahaan Anda, dan kredensi Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas terfederasi, administrator Anda sebelumnya

menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS.

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensil Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Guna mengetahui informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat Autentikasi multi-faktor dalam Panduan Pengguna AWS IAM Identity Center dan Autentikasi multi-faktor AWS di IAM dalam Panduan Pengguna IAM.

### Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

### Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensil yang disediakan melalui sumber identitas. Ketika

identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensi sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat Identitas IAM, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat Identitas IAM, lihat Apakah itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

## Pengguna dan grup IAM

Pengguna IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, kami merekomendasikan untuk mengandalkan kredensial sementara, bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan tertentu yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami merekomendasikan Anda merotasi kunci akses. Untuk informasi selengkapnya, lihat Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang dalam Panduan Pengguna IAM.

Grup IAM adalah identitas yang menentukan sekumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat meminta kelompok untuk menyebutkan IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

## Peran IAM

Peran IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Untuk mengambil peran IAM sementara AWS Management Console, Anda dapat beralih dari pengguna ke peran IAM (konsol). Anda dapat mengambil peran dengan memanggil operasi AWS CLI atau AWS API atau dengan menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat Buat peran untuk penyedia identitas pihak ketiga dalam Panduan Pengguna IAM. Jika menggunakan Pusat Identitas IAM, Anda harus mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM akan mengorelasikan set izin ke peran dalam IAM. Untuk informasi tentang set izin, lihat Set izin dalam Panduan Pengguna AWS IAM Identity Center .
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (prinsipal tepercaya) di akun lain untuk mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.
- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
  - Sesi akses teruskan (FAS) — Saat Anda menggunakan pengguna IAM atau peran untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.
  - Peran layanan – Peran layanan adalah peran IAM yang dijalankan oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

  - Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke peran layanan. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan peran IAM untuk mengelola kredensi sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS permintaan API. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instans yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensi sementara. Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon di Panduan Pengguna IAM.

## Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasinya. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut bisa mendapatkan informasi peran dari AWS Management Console, API AWS CLI, atau AWS API.

## Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam. Akun AWS Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan yang dikelola atau kebijakan inline, lihat Pilih antara kebijakan yang dikelola dan kebijakan inline dalam Panduan Pengguna IAM.

## Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

## Daftar kontrol akses (ACLs)

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung. ACLs Untuk mempelajari selengkapnya ACLs, lihat Ringkasan daftar kontrol akses (ACL) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

## Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan tempat Anda mengatur izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas ke entitas IAM (pengguna IAM atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batasan izin, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — SCPs adalah kebijakan JSON yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di. AWS Organizations AWS Organizations adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas di akun anggota, termasuk masing-masing. Pengguna root akun AWS Untuk informasi selengkapnya tentang Organizations dan SCPs, lihat Kebijakan kontrol layanan di Panduan AWS Organizations Pengguna.
- Kebijakan kontrol sumber daya (RCPs) — RCPs adalah kebijakan JSON yang dapat Anda gunakan untuk menetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda tanpa memperbarui kebijakan IAM yang dilampirkan ke setiap sumber daya yang Anda miliki. RCP membatasi izin untuk sumber daya di akun anggota dan dapat memengaruhi izin efektif untuk identitas, termasuk Pengguna root akun AWS, terlepas dari apakah itu milik organisasi Anda. Untuk informasi selengkapnya tentang Organizations dan RCPs, termasuk daftar dukungan Layanan AWS tersebut RCPs, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

### Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

## Bagaimana AWS Cloud9 bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses AWS Cloud9, pelajari fitur IAM yang tersedia untuk digunakan. AWS Cloud9

Fitur IAM yang dapat Anda gunakan AWS Cloud9

| Fitur IAM | AWS Cloud9 dukungan |
|---|---|
| Kebijakan berbasis identitas | Ya |
| Kebijakan berbasis sumber daya | Tidak |
| Tindakan kebijakan | Ya |
| Sumber daya kebijakan | Ya |
| kunci-kunci persyaratan kebijakan (spesifik layanan) | Ya |
| ACLs | Tidak |
| ABAC (tanda dalam kebijakan) | Ya |
| Kredensial sementara | Ya |
| Sesi akses teruskan (FAS) | Ya |
| Peran layanan | Ya |
| Peran terkait layanan | Ya |

Untuk mendapatkan tampilan tingkat tinggi tentang cara AWS Cloud9 dan AWS layanan lain bekerja dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

## Kebijakan berbasis identitas untuk AWS Cloud9

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Anda tidak dapat menentukan secara spesifik prinsipal dalam sebuah kebijakan berbasis identitas karena prinsipal berlaku bagi pengguna atau peran yang melekat kepadanya. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk AWS Cloud9

Untuk melihat contoh kebijakan AWS Cloud9 berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Cloud9

## Kebijakan berbasis sumber daya dalam AWS Cloud9

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai prinsipal dalam kebijakan berbasis sumber daya. Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika prinsipal dan sumber daya berbeda Akun AWS, administrator IAM di akun tepercaya juga harus memberikan izin entitas utama (pengguna atau peran) untuk mengakses sumber daya. Mereka memberikan izin dengan melampirkan kebijakan berbasis identitas kepada entitas. Namun, jika kebijakan berbasis sumber daya memberikan akses ke principal dalam akun yang sama, tidak diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

AWS Cloud9 tidak mendukung kebijakan berbasis sumber daya tetapi Anda masih dapat mengontrol izin sumber daya AWS Cloud9 lingkungan untuk anggota AWS Cloud9 lingkungan melalui API dan IDE. AWS Cloud9 AWS Cloud9

## Tindakan kebijakan untuk AWS Cloud9

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan operasi AWS API terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar AWS Cloud9 tindakan, lihat Tindakan yang ditentukan oleh AWS Cloud9 dalam Referensi Otorisasi Layanan.

Tindakan kebijakan AWS Cloud9 menggunakan awalan berikut sebelum tindakan:

```
account
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
      "account:action1",
      "account:action2"
         ]
```

Untuk melihat contoh kebijakan AWS Cloud9 berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Cloud9

## Sumber daya kebijakan untuk AWS Cloud9

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Untuk melihat daftar jenis sumber daya dan jenis AWS Cloud9 sumber daya ARNs, lihat Sumber daya yang ditentukan oleh AWS Cloud9 dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan yang dapat menentukan ARN setiap sumber daya, lihat Tindakan yang ditentukan oleh AWS Cloud9.

Untuk melihat contoh kebijakan AWS Cloud9 berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Cloud9

## Kunci kondisi kebijakan untuk AWS Cloud9

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Condition` (atau blok `Condition`) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam sebuah pernyataan, atau beberapa kunci dalam elemen `Condition` tunggal, maka AWS akan mengevaluasinya menggunakan operasi `AND` logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan `OR` operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Sebagai contoh, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tanda yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, lihat Elemen kebijakan IAM: variabel dan tanda dalam Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci AWS Cloud9 kondisi, lihat Kunci kondisi untuk AWS Cloud9 dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang ditentukan oleh AWS Cloud9.

Untuk melihat contoh kebijakan AWS Cloud9 berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Cloud9

## ACLs di AWS Cloud9

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

## ABAC dengan AWS Cloud9

Mendukung ABAC (tanda dalam kebijakan): Ya

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut. Dalam AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke entitas IAM (pengguna atau peran) dan ke banyak AWS sumber daya. Penandaan ke entitas dan sumber daya adalah langkah pertama dari ABAC. Kemudian rancanglah kebijakan ABAC untuk mengizinkan operasi ketika tanda milik prinsipal cocok dengan tanda yang ada di sumber daya yang ingin diakses.

ABAC sangat berguna di lingkungan yang berkembang dengan cepat dan berguna di situasi saat manajemen kebijakan menjadi rumit.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

## Menggunakan kredensi sementara dengan AWS Cloud9

Mendukung kredensial sementara: Ya

Beberapa Layanan AWS tidak berfungsi saat Anda masuk menggunakan kredensi sementara. Untuk informasi tambahan, termasuk yang Layanan AWS bekerja dengan kredensi sementara, lihat Layanan AWS yang bekerja dengan IAM di Panduan Pengguna IAM.

Anda menggunakan kredensi sementara jika Anda masuk AWS Management Console menggunakan metode apa pun kecuali nama pengguna dan kata sandi. Misalnya, ketika Anda mengakses AWS menggunakan tautan masuk tunggal (SSO) perusahaan Anda, proses tersebut secara otomatis membuat kredensil sementara. Anda juga akan secara otomatis membuat kredensial sementara ketika Anda masuk ke konsol sebagai seorang pengguna lalu beralih peran. Untuk informasi selengkapnya tentang peralihan peran, lihat Beralih dari pengguna ke peran IAM (konsol) dalam Panduan Pengguna IAM.

Anda dapat membuat kredensil sementara secara manual menggunakan API AWS CLI atau AWS . Anda kemudian dapat menggunakan kredensi sementara tersebut untuk mengakses. AWS AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM.

## Teruskan sesi akses untuk AWS Cloud9

Mendukung sesi akses maju (FAS): Ya

Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.

## Peran layanan untuk AWS Cloud9

Mendukung peran layanan: Ya

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak AWS Cloud9 fungsionalitas. Edit peran layanan hanya jika AWS Cloud9 memberikan panduan untuk melakukannya.

## Peran terkait layanan untuk AWS Cloud9

Mendukung peran terkait layanan: Ya

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul

di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

## Contoh kebijakan berbasis identitas untuk AWS Cloud9

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau mengubah sumber daya AWS Cloud9 . Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh AWS Cloud9, termasuk format ARNs untuk setiap jenis sumber daya, lihat Kunci tindakan, sumber daya, dan kondisi untuk AWS Cloud9 dalam Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- Menggunakan konsol AWS Cloud9
- Mengizinkan pengguna melihat izin mereka sendiri

### Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus AWS Cloud9 sumber daya di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun

AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.

- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

## Menggunakan konsol AWS Cloud9

Untuk mengakses AWS Cloud9 konsol, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang AWS Cloud9 sumber daya di Anda Akun AWS. Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan AWS Cloud9 konsol, lampirkan juga kebijakan AWS Cloud9 *ConsoleAccess* atau *ReadOnly* AWS terkelola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

## Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
```

```
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

## Memecahkan masalah AWS Cloud9 identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS Cloud9 dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di AWS Cloud9
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS Cloud9 sumber daya saya

### Saya tidak berwenang untuk melakukan tindakan di AWS Cloud9

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `awes:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 awes:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `awes:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

## Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS Cloud9.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di AWS Cloud9. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

## Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS Cloud9 sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah AWS Cloud9 mendukung fitur-fitur ini, lihatBagaimana AWS Cloud9 bekerja dengan IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

## Bagaimana AWS Cloud9 bekerja dengan sumber daya dan operasi IAM

AWS Identity and Access Management digunakan untuk mengelola izin yang memungkinkan Anda untuk bekerja dengan lingkungan AWS Cloud9 pengembangan dan lainnya Layanan AWS dan sumber daya.

### AWS Cloud9 sumber daya dan operasi

Pada tahun AWS Cloud9, sumber daya utama adalah lingkungan AWS Cloud9 pembangunan. Dalam kebijakan, Anda menggunakan Amazon Resource Name (ARN) untuk mengidentifikasi sumber daya yang mengikuti kebijakan tersebut. Tabel berikut mencantumkan lingkungan ARNs. Untuk informasi selengkapnya, lihat Amazon Resource Names (ARNs) dan Ruang Nama AWS Layanan di bagian. Referensi Umum Amazon Web Services

| Jenis sumber daya | Format ARN |
| --- | --- |
| Lingkungan | `arn:aws:cloud9: REGION_ID :ACCOUNT_ID :environment: ENVIRONMENT_ID` |
| Setiap lingkungan yang dimiliki oleh akun yang ditentukan dalam yang ditentukan Wilayah AWS | `arn:aws:cloud9: REGION_ID :ACCOUNT_ID :environment:*` |

| Jenis sumber daya | Format ARN |
| --- | --- |
| Setiap lingkungan yang dimiliki oleh akun yang ditentukan di Wilayah yang ditentukan | `arn:aws:cloud9: REGION_ID :ACCOUNT_ID :*` |
| Setiap AWS Cloud9 sumber daya, terlepas dari akun dan Wilayah | `arn:aws:cloud9:*` |

Misalnya, Anda dapat menunjukkan lingkungan tertentu dalam pernyataan Anda menggunakan Nama Sumber Daya Amazon (ARN), sebagai berikut.

```
"Resource": "arn:aws:cloud9:us-
east-2:123456789012:environment:70d899206236474f9590d93b7c41dfEX"
```

Untuk menentukan semua sumber daya, gunakan karakter wildcard (*) dalam `Resource` elemen.

```
"Resource": "*"
```

Untuk menentukan beberapa sumber daya dalam satu pernyataan, pisahkan Amazon Resource Names (ARNs) dengan koma.

```
"Resource": [
  "arn:aws:cloud9:us-east-2:123456789012:environment:70d899206236474f9590d93b7c41dfEX",
  "arn:aws:cloud9:us-east-2:123456789012:environment:81e900317347585a0601e04c8d52eaEX"
]
```

AWS Cloud9 menyediakan satu set operasi untuk bekerja dengan AWS Cloud9 sumber daya. Untuk daftar, lihat AWS Cloud9 referensi izin.

## Memahami kepemilikan sumber daya

Akun AWS Akun memiliki sumber daya yang dibuat di akun, terlepas dari siapa yang membuat sumber daya.

Pertimbangkan kasus penggunaan dan skenario berikut:

- Misalkan Anda menggunakan kredensi akun root Anda Akun AWS untuk membuat lingkungan AWS Cloud9 pengembangan. Meskipun memungkinkan, ini tidak direkomendasikan. Dalam hal ini, Anda Akun AWS adalah pemilik lingkungan.

- Misalkan Anda membuat pengguna IAM di Anda Akun AWS dan Anda memberikan izin untuk membuat lingkungan bagi pengguna tersebut. Kemudian, pengguna dapat membuat lingkungan. Namun, milik Anda Akun AWS, yang menjadi milik pengguna, masih memiliki lingkungan.
- Misalkan Anda membuat peran IAM di Anda Akun AWS dengan izin untuk membuat lingkungan. Kemudian, siapa pun yang dapat mengambil peran dapat menciptakan lingkungan. Anda Akun AWS, yang menjadi milik perannya, memiliki lingkungan.

> **Note**
>
> Jika Anda menghapus akun pengguna yang merupakan pemilik ARN dari satu atau beberapa AWS Cloud9 lingkungan, lingkungan ini tidak akan memiliki pemilik. Solusi untuk skenario ini adalah dengan menggunakan AWS Cloud9 SDK untuk menambahkan pengguna IAM lain dengan hak baca dan tulis menggunakan `CreateEnvironmentMembership` tindakan, dan tipe data. `EnvironmentMember` Setelah Anda menambahkan pengguna IAM ini, Anda dapat menyalin file lingkungan ke AWS Cloud9 lingkungan baru dan menjadikan pemilik ini pemilik ARN. Untuk informasi selengkapnya tentang tindakan ini, lihat CreateEnvironmentMembership, dan untuk informasi selengkapnya tentang tipe data ini, lihat EnvironmentMemberdi Panduan Referensi AWS Cloud9 API.

## Mengelola akses ke sumber daya

Kebijakan izin menjelaskan siapa yang memiliki akses ke sumber daya mana.

> **Note**
>
> Bagian ini membahas penggunaan IAM di AWS Cloud9. Bagian ini tidak memberikan informasi yang mendetail tentang layanan IAM. Untuk dokumentasi IAM lengkap, lihat Apa yang Dimaksud dengan IAM? dalam Panduan Pengguna IAM. Untuk informasi tentang sintaks dan deskripsi kebijakan IAM, lihat Referensi Kebijakan JSON IAM dalam Panduan Pengguna IAM.

Kebijakan yang terlampir pada identitas IAM disebut sebagai kebijakan berbasis identitas (atau kebijakan IAM). Kebijakan yang melekat pada sumber daya disebut sebagai kebijakan berbasis sumber daya. AWS Cloud9 mendukung kebijakan berbasis identitas dan berbasis sumber daya.

Setiap tindakan API berikut hanya memerlukan kebijakan IAM untuk dilampirkan ke identitas IAM yang ingin memanggil tindakan API ini:

- `CreateEnvironmentEC2`
- `DescribeEnvironments`

Tindakan API berikut memerlukan kebijakan berbasis sumber daya. Kebijakan IAM tidak diperlukan, tetapi AWS Cloud9 menggunakan kebijakan IAM jika dilampirkan ke identitas IAM yang ingin memanggil tindakan API ini. Kebijakan berbasis sumber daya harus diterapkan pada sumber daya yang diinginkan: AWS Cloud9

- `CreateEnvironmentMembership`
- `DeleteEnvironment`
- `DeleteEnvironmentMembership`
- `DescribeEnvironmentMemberships`
- `DescribeEnvironmentStatus`
- `UpdateEnvironment`
- `UpdateEnvironmentMembership`

Untuk informasi selengkapnya tentang apa yang dilakukan masing-masing tindakan API ini, lihat Referensi AWS Cloud9 API.

Anda tidak dapat melampirkan kebijakan berbasis sumber daya ke sumber daya secara langsung. AWS Cloud9 Sebagai gantinya, AWS Cloud9 lampirkan kebijakan berbasis sumber daya yang sesuai ke sumber AWS Cloud9 daya saat Anda menambahkan, memodifikasi, memperbarui, atau menghapus anggota lingkungan.

Untuk memberikan izin pengguna untuk melakukan tindakan pada AWS Cloud9 sumber daya, Anda melampirkan kebijakan izin ke grup IAM yang menjadi milik pengguna tersebut. Kami menyarankan Anda melampirkan kebijakan AWS terkelola (yang telah ditentukan sebelumnya) AWS Cloud9 bila memungkinkan. AWS kebijakan terkelola berisi kumpulan izin akses yang telah ditentukan sebelumnya untuk skenario penggunaan umum dan jenis pengguna, seperti administrasi penuh lingkungan, pengguna lingkungan, dan pengguna yang hanya memiliki akses hanya-baca ke lingkungan. Untuk daftar kebijakan AWS terkelola AWS Cloud9, lihat[AWS kebijakan terkelola untuk AWS Cloud9](#).

Untuk skenario penggunaan yang lebih mendetail dan tipe pengguna yang unik, Anda dapat membuat dan melampirkan kebijakan terkelola pelanggan Anda sendiri. Lihat Opsi pengaturan tambahan untuk AWS Cloud9 dan Membuat kebijakan terkelola pelanggan untuk AWS Cloud9.

Untuk melampirkan kebijakan IAM (AWS dikelola atau dikelola pelanggan) ke identitas IAM, lihat Melampirkan Kebijakan IAM (Konsol) di Panduan Pengguna IAM.

### Izin sesi untuk operasi API

Saat menggunakan AWS CLI atau AWS API untuk membuat sesi sementara untuk peran atau pengguna gabungan secara terprogram, Anda dapat meneruskan kebijakan sesi sebagai parameter untuk memperluas cakupan sesi peran. Ini berarti bahwa izin sesi yang efektif merupakan titik temu antara kebijakan berbasis identitas peran dan kebijakan sesi.

Saat permintaan dibuat untuk mengakses sumber daya selama sesi, jika tidak ada pernyataan `Deny` yang berlaku tetapi juga tidak ada pernyataan `Allow` yang berlaku dalam kebijakan sesi, hasil evaluasi kebijakan adalah penolakan implisit. (Untuk informasi selengkapnya, lihat Menentukan apakah permintaan diizinkan atau ditolak dalam akun di Panduan Pengguna IAM.)

Namun, untuk operasi AWS Cloud9 API yang memerlukan kebijakan berbasis sumber daya (lihat di atas), izin diberikan kepada entitas IAM yang memanggil jika ditentukan sebagai kebijakan sumber daya. `Principal` Izin eksplisit ini lebih diutamakan daripada penolakan implisit dari kebijakan sesi, sehingga memungkinkan sesi untuk memanggil operasi API dengan sukses. AWS Cloud9

## AWS kebijakan terkelola untuk AWS Cloud9

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa kebijakan AWS terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk digunakan semua pelanggan. AWS Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Jika AWS memperbarui izin yang ditentukan dalam kebijakan AWS terkelola, pemutakhiran akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat Kebijakan terkelola AWS dalam Panduan Pengguna IAM.

## AWS kebijakan terkelola: AWSCloud9 Administrator

Anda dapat melampirkan kebijakan `AWSCloud9Administrator` ke identitas IAM Anda.

Kebijakan ini memberikan *`administrative`* izin yang menyediakan akses administrator. AWS Cloud9

Detail izin

Kebijakan ini mencakup izin berikut.

- AWS Cloud9 Semua AWS Cloud9 tindakan dalam diri mereka Akun AWS.
- Amazon EC2 — Dapatkan informasi tentang beberapa VPC Amazon dan sumber daya subnet di dalamnya. Akun AWS
- IAM — Dapatkan informasi tentang pengguna IAM di dalamnya Akun AWS, dan buat peran AWS Cloud9 terkait layanan sesuai kebutuhan mereka. Akun AWS
- Systems Manager- Memungkinkan pengguna StartSession untuk memanggil untuk memulai koneksi ke instance untuk sesi Session Manager. Izin ini diperlukan untuk pengguna yang membuka lingkungan yang berkomunikasi dengan EC2 instance-nya melalui Systems Manager. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:*",
                "iam:GetUser",
                "iam:ListUsers",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeInstanceTypeOfferings",
                "ec2:DescribeRouteTables"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:GetConnectionStatus"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        }
    ]
}
```

## AWS kebijakan terkelola: AWSCloud9 Pengguna

Anda dapat melampirkan kebijakan `AWSCloud9User` ke identitas IAM Anda.

Kebijakan ini memberikan *user* izin untuk menciptakan lingkungan AWS Cloud9 pengembangan dan mengelola lingkungan yang dimiliki.

Detail izin

Kebijakan ini mencakup izin berikut.

- AWS Cloud9 — Buat dan dapatkan informasi tentang lingkungan mereka, dan dapatkan dan ubah pengaturan pengguna untuk lingkungan mereka.
- Amazon EC2 — Dapatkan informasi tentang beberapa VPC Amazon dan sumber daya subnet di dalamnya. Akun AWS
- IAM — Dapatkan informasi tentang pengguna IAM di dalamnya Akun AWS, dan buat peran AWS Cloud9 terkait layanan sesuai kebutuhan mereka. Akun AWS
- Systems Manager- Memungkinkan pengguna StartSession untuk memanggil untuk memulai koneksi ke instance untuk sesi Session Manager. Izin ini diperlukan untuk pengguna yang membuka lingkungan yang berkomunikasi dengan EC2 instance-nya melalui Systems Manager. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:UpdateUserSettings",
                "cloud9:GetUserSettings",
                "iam:GetUser",
                "iam:ListUsers",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeInstanceTypeOfferings",
                "ec2:DescribeRouteTables"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:CreateEnvironmentEC2",
```

```
                "cloud9:CreateEnvironmentSSH"
            ],
            "Resource": "*",
            "Condition": {
                "Null": {
                    "cloud9:OwnerArn": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:GetUserPublicKey"
            ],
            "Resource": "*",
            "Condition": {
                "Null": {
                    "cloud9:UserArn": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:DescribeEnvironmentMemberships"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null": {
                    "cloud9:UserArn": "true",
                    "cloud9:EnvironmentId": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
```

```
                    "iam:AWSServiceName": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:GetConnectionStatus"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        }
    ]
}
```

## AWS kebijakan terkelola: AWSCloud9 EnvironmentMember

Anda dapat melampirkan kebijakan `AWSCloud9EnvironmentMember` ke identitas IAM Anda.

Kebijakan ini memberikan *membership* izin yang menyediakan kemampuan untuk bergabung dengan lingkungan AWS Cloud9 bersama.

Detail izin

Kebijakan ini mencakup izin berikut:

- AWS Cloud9 — Dapatkan informasi tentang lingkungan mereka, dan dapatkan dan ubah pengaturan pengguna untuk lingkungan mereka.
- IAM — Dapatkan informasi tentang pengguna IAM di dalamnya Akun AWS, dan buat peran AWS Cloud9 terkait layanan sesuai kebutuhan mereka. Akun AWS
- Systems Manager- Memungkinkan pengguna StartSession untuk memanggil untuk memulai koneksi ke instance untuk sesi Session Manager. Izin ini diperlukan untuk pengguna yang membuka lingkungan yang berkomunikasi dengan EC2 instance-nya melalui Systems Manager. Untuk informasi selengkapnya, lihat Mengakses instance no-ingress EC2 dengan AWS Systems Manager

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:GetUserSettings",
                "cloud9:UpdateUserSettings",
                "iam:GetUser",
                "iam:ListUsers"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:DescribeEnvironmentMemberships"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null": {
                    "cloud9:UserArn": "true",
                    "cloud9:EnvironmentId": "true"
                }
            }
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "ssm:StartSession",
                "ssm:GetConnectionStatus"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        }
    ]
}
```

## AWS kebijakan terkelola: **AWSCloud9ServiceRolePolicy**

Peran terkait layanan AWSServiceRoleForAWSCloud9menggunakan kebijakan ini untuk memungkinkan AWS Cloud9 lingkungan berinteraksi dengan Amazon EC2 dan AWS CloudFormation sumber daya.

Detail izin

AWSCloud9ServiceRolePolicyMemberikan izin AWSService RoleFor AWSCloud9 yang diperlukan AWS Cloud9 untuk memungkinkan berinteraksi dengan ( Layanan AWS Amazon EC2 dan AWS CloudFormation) yang diperlukan untuk membuat dan menjalankan lingkungan pengembangan.

AWS Cloud9 mendefinisikan izin peran terkait layanan, dan hanya AWS Cloud9 dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Untuk informasi selengkapnya tentang cara AWS Cloud9 menggunakan peran terkait layanan, lihat. [Menggunakan peran tertaut layanan untuk AWS Cloud9]

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ec2:RunInstances",
    "ec2:CreateSecurityGroup",
    "ec2:DescribeVpcs",
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeInstances",
    "ec2:DescribeInstanceStatus",
    "cloudformation:CreateStack",
    "cloudformation:DescribeStacks",
    "cloudformation:DescribeStackEvents",
    "cloudformation:DescribeStackResources"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:TerminateInstances",
    "ec2:DeleteSecurityGroup",
    "ec2:AuthorizeSecurityGroupIngress"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "cloudformation:DeleteStack"
   ],
   "Resource": "arn:aws:cloudformation:*:*:stack/aws-cloud9-*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:CreateTags"
```

```
    ],
    "Resource": [
     "arn:aws:ec2:*:*:instance/*",
     "arn:aws:ec2:*:*:security-group/*"
    ],
    "Condition": {
     "StringLike": {
      "aws:RequestTag/Name": "aws-cloud9-*"
     }
    }
   },
   {
    "Effect": "Allow",
    "Action": [
     "ec2:StartInstances",
     "ec2:StopInstances"
    ],
    "Resource": "*",
    "Condition": {
     "StringLike": {
      "ec2:ResourceTag/aws:cloudformation:stack-name": "aws-cloud9-*"
     }
    }
   },
   {
    "Effect": "Allow",
    "Action": [
     "ec2:StartInstances",
     "ec2:StopInstances"
    ],
    "Resource": [
     "arn:aws:license-manager:*:*:license-configuration:*"
    ]
   },
   {
    "Effect": "Allow",
    "Action": [
     "iam:ListInstanceProfiles",
     "iam:GetInstanceProfile"
    ],
    "Resource": [
     "arn:aws:iam::*:instance-profile/cloud9/*"
    ]
   },
```

```
  {
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/service-role/AWSCloud9SSMAccessRole"
   ],
   "Condition": {
    "StringLike": {
     "iam:PassedToService": "ec2.amazonaws.com"
    }
   }
  }
 ]
}
```

## AWS Cloud9 pembaruan kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola AWS Cloud9 sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman Riwayat AWS Cloud9 dokumen.

| Perubahan | Deskripsi | Tanggal |
| --- | --- | --- |
| Tindakan baru telah ditambahkan ke AWSCloud9 Pengguna, AWSCloud9 Administrator, dan AWSCloud9EnvironmentMemberkebijakan. | `ssm:GetConnectionStatus` Tindakan telah ditambahkan ke AWSCloud9 Administrator AWSCloud9 Pengguna dan AWSCloud9 EnvironmentMemberkebijakan. Tindakan ini akan memberi pengguna izin untuk memeriksa status koneksi SSM. `cloud9:ValidateEnvironmentName` API telah dihapus dari kebijakan AWSCloud9 | 12 Oktober 2023 |

| Perubahan | Deskripsi | Tanggal |
| --- | --- | --- |
| | Pengguna karena sudah usang. | |
| API ditambahkan ke kebijakan AWSCloud9Pengguna dan AWSCloud9Administrator. | Dua API baru telah ditambahkan ke kebijakan AWSCloud9 Pengguna dan AWSCloud9 Administrator, API ini adalah `ec2:DescribeInstanceTypeOfferings` dan`ec2:DescribeRouteTables` . Tujuan dari API ini adalah untuk memungkinkan AWS Cloud9 untuk memvalidasi bahwa subnet default mendukung jenis instance yang dipilih oleh pelanggan saat mereka membuat AWS Cloud9 lingkungan. | Agustus 02, 2023 |
| Perbarui ke AWSCloud9 ServiceRolePolicy | AWSCloud9ServiceRolePolicytelah diperbarui untuk memungkinkan AWS Cloud9 untuk memulai dan menghentikan EC2 instans Amazon yang dikelola oleh konfigurasi lisensi License Manager. | 12 Januari 2022 |
| AWS Cloud9 mulai melacak perubahan | AWS Cloud9 mulai melacak perubahan untuk kebijakan AWS terkelolanya. | 15 Maret 2021 |

## Membuat kebijakan terkelola pelanggan untuk AWS Cloud9

Jika tidak ada kebijakan AWS terkelola yang memenuhi persyaratan kontrol akses, Anda dapat membuat dan melampirkan kebijakan yang dikelola pelanggan Anda sendiri.

Untuk membuat kebijakan terkelola pelanggan, lihat Membuat Kebijakan IAM (konsol) di Panduan Pengguna IAM.

Topik

- Menentukan elemen kebijakan: efek, prinsipal, tindakan, dan sumber daya
- Contoh kebijakan yang dikelola pelanggan

## Menentukan elemen kebijakan: efek, prinsipal, tindakan, dan sumber daya

Untuk setiap AWS Cloud9 sumber daya, layanan mendefinisikan satu set operasi API. Untuk memberikan izin untuk operasi API ini, AWS Cloud9 tentukan serangkaian tindakan yang dapat Anda tentukan dalam kebijakan.

Berikut ini adalah elemen-elemen kebijakan dasar:

- `Effect` – Anda menetapkan efek, baik mengizinkan atau menolak, ketika pengguna meminta tindakan. Jika Anda tidak secara eksplisit memberikan akses ke (mengizinkan) sumber daya, akses akan ditolak secara implisit. Anda juga dapat secara eksplisit menolak akses ke sumber daya. Anda dapat melakukan ini untuk memastikan pengguna tidak dapat mengakses sumber daya, bahkan jika kebijakan yang berbeda memberikan akses.
- `Principal` – Dalam kebijakan berbasis identitas (kebijakan IAM), pengguna yang kebijakannya terlampir adalah prinsipal implisit. Untuk kebijakan berbasis sumber daya, Anda menentukan pengguna, akun, layanan, atau entitas lain yang ingin Anda terima izinnya.
- `Resource`— Gunakan Nama Sumber Daya Amazon (ARN) untuk mengidentifikasi sumber daya yang berlaku untuk kebijakan tersebut.
- `Action` – Gunakan kata kunci tindakan untuk mengidentifikasi operasi sumber daya yang ingin Anda izinkan atau tolak. Misalnya, izin `cloud9:CreateEnvironmentEC2` memberikan pengguna izin untuk melakukan operasi `CreateEnvironmentEC2`.

Untuk mempelajari lebih lanjut tentang sintaks dan deskripsi kebijakan IAM, lihat Referensi Kebijakan JSON IAM dalam Panduan Pengguna IAM.

Untuk tabel yang menunjukkan semua tindakan AWS Cloud9 API dan sumber daya yang diterapkan, lihatAWS Cloud9 referensi izin.

## Contoh kebijakan yang dikelola pelanggan

Dalam bagian ini, Anda dapat menemukan contoh kebijakan yang memberikan izin untuk tindakan AWS Cloud9 . Anda dapat menyesuaikan contoh kebijakan IAM berikut untuk mengizinkan atau secara eksplisit menolak akses AWS Cloud9 untuk identitas IAM Anda.

Untuk membuat atau melampirkan kebijakan yang dikelola pelanggan ke identitas IAM, lihat Membuat Kebijakan IAM (Konsol) dan Melampirkan Kebijakan IAM (Konsol) di Panduan Pengguna IAM.

> **Note**
>
> Contoh berikut menggunakan US East (Ohio) Region (`us-east-2`), Akun AWS ID fiktif (`123456789012`), dan ID lingkungan AWS Cloud9 pengembangan fiktif (). `81e900317347585a0601e04c8d52eaEX`

Topik

- Mendapatkan informasi tentang lingkungan
- Buat EC2 lingkungan
- Buat EC2 lingkungan dengan jenis EC2 instans Amazon tertentu
- Buat EC2 lingkungan di subnet VPC Amazon tertentu
- Buat EC2 lingkungan dengan nama lingkungan tertentu
- Membuat lingkungan SSH saja
- Memperbarui lingkungan atau mencegah memperbarui lingkungan
- Mendapatkan daftar anggota lingkungan
- Berbagi lingkungan hanya dengan pengguna tertentu
- Mencegah berbagi lingkungan
- Mengubah, atau mencegah perubahan, pengaturan anggota lingkungan
- Menghapus, atau mencegah penghapusan, anggota lingkungan
- Menghapus, atau mencegah penghapusan, lingkungan
- Kebijakan IAM khusus untuk pembuatan lingkungan SSM

### Mendapatkan informasi tentang lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, memungkinkan entitas tersebut untuk mendapatkan informasi tentang lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:DescribeEnvironments",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS `AWSCloud9Administrator` terkelola dan. `AWSCloud9User`

### Buat EC2 lingkungan

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan ke entitas IAM, memungkinkan entitas tersebut untuk membuat lingkungan AWS Cloud9 EC2 pengembangan di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS `AWSCloud9Administrator` terkelola dan. `AWSCloud9User`

Buat EC2 lingkungan dengan jenis EC2 instans Amazon tertentu

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan ke entitas IAM, memungkinkan entitas tersebut untuk membuat lingkungan AWS Cloud9 EC2 pengembangan di akun mereka. Namun, EC2 lingkungan hanya dapat menggunakan kelas tertentu dari jenis EC2 instans Amazon.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "cloud9:InstanceType": "t3.*"
        }
      }
    }
  ]
}
```

> **Note**
>
> Jika kebijakan AWS terkelola `AWSCloud9Administrator` atau `AWSCloud9User` sudah dilampirkan ke entitas IAM, kebijakan AWS terkelola itu mengesampingkan perilaku pernyataan kebijakan IAM sebelumnya. Ini karena kebijakan yang AWS dikelola tersebut lebih permisif.

Buat EC2 lingkungan di subnet VPC Amazon tertentu

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan ke entitas IAM, memungkinkan entitas tersebut untuk membuat lingkungan AWS Cloud9 EC2 pengembangan di akun mereka. Namun, EC2 lingkungan hanya dapat menggunakan subnet VPC Amazon yang ditentukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "cloud9:SubnetId": [
            "subnet-12345678",
            "subnet-23456789"
          ]
        }
      }
    }
  ]
}
```

> **Note**
>
> Jika kebijakan AWS terkelola `AWSCloud9Administrator` atau `AWSCloud9User` sudah dilampirkan ke entitas IAM, kebijakan AWS terkelola itu mengesampingkan perilaku pernyataan kebijakan IAM sebelumnya. Ini karena kebijakan yang AWS dikelola tersebut lebih permisif.

Buat EC2 lingkungan dengan nama lingkungan tertentu

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan pada entitas IAM, memungkinkan entitas tersebut untuk menciptakan lingkungan AWS Cloud9 EC2 pengembangan di akun mereka. Namun, EC2 lingkungan hanya dapat menggunakan nama yang ditentukan.

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloud9:EnvironmentName": "my-demo-environment"
        }
      }
    }
  ]
}
```

> **Note**
>
> Jika kebijakan AWS terkelola `AWSCloud9Administrator` atau `AWSCloud9User` sudah dilampirkan ke entitas IAM, kebijakan AWS terkelola itu mengesampingkan perilaku pernyataan kebijakan IAM sebelumnya. Ini karena kebijakan yang AWS dikelola tersebut lebih permisif.

### Membuat lingkungan SSH saja

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan ke entitas IAM, memungkinkan entitas tersebut untuk membuat lingkungan pengembangan AWS Cloud9 SSH di akun mereka. Namun, entitas tidak dapat membuat lingkungan AWS Cloud9 EC2 pengembangan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:CreateEnvironmentSSH",
      "Resource": "*"
    },
    {
      "Effect": "Deny",
      "Action": "cloud9:CreateEnvironmentEC2",
      "Resource": "*"
    }
  ]
```

```
}
```

Memperbarui lingkungan atau mencegah memperbarui lingkungan

Contoh berikut pernyataan kebijakan IAM, yang dilampirkan pada entitas IAM, memungkinkan entitas tersebut untuk mengubah informasi tentang lingkungan AWS Cloud9 pengembangan apa pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:UpdateEnvironment",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator`

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, secara eksplisit mencegah entitas tersebut mengubah informasi tentang lingkungan dengan Nama Sumber Daya Amazon (ARN) yang ditentukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "cloud9:UpdateEnvironment",
      "Resource": "arn:aws:cloud9:us-
east-2:123456789012:environment:81e900317347585a0601e04c8d52eaEX"
    }
  ]
}
```

Mendapatkan daftar anggota lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mengizinkan entitas tersebut mendapatkan daftar anggota untuk lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:DescribeEnvironmentMemberships",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator` Selain itu, izin akses sebelumnya lebih permisif daripada izin akses yang setara dalam kebijakan AWS terkelola. `AWSCloud9User`

Berbagi lingkungan hanya dengan pengguna tertentu

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mengizinkan entitas tersebut berbagi lingkungan mana pun di akun mereka hanya dengan pengguna tertentu.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:CreateEnvironmentMembership"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloud9:UserArn": "arn:aws:iam::123456789012:user/MyDemoUser"
        }
```

```
      }
    }
  ]
}
```

> **Note**
>
> Jika kebijakan AWS terkelola `AWSCloud9Administrator` atau `AWSCloud9User` sudah dilampirkan ke entitas IAM, kebijakan yang AWS dikelola tersebut mengesampingkan perilaku pernyataan kebijakan IAM sebelumnya. Ini karena kebijakan yang AWS dikelola tersebut lebih permisif.

Mencegah berbagi lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mencegah entitas tersebut berbagi lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "cloud9:CreateEnvironmentMembership",
        "cloud9:UpdateEnvironmentMembership"
      ],
      "Resource": "*"
    }
  ]
}
```

Mengubah, atau mencegah perubahan, pengaturan anggota lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mengizinkan entitas tersebut mengubah pengaturan anggota di lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Action": "cloud9:UpdateEnvironmentMembership",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator`

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, secara eksplisit mencegah entitas tersebut mengubah pengaturan anggota di lingkungan dengan Nama Sumber Daya Amazon (ARN) yang ditentukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "cloud9:UpdateEnvironmentMembership",
      "Resource": "arn:aws:cloud9:us-
east-2:123456789012:environment:81e900317347585a0601e04c8d52eaEX"
    }
  ]
}
```

Menghapus, atau mencegah penghapusan, anggota lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mengizinkan entitas tersebut menghapus anggota mana pun dari lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:DeleteEnvironmentMembership",
      "Resource": "*"
```

```
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator`

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, secara eksplisit mencegah entitas tersebut menghapus anggota mana pun dari lingkungan dengan Nama Sumber Daya Amazon (ARN) yang ditentukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "cloud9:DeleteEnvironmentMembership",
      "Resource": "arn:aws:cloud9:us-
east-2:123456789012:environment:81e900317347585a0601e04c8d52eaEX"
    }
  ]
}
```

Menghapus, atau mencegah penghapusan, lingkungan

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, mengizinkan entitas tersebut menghapus lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "cloud9:DeleteEnvironment",
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator`

Contoh pernyataan kebijakan IAM berikut, yang dilampirkan ke entitas IAM, secara eksplisit mencegah entitas tersebut menghapus lingkungan dengan Nama Sumber Daya Amazon (ARN) yang ditentukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "cloud9:DeleteEnvironment",
      "Resource": "arn:aws:cloud9:us-
east-2:123456789012:environment:81e900317347585a0601e04c8d52eaEX"
    }
  ]
}
```

Kebijakan IAM khusus untuk pembuatan lingkungan SSM

Ada masalah izin saat ini yang terjadi saat membuat lingkungan SSM dengan `AWSCloud9Administrator` atau `AWSCloud9User` kebijakan yang dilampirkan. Contoh berikut pernyataan kebijakan IAM, ketika dilampirkan ke entitas IAM, memungkinkan pengguna untuk melampirkan dan menggunakan kebijakan AWS `AWSCloud9Administrator` terkelola atau. `AWSCloud9User`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:UpdateUserSettings",
                "cloud9:GetUserSettings",
                "iam:GetUser",
                "iam:ListUsers",
                "iam:ListRoles",
```

```
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeRouteTables"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:CreateEnvironmentEC2",
                "cloud9:CreateEnvironmentSSH"
            ],
            "Resource": "*",
            "Condition": {
                "Null": {
                    "cloud9:OwnerArn": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:GetUserPublicKey"
            ],
            "Resource": "*",
            "Condition": {
                "Null": {
                    "cloud9:UserArn": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloud9:DescribeEnvironmentMemberships"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null": {
                    "cloud9:UserArn": "true",
                    "cloud9:EnvironmentId": "true"
                }
```

```
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:cloud9:environment": "*"
                },
                "StringEquals": {
                    "aws:CalledViaFirst": "cloud9.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": ["iam:ListInstanceProfilesForRole", "iam:CreateRole"],
            "Resource": ["arn:aws:iam::*:role/service-role/AWSCloud9SSMAccessRole"]
        },
        {
            "Effect": "Allow",
            "Action": ["iam:AttachRolePolicy"],
```

```
            "Resource": ["arn:aws:iam::*:role/service-role/AWSCloud9SSMAccessRole"],
            "Condition": {
                "StringEquals": {
                    "iam:PolicyARN": "arn:aws:iam::aws:policy/
AWSCloud9SSMInstanceProfile"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/service-role/AWSCloud9SSMAccessRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ec2.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateInstanceProfile",
                "iam:AddRoleToInstanceProfile"
            ],
            "Resource": [
                "arn:aws:iam::*:instance-profile/cloud9/AWSCloud9SSMInstanceProfile"
            ]
        }
    ]
}
```

## AWS Cloud9 referensi izin

Anda dapat menggunakan kunci kondisi AWS lebar dalam AWS Cloud9 kebijakan Anda untuk menyatakan kondisi. Untuk daftar, lihat Elemen Kebijakan IAM JSON: Persyaratan dalam Panduan Pengguna IAM.

Anda menentukan tindakan di bidang `Action` kebijakan. Untuk menentukan tindakan, gunakan awalan `cloud9:` diikuti dengan nama operasi API (misalnya, `"Action": "cloud9:DescribeEnvironments"`). Untuk menentukan beberapa tindakan dalam satu pernyataan, pisahkan dengan koma (misalnya, `"Action": [ "cloud9:UpdateEnvironment", "cloud9:DeleteEnvironment" ]`).

## Menggunakan karakter wildcard

Anda menentukan ARN, dengan atau tanpa karakter wildcard (*), sebagai nilai sumber daya di bidang `Resource` kebijakan. Anda dapat menggunakan wildcard untuk menentukan beberapa tindakan atau sumber daya. Misalnya, `cloud9:*` menentukan semua AWS Cloud9 tindakan dan `cloud9:Describe*` menentukan semua AWS Cloud9 tindakan yang dimulai dengan. `Describe`

Contoh berikut mengizinkan entitas IAM untuk mendapatkan informasi tentang lingkungan dan keanggotaan lingkungan untuk lingkungan mana pun di akun mereka.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloud9:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```

> **Note**
>
> Izin akses sebelumnya sudah termasuk dalam kebijakan AWS terkelola. `AWSCloud9Administrator` Selain itu, izin akses sebelumnya lebih permisif daripada izin akses yang setara dalam kebijakan terkelola. AWS `AWSCloud9User`

## AWS Cloud9 Operasi API dan izin yang diperlukan untuk tindakan

> **Note**
>
> Anda dapat menggunakan tabel di bawah ini sebagai referensi saat menyiapkan kontrol akses dan menulis kebijakan izin untuk dilampirkan ke identitas IAM (kebijakan berbasis identitas).
> Public API operationsTabel mencantumkan operasi API yang dapat dipanggil oleh pelanggan yang menggunakan SDKs dan AWS Command Line Interface.

Permission-only API operationsDaftar operasi API yang tidak langsung dapat dipanggil oleh kode pelanggan atau. AWS Command Line Interface Tetapi pengguna IAM memang memerlukan izin untuk operasi ini yang dipanggil saat AWS Cloud9 tindakan dilakukan menggunakan konsol.

Operasi API publik

| AWS Cloud9 operasi | Izin yang diperlukan (tindakan API) | Sumber Daya |
|---|---|---|
| `CreateEnvironmentEC2` | `cloud9:CreateEnvironmentEC2`<br><br>Diperlukan untuk menciptakan lingkungan AWS Cloud9 EC2 pengembangan. | `*` |
| `CreateEnvironmentMembership` | `cloud9:CreateEnvironmentMembership`<br><br>Diperlukan untuk menambahkan anggota ke lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `DeleteEnvironment` | `cloud9:DeleteEnvironment`<br><br>Diperlukan untuk menghapus lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `DeleteEnvironmentMembership` | `cloud9:DeleteEnvironmentMembership`<br><br>Diperlukan untuk menghapus anggota dari lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `DescribeEnvironmentMemberships` | `cloud9:DescribeEnvironmentMemberships` | `*` |

| AWS Cloud9 operasi | Izin yang diperlukan (tindakan API) | Sumber Daya |
| --- | --- | --- |
| | Diperlukan untuk mendapatkan daftar anggota di lingkungan. | |
| `DescribeEnvironments` | `cloud9:DescribeEnvironments`<br><br>Diperlukan untuk mendapatkan informasi tentang lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `DescribeEnvironmentStatus` | `cloud9:DescribeEnvironmentStatus`<br><br>Diperlukan untuk mendapatkan informasi tentang status lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `UpdateEnvironment` | `cloud9:UpdateEnvironment`<br><br>Diperlukan untuk memperbarui pengaturan untuk lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |
| `UpdateEnvironmentMembership` | `cloud9:UpdateEnvironmentMembership`<br><br>Diperlukan untuk memperbarui pengaturan untuk anggota di lingkungan. | `arn:aws:cloud9:` *REGION_ID* `:` *ACCOUNT_ID* `:environment:` *ENVIRONMENT_ID* |

Operasi API hanya izin

| AWS Cloud9 operasi | Deskripsi | Dokumentasi konsol |
| --- | --- | --- |
| `ActivateEC2Remote` | `cloud9:ActivateEC2Remote`<br><br>Memulai EC2 instans Amazon yang terhubung dengan AWS Cloud9 IDE Anda. | Membuka lingkungan di AWS Cloud9 |
| `CreateEnvironmentSSH` | `cloud9:CreateEnvironmentSSH`<br><br>Menciptakan lingkungan pengembangan AWS Cloud9 SSH. | Membuat Lingkungan SSH |
| `CreateEnvironmentToken` | `cloud9:CreateEnvironmentToken`<br><br>Membuat token otentikasi yang memungkinkan koneksi antara AWS Cloud9 IDE dan lingkungan pengguna. | Menciptakan EC2 Lingkungan |
| `DescribeEC2Remote` | `cloud9:DescribeEC2Remote`<br><br>Mendapat detail tentang koneksi ke lingkungan EC2 pengembangan, termasuk host, pengguna, dan port. | Menciptakan EC2 Lingkungan |
| `DescribeSSHRemote` | `cloud9:DescribeSSHRemote`<br><br>Mendapat detail tentang koneksi ke lingkungan pengembangan SSH, | Membuat Lingkungan SSH |

| AWS Cloud9 operasi | Deskripsi | Dokumentasi konsol |
| --- | --- | --- |
| | termasuk host, pengguna, dan port. | |
| `GetEnvironmentConfig` | `cloud9:GetEnvironmentConfig`<br><br>Mendapat informasi konfigurasi yang digunakan untuk menginisialisasi AWS Cloud9 IDE. | Bekerja dengan AWS Cloud9 IDE |
| `GetEnvironmentSettings` | `cloud9:GetEnvironmentSettings`<br><br>Mendapat pengaturan AWS Cloud9 IDE untuk lingkungan pengembangan tertentu. | Bekerja dengan AWS Cloud9 IDE |
| `GetMembershipSettings` | `cloud9:GetMembershipSettings`<br><br>Mendapat pengaturan AWS Cloud9 IDE untuk anggota lingkungan tertentu. | Bekerja dengan lingkungan bersama di AWS Cloud9 |
| `GetUserPublicKey` | `cloud9:GetUserPublicKey`<br><br>Mendapat kunci SSH publik pengguna, yang digunakan oleh AWS Cloud9 untuk terhubung ke lingkungan pengembangan SSH. | Membuat Lingkungan SSH |

| AWS Cloud9 operasi | Deskripsi | Dokumentasi konsol |
| --- | --- | --- |
| `GetUserSettings` | `cloud9:GetUserSettings`<br><br>Mendapat pengaturan AWS Cloud9 IDE untuk pengguna tertentu. | Bekerja dengan AWS Cloud9 IDE |
| `ModifyTemporaryCredentialsOnEnvironmentEC2` | `cloud9:ModifyTemporaryCredentialsOnEnvironmentEC2`<br><br>Menetapkan kredensil sementara AWS terkelola pada EC2 instans Amazon yang digunakan oleh lingkungan pengembangan AWS Cloud9 terintegrasi (IDE). | AWS kredensi sementara yang dikelola |
| `UpdateEnvironmentSettings` | `cloud9:UpdateEnvironmentSettings`<br><br>Memperbarui pengaturan AWS Cloud9 IDE untuk lingkungan pengembangan tertentu. | Bekerja dengan AWS Cloud9 IDE |
| `UpdateMembershipSettings` | `cloud9:UpdateMembershipSettings`<br><br>Memperbarui pengaturan AWS Cloud9 IDE untuk anggota lingkungan tertentu. | Bekerja dengan lingkungan bersama di AWS Cloud9 |

| AWS Cloud9 operasi | Deskripsi | Dokumentasi konsol |
| --- | --- | --- |
| `UpdateSSHRemote` | `cloud9:UpdateSSHRemote`<br><br>Memperbarui detail tentang koneksi ke lingkungan pengembangan SSH, termasuk host, pengguna, dan port. | Membuat Lingkungan SSH |
| `UpdateUserSettings` | `cloud9:UpdateUserSettings`<br><br>Memperbarui pengaturan AWS Cloud9 IDE untuk pengguna tertentu. | Bekerja dengan AWS Cloud9 IDE |
| `GetMigrationExperiences` | `cloud9:GetMigrationExperiences`<br><br>Memberikan izin kepada AWS Cloud9 pengguna untuk mendapatkan pengalaman migrasi dari AWS Cloud9 ke CodeCatalyst. | |

## AWS kredensi sementara yang dikelola

Jika Anda hanya mencari daftar tindakan yang mendukung kredenal sementara yang AWS dikelola, lanjutkan ke. Tindakan yang didukung oleh kredensial sementara yang AWS dikelola

Untuk lingkungan AWS Cloud9 EC2 pengembangan, AWS Cloud9 buat kredenal AWS akses sementara tersedia untuk Anda di lingkungan. Kami menyebutnya kredensial sementara terkelola AWS . Hal ini memberikan manfaat berikut:

- Anda tidak perlu menyimpan kredensi AWS akses permanen AWS entitas (misalnya, pengguna IAM) di mana pun di lingkungan. Hal ini mencegah kredensial tersebut diakses oleh anggota lingkungan tanpa sepengetahuan dan persetujuan Anda.
- Anda tidak perlu mengatur, mengelola, atau melampirkan profil instance secara manual ke EC2 instans Amazon yang terhubung ke lingkungan. Profil instance adalah pendekatan lain untuk mengelola kredensil AWS akses sementara.
- AWS Cloud9 terus memperbarui kredensialnya sementara, sehingga satu set kredensil hanya dapat digunakan untuk waktu yang terbatas. Ini adalah praktik terbaik AWS keamanan. Untuk informasi selengkapnya, lihat Membuat dan memperbarui kredensi sementara AWS terkelola.
- AWS Cloud9 Memberikan batasan tambahan tentang bagaimana kredensialnya sementara dapat digunakan untuk mengakses AWS tindakan dan sumber daya dari lingkungan. Ini juga merupakan praktik terbaik AWS keamanan.

**Important**

Saat ini, jika EC2 instance lingkungan Anda diluncurkan ke subnet pribadi, Anda tidak dapat menggunakan kredensil sementara AWS terkelola untuk memungkinkan EC2 lingkungan mengakses AWS layanan atas nama AWS entitas (misalnya, pengguna IAM).
Untuk informasi selengkapnya tentang kapan Anda dapat meluncurkan EC2 instance ke subnet pribadi, lihatBuat subnet untuk AWS Cloud9.

**Note**

Pertimbangkan untuk menggunakan kebijakan AWS terkelola, bukan kebijakan sebaris saat Anda menggunakan kredensil sementara AWS terkelola.

Begini cara kerja kredensi sementara AWS terkelola setiap kali EC2 lingkungan mencoba mengakses Layanan AWS atas nama AWS entitas (misalnya, pengguna IAM):

1. AWS Cloud9 memeriksa untuk melihat apakah AWS entitas pemanggil (misalnya, pengguna IAM) memiliki izin untuk mengambil tindakan yang diminta untuk sumber daya yang diminta di. AWS Jika izin tidak ada atau ditolak secara eksplisit, permintaan gagal.
2. AWS Cloud9 memeriksa kredensil sementara AWS terkelola untuk melihat apakah izinnya mengizinkan tindakan yang diminta untuk sumber daya yang diminta masuk. AWS Jika izin

tidak ada atau ditolak secara eksplisit, permintaan gagal. Untuk daftar izin yang AWS mengelola dukungan kredensial sementara, lihat. Tindakan yang didukung oleh kredensial sementara yang AWS dikelola

- Jika AWS entitas dan kredenal sementara AWS terkelola mengizinkan tindakan yang diminta untuk sumber daya yang diminta, permintaan akan berhasil.
- Jika AWS entitas atau kredensi sementara yang AWS dikelola secara eksplisit menolak atau gagal secara eksplisit mengizinkan tindakan yang diminta untuk sumber daya yang diminta, permintaan gagal. Ini berarti bahwa meskipun AWS entitas pemanggil memiliki izin yang benar, permintaan akan gagal jika AWS Cloud9 tidak juga secara eksplisit mengizinkannya. Demikian juga, jika AWS Cloud9 memungkinkan tindakan tertentu diambil untuk sumber daya tertentu, permintaan gagal jika AWS entitas juga tidak secara eksplisit mengizinkannya.

Pemilik EC2 lingkungan dapat mengaktifkan atau menonaktifkan kredensil sementara yang AWS dikelola untuk lingkungan tersebut kapan saja, sebagai berikut:

1. Dengan lingkungan terbuka, di AWS Cloud9 IDE, pada bilah menu pilih AWS Cloud9, Preferensi.
2. Pada tab Preferensi, di panel navigasi, pilih Pengaturan AWS , Kredensial.
3. Gunakan Kredensial sementara terkelola AWS untuk mengaktifkan atau menonaktifkan kredensial sementara terkelola AWS .

> **Note**
>
> Anda juga dapat mengaktifkan atau menonaktifkan kredenal sementara AWS terkelola dengan memanggil operasi AWS Cloud9 API `UpdateEnvironment`dan menetapkan nilai ke parameter. `managedCredentialsAction` Anda dapat meminta operasi API ini menggunakan AWS alat standar seperti AWS SDKs dan AWS CLI.

Jika Anda menonaktifkan kredensi sementara AWS terkelola, lingkungan tidak dapat mengakses manapun Layanan AWS, terlepas dari AWS entitas yang membuat permintaan. Tetapi, misalkan Anda tidak dapat atau tidak ingin mengaktifkan kredensil sementara AWS terkelola untuk suatu lingkungan, dan Anda masih memerlukan lingkungan untuk mengakses. Layanan AWS Kemudian, pertimbangkan alternatif berikut:

- Lampirkan profil instance ke EC2 instans Amazon yang terhubung ke lingkungan. Untuk petunjuk, lihat Membuat dan Menggunakan Profil Instans untuk mengelola Kredensial Sementara.
- Simpan kredenal AWS akses permanen Anda di lingkungan, misalnya, dengan mengatur variabel lingkungan khusus atau dengan menjalankan perintah. `aws configure` Untuk petunjuk, lihat Membuat dan menyimpan kredensial akses permanen di Lingkungan.

Alternatif sebelumnya mengesampingkan semua izin yang diizinkan (atau ditolak) oleh kredensil sementara yang AWS dikelola di lingkungan. EC2

## Tindakan yang didukung oleh kredensial sementara yang AWS dikelola

Untuk lingkungan AWS Cloud9 EC2 pengembangan, kredensi sementara AWS terkelola memungkinkan semua AWS tindakan untuk semua AWS sumber daya di pemanggil Akun AWS, dengan batasan berikut:

- Untuk AWS Cloud9, hanya tindakan berikut yang diizinkan:
  - `cloud9:CreateEnvironmentEC2`
  - `cloud9:CreateEnvironmentSSH`
  - `cloud9:DescribeEnvironmentMemberships`
  - `cloud9:DescribeEnvironments`
  - `cloud9:DescribeEnvironmentStatus`
  - `cloud9:UpdateEnvironment`
- Untuk IAM, hanya tindakan berikut yang diizinkan:
  - `iam:AttachRolePolicy`
  - `iam:ChangePassword`
  - `iam:CreatePolicy`
  - `iam:CreatePolicyVersion`
  - `iam:CreateRole`
  - `iam:CreateServiceLinkedRole`
  - `iam:DeletePolicy`
  - `iam:DeletePolicyVersion`
  - `iam:DeleteRole`
  - `iam:DeleteRolePolicy`

  - `iam:DeleteSSHPublicKey`
  - `iam:DetachRolePolicy`
  - `iam:GetInstanceProfile`
  - `iam:GetPolicy`
  - `iam:GetPolicyVersion`
  - `iam:GetRole`
  - `iam:GetRolePolicy`
  - `iam:GetSSHPublicKey`
  - `iam:GetUser`
  - `iam:List*`
  - `iam:PassRole`
  - `iam:PutRolePolicy`
  - `iam:SetDefaultPolicyVersion`
  - `iam:UpdateAssumeRolePolicy`
  - `iam:UpdateRoleDescription`
  - `iam:UpdateSSHPublicKey`
  - `iam:UploadSSHPublicKey`
- Semua tindakan IAM yang berinteraksi dengan peran hanya diizinkan untuk nama peran yang dimulai dengan `Cloud9-`. Namun, `iam:PassRole` bekerja dengan semua nama peran.
- Untuk AWS Security Token Service (AWS STS), hanya tindakan berikut yang diizinkan:
  - `sts:GetCallerIdentity`
  - `sts:DecodeAuthorizationMessage`
- Semua AWS tindakan yang didukung dibatasi pada alamat IP lingkungan. Ini adalah praktik terbaik AWS keamanan.

Jika AWS Cloud9 tidak mendukung tindakan atau sumber daya yang perlu diakses oleh EC2 lingkungan, atau jika kredenal sementara AWS terkelola dimatikan untuk suatu EC2 lingkungan dan Anda tidak dapat menyalakannya kembali, pertimbangkan alternatif berikut:

- Lampirkan profil instance ke EC2 instans Amazon yang terhubung ke EC2 lingkungan. Untuk petunjuk, lihat Membuat dan menggunakan profil instans untuk mengelola kredensial sementara.

- Simpan kredenal AWS akses permanen Anda di EC2 lingkungan, misalnya, dengan mengatur variabel lingkungan khusus atau dengan menjalankan perintah. `aws configure` Untuk petunjuk, lihat Membuat dan menyimpan kredensial akses permanen di Lingkungan.

Alternatif sebelumnya mengesampingkan semua izin yang diizinkan (atau ditolak) oleh kredensil sementara yang AWS dikelola di lingkungan. EC2

Membuat dan memperbarui kredensi sementara AWS terkelola

Untuk lingkungan AWS Cloud9 EC2 pengembangan, kredensi sementara yang AWS dikelola dibuat saat pertama kali Anda membuka lingkungan.

AWS kredensi sementara terkelola diperbarui di bawah salah satu kondisi berikut:

- Setiap kali periode waktu tertentu berlalu. Saat ini, ini setiap lima menit.
- Setiap kali Anda memuat ulang tab peramban web yang menampilkan IDE untuk lingkungan.
- Ketika stempel waktu yang tercantum dalam file `~/.aws/credentials` untuk lingkungan tercapai.
- Jika pengaturan kredensial sementara terkelola AWS diatur ke nonaktif, kapan pun Anda mengaktifkannya kembali. (Untuk melihat atau mengubah pengaturan ini, pilih AWS Cloud9, Preferensi di bilah menu IDE. Pada tab Preferensi, di panel navigasi, pilih Pengaturan AWS , Kredensial.)
- Untuk keamanan, kredensi sementara yang AWS dikelola akan kedaluwarsa secara otomatis setelah 15 menit. Agar kredensial di-refresh, pemilik lingkungan harus terhubung ke lingkungan AWS Cloud9 melalui IDE. Untuk informasi selengkapnya tentang peran pemilik lingkungan, lihat Mengontrol akses ke kredensial sementara terkelola AWS.

Mengontrol akses ke kredensial sementara terkelola AWS

Kolaborator dengan kredensial sementara AWS terkelola dapat digunakan AWS Cloud9 untuk berinteraksi dengan orang lain. Layanan AWS Untuk memastikan bahwa hanya kolaborator tepercaya yang diberikan kredensial sementara terkelola AWS , kredensial ini dinonaktifkan jika anggota baru ditambahkan oleh siapa pun selain pemilik lingkungan. Kredensial dinonaktifkan oleh penghapusan file `~/.aws/credentials`.

> **Important**
>
> AWS kredensi sementara yang dikelola juga kedaluwarsa secara otomatis setiap 15 menit. Agar kredensi disegarkan sehingga kolaborator dapat terus menggunakannya, pemilik lingkungan harus terhubung ke AWS Cloud9 lingkungan melalui IDE.

Hanya pemilik lingkungan yang dapat mengaktifkan kembali kredensil sementara yang AWS dikelola sehingga dapat dibagikan dengan anggota lain. Saat pemilik lingkungan membuka IDE, kotak dialog mengonfirmasi bahwa kredensil sementara AWS terkelola dinonaktifkan. Pemilik lingkungan dapat mengaktifkan kembali kredensial untuk semua anggota atau tetap membuatnya dinonaktifkan untuk semua anggota.

> **Warning**
>
> Untuk mematuhi praktik keamanan terbaik, tetap nonaktifkan kredensial sementara terkelola jika Anda tidak yakin tentang identitas pengguna terakhir yang ditambahkan ke lingkungan. Anda dapat memeriksa daftar anggota dengan read/write izin di jendela Kolaborasi.

# Penebangan dan pemantauan di AWS Cloud9

## Memantau aktivitas dengan CloudTrail

AWS Cloud9 terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Cloud9. CloudTrail menangkap semua panggilan API untuk AWS Cloud9 sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari AWS Cloud9 konsol dan dari panggilan kode ke AWS Cloud9 APIs.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon Simple Storage Service (Amazon S3), termasuk acara untuk. AWS Cloud9

Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat AWS Cloud9, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Cloud9 API dengan AWS CloudTrail.

## Memantau kinerja EC2 lingkungan

Jika Anda menggunakan lingkungan AWS Cloud9 EC2 pengembangan, Anda dapat memantau keandalan, ketersediaan, dan kinerja EC2 instans Amazon terkait. Dengan pemantauan status instans, misalnya, Anda dapat dengan cepat menentukan apakah Amazon EC2 telah mendeteksi masalah yang mungkin mencegah instans Anda menjalankan aplikasi.

Untuk informasi selengkapnya, lihat Memantau Amazon EC2 di Panduan EC2 Pengguna Amazon.

# Validasi kepatuhan untuk AWS Cloud9

Auditor pihak ketiga menilai keamanan dan kepatuhan AWS layanan sebagai bagian dari beberapa program AWS kepatuhan.

AWS Cloud9 berada dalam lingkup dengan program kepatuhan berikut:

### SOC

AWS Laporan Sistem dan Kontrol Organisasi (SOC) adalah laporan pemeriksaan pihak ketiga independen yang menunjukkan bagaimana AWS mencapai kontrol dan tujuan kepatuhan utama.

| Layanan | SDK | SOC 1,2,3 |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

### PCI

Standar Keamanan Data Industri Kartu Pembayaran (PCI DSS) adalah standar keamanan informasi eksklusif yang dikelola oleh Dewan Standar Keamanan PCI, yang didirikan oleh American Express, Discover Financial Services, JCB International, Worldwide dan Visa Inc. MasterCard

| Layanan | SDK | PCI |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

## FedRAMP

Federal Risk and Authorization Management Program (FedRAMP) adalah program pemerintah AS yang memberikan pendekatan standar untuk penilaian keamanan, otorisasi, dan pemantauan berkelanjutan untuk produk dan layanan cloud.

Layanan melalui penilaian dan otorisasi FedRAMP akan memiliki status sebagai berikut:

- Penilaian Organisasi Penilaian Pihak Ketiga (3PAO): Layanan ini saat ini sedang menjalani penilaian oleh penilai pihak ketiga kami.
- Tinjauan Joint Authorization Board (JAB): Layanan ini saat ini sedang menjalani tinjauan JAB.

| Layanan | SDK | FedRAMP Moderat (Timur/Barat) | FedRAMP Tinggi () GovCloud |
|---|---|---|---|
| AWS Cloud9 | awan9 | JAB Ulasan | N/A |

## DoD CC SRG

The Department of Defense (DoD) Cloud Computing Security Requirements Guide (SRG) menyediakan penilaian standar dan proses otorisasi untuk penyedia layanan cloud () CSPs untuk mendapatkan otorisasi sementara DoD, sehingga mereka dapat melayani pelanggan DoD.

Layanan yang melalui penilaian dan otorisasi DoD CC SRG akan memiliki status sebagai berikut:

- Penilaian Organisasi Penilaian Pihak Ketiga (3PAO): Layanan ini saat ini sedang menjalani penilaian oleh penilai pihak ketiga kami.
- Tinjauan Joint Authorization Board (JAB): Layanan ini saat ini sedang menjalani tinjauan JAB.
- Tinjauan Badan Sistem Informasi Pertahanan (DISA): Layanan ini saat ini sedang menjalani tinjauan DISA.

| Layanan | SDKs | DoD CC SRG IL2 (Timur/Barat) | DoD CC IL2 SRG () GovCloud | DoD CC IL4 SRG () GovCloud | DoD CC IL5 SRG () GovCloud | DoD CC SRG IL6 (Wilayah Rahasia)AWS |
|---|---|---|---|---|---|---|
| AWS Cloud9 | awan9 | JAB Ulasan | N/A | N/A | N/A | N/A |

## HIPAA BAA

Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan 1996 (HIPAA) adalah undang-undang federal yang mewajibkan pembuatan standar nasional untuk melindungi informasi kesehatan pasien yang sensitif agar tidak diungkapkan tanpa persetujuan atau sepengetahuan pasien.

AWS memungkinkan entitas yang dilindungi dan rekan bisnis mereka yang tunduk pada HIPAA untuk memproses, menyimpan, dan mengirimkan informasi kesehatan yang dilindungi (PHI) dengan aman. Selain itu, pada Juli 2013, AWS menawarkan Adendum Asosiasi Bisnis standar (BAA) untuk pelanggan tersebut

| Layanan | SDK | HIPAA BAA |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

## IRAP

Program Penilai Terdaftar Keamanan Informasi (IRAP) memungkinkan pelanggan Pemerintah Australia untuk memvalidasi bahwa kontrol yang sesuai telah ada dan menentukan model tanggung jawab yang sesuai untuk memenuhi persyaratan Manual Keamanan Informasi Pemerintah Australia (ISM) yang diproduksi oleh Australian Cyber Security Centre (ACSC).

| Layanan | Ruang nama* | IRAP dilindungi |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

* Ruang nama membantu Anda mengidentifikasi layanan di seluruh lingkungan Anda. AWS Misalnya, saat Anda membuat kebijakan IAM, bekerja dengan Amazon Resource Names (ARNs), dan membaca AWS CloudTrail log.

## C5

Cloud Computing Compliance Controls Catalog (C5) adalah skema pengesahan yang didukung Pemerintah Jerman yang diperkenalkan di Jerman oleh Kantor Federal untuk Keamanan Informasi (BSI) untuk membantu organisasi menunjukkan keamanan operasional terhadap serangan cyber umum saat menggunakan layanan cloud dalam konteks “Rekomendasi Keamanan untuk Penyedia Cloud” Pemerintah Jerman.

| Layanan | SDK | C5 |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

## FINMA

FINMA adalah regulator pasar keuangan independen Swiss. Amazon Web Services (AWS) telah menyelesaikan Laporan FINMA ISAE 3000 Tipe 2.

| Layanan | SDK | FINMA |
|---|---|---|
| AWS Cloud9 | awan9 | ✓ |

## GSMA

Asosiasi GSM adalah organisasi industri yang mewakili kepentingan operator jaringan seluler di seluruh dunia. Amazon Web Services (AWS) Eropa (Paris) dan Wilayah Timur AS (Ohio) sekarang disertifikasi oleh Asosiasi GSM (GSMA) di bawah Security Accreditation Scheme Subscription Management (SAS-SM) dengan lingkup Data Center Operations and Management (DCOM). Keselarasan dengan persyaratan GSMA ini menunjukkan komitmen berkelanjutan kami untuk mematuhi harapan yang meningkat bagi penyedia layanan cloud.

| Layanan | AS-Timur (Ohio) | Eropa (Paris) |
|---|---|---|
| AWS Cloud9 | ✓ | ✓ |

## PiTuKri

AWS Keselarasan dengan PiTuKri persyaratan menunjukkan komitmen berkelanjutan kami untuk memenuhi harapan yang meningkat untuk penyedia layanan cloud yang ditetapkan oleh Badan Transportasi dan Komunikasi Finlandia, Traficom.

| Layanan | SDK | PiTuKri |
| --- | --- | --- |
| AWS Cloud9 | awan9 | ✓ |

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program AWS Kepatuhan Program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- Kepatuhan dan Tata Kelola Keamanan – Panduan implementasi solusi ini membahas pertimbangan arsitektur serta memberikan langkah-langkah untuk menerapkan fitur keamanan dan kepatuhan.
- Referensi Layanan yang Memenuhi Syarat HIPAA — Daftar layanan yang memenuhi syarat HIPAA. Tidak semua memenuhi Layanan AWS syarat HIPAA.
- AWS Sumber Daya AWS — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- AWS Panduan Kepatuhan Pelanggan - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO)).
- Mengevaluasi Sumber Daya dengan Aturan dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.

- AWS Security Hub— Ini Layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS. Security Hub menggunakan kontrol keamanan untuk sumber daya AWS Anda serta untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat Referensi kontrol Security Hub.
- Amazon GuardDuty — Ini Layanan AWS mendeteksi potensi ancaman terhadap beban kerja Akun AWS, kontainer, dan data Anda dengan memantau lingkungan Anda untuk aktivitas yang mencurigakan dan berbahaya. GuardDuty dapat membantu Anda mengatasi berbagai persyaratan kepatuhan, seperti PCI DSS, dengan memenuhi persyaratan deteksi intrusi yang diamanatkan oleh kerangka kerja kepatuhan tertentu.
- AWS Audit ManagerIni Layanan AWS membantu Anda terus mengaudit AWS penggunaan Anda untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

## Ketahanan di AWS Cloud9

Infrastruktur AWS global dibangun di sekitar AWS Wilayah dan Zona Ketersediaan. AWS Wilayah menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat Infrastruktur AWS Global.

Selain infrastruktur AWS global, AWS Cloud9 mendukung fitur khusus untuk mendukung ketahanan data dan kebutuhan cadangan Anda.

- Integrasikan AWS Cloud9 dengan AWS CodeCommit, layanan kontrol versi yang dihosting oleh Amazon Web Services yang dapat Anda gunakan untuk menyimpan dan mengelola aset secara pribadi (seperti dokumen, kode sumber, dan file biner) di cloud. Untuk informasi selengkapnya, lihat Mengintegrasikan AWS Cloud9 dengan AWS CodeCommit di Panduan AWS CodeCommit Pengguna.

- Gunakan sistem kontrol versi Git pada lingkungan AWS Cloud9 pengembangan untuk mencadangkan file dan data pada GitHub repositori jarak jauh. Untuk informasi selengkapnya, lihat Kontrol sumber visual dengan panel Git.

## Keamanan infrastruktur di AWS Cloud9

Sebagai layanan terkelola, AWS Cloud9 dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat Keamanan AWS Cloud. Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses AWS Cloud9 melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan prinsipal IAM. Atau Anda dapat menggunakan AWS Security Token Service (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

> **Note**
>
> Secara default, lingkungan AWS Cloud9 EC2 pengembangan secara otomatis menginstal patch keamanan untuk paket sistem instance.

## Pembaruan perangkat lunak dan patch

AWS Cloud9 lingkungan pengembangan berjalan di atas sumber daya komputasi awan. Sumber daya komputasi awan dapat berupa EC2 instans Amazon untuk EC2 lingkungan, atau sumber daya komputasi cloud Anda sendiri untuk lingkungan SSH. Untuk informasi lebih lanjut tentang kedua opsi ini, lihat Lingkungan dan sumber daya komputasi bagian.

AWS Cloud9 EC2 lingkungan secara otomatis menginstal patch keamanan sistem operasi dan pembaruan setelah lingkungan diluncurkan. AWS Cloud9 lingkungan juga berisi paket perangkat lunak yang diperlukan AWS Cloud9 untuk berfungsi dan mendukung fitur IDE. Paket-paket ini ditambal secara otomatis saat lingkungan dimuat.

AWS Cloud9 EC2 lingkungan juga mencakup beberapa alat AWS pengembangan pra-instal. Pembaruan untuk alat ini harus diinstal secara manual oleh AWS Cloud9 pengguna atau administrator mereka. Untuk informasi selengkapnya tentang memperbarui alat ini, lihat bagian berikut:

- Instal atau perbarui versi terbaru AWS CLI di Panduan AWS Command Line Interface Pengguna.
- Mengelola versi AWS SAM CLI di Panduan AWS Serverless Application Model Pengembang.
- Instal AWS CDK di Panduan AWS Cloud Development Kit (AWS CDK) Pengembang.

## Praktik terbaik keamanan untuk AWS Cloud9

Praktik terbaik berikut adalah pedoman umum dan tidak mewakili solusi keamanan yang lengkap. Karena praktik terbaik ini mungkin tidak sesuai atau tidak memadai untuk lingkungan Anda, perlakukan itu sebagai pertimbangan yang bermanfaat, bukan preskripsi.

Beberapa praktik terbaik keamanan untuk AWS Cloud9

- Simpan kode Anda dengan aman di sistem kontrol versi, misalnya, AWS CodeCommit.
- Untuk lingkungan AWS Cloud9 EC2 pengembangan Anda, konfigurasikan dan gunakan volume terenkripsi Amazon Elastic Block Store.
- Untuk EC2 lingkungan Anda, gunakan tag untuk mengontrol akses ke AWS Cloud9 sumber daya Anda.
- Untuk lingkungan AWS Cloud9 pengembangan bersama Anda, ikuti praktik terbaik untuk mereka.

# Pemecahan masalah AWS Cloud9

Gunakan informasi berikut untuk mengidentifikasi dan mengatasi masalah AWS Cloud9.

Jika masalah Anda tidak terdaftar atau Anda memerlukan bantuan tambahan, lihat Forum AWS Cloud9 Diskusi. Anda mungkin diminta untuk masuk saat memasuki forum ini. Anda juga dapat kontak kami secara langsung.

Topik


- Penginstal
- AWS Cloud9 Lingkungan
- Amazon EC2
- AWS Layanan lainnya
- Pratinjau aplikasi
- Kinerja
- Aplikasi dan layanan pihak ketiga


## Penginstal

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan penginstal. AWS Cloud9

### AWS Cloud9 Pemasang hang atau gagal

Masalah: Saat Anda mengunduh dan menjalankan AWS Cloud9 Penginstal, satu atau lebih kesalahan terjadi, dan skrip penginstal tidak ditampilkan. `Done`

Penyebab: AWS Cloud9 Pemasang mengalami satu atau lebih kesalahan yang tidak dapat dipulihkan dan gagal sebagai hasilnya.

Solusi: Untuk informasi lebih lanjut, lihatMemecahkan Masalah Installer AWS Cloud9. Lihat masalah umum, kemungkinan penyebab, dan solusi yang direkomendasikan yang diberikan.

## AWS Cloud9 installer tidak selesai setelah menampilkan: “Package Cloud9 IDE 1"

Masalah: AWS Cloud9 diinstal pada EC2 instans Amazon yang ada atau di server Anda sendiri sebagai bagian dari proses pembuatan lingkungan pengembangan SSH. Instalasi terhenti setelah Anda melihat olahpesan ini di kotak dialog Penginstal AWS Cloud9 : "Paket Cloud9 IDE 1". Jika Anda memilih Batalkan, Anda melihat olahpesan berikut: "Instalasi Gagal." Kesalahan ini terjadi ketika AWS Cloud9 paket tidak dapat diinstal pada host SSH pelanggan.

Penyebab: Host SSH mengharuskan Anda menginstal Node.js. Kami merekomendasikan menginstal yang terbaru Node.js versi yang didukung oleh sistem operasi host. Jika Anda memiliki versi Node.js pada host Anda yang AWS Cloud9 tidak mendukung, kesalahan penginstalan mungkin terjadi.

Solusi yang disarankan: Instal versi Node.js yang AWS Cloud9 mendukung host SSH Anda.

## Gagal menginstal dependensi

Masalah: AWS Cloud9 membutuhkan akses internet untuk mengunduh dependensi.

Kemungkinan penyebabnya:

- Jika AWS Cloud9 lingkungan Anda menggunakan proxy untuk mengakses internet, AWS Cloud9 perlu detail proxy untuk menginstal dependensi. Jika Anda tidak memberikan detail proxy Anda AWS Cloud9, kesalahan ini akan muncul.
- Penyebab lain dari ini bisa jadi jika lingkungan Anda tidak mengizinkan lalu lintas keluar.

Solusi yang direkomendasikan:

- Untuk memberikan detail proxy Anda AWS Cloud9, tambahkan kode berikut ke `~/.bashrc` file lingkungan Anda:

  ```
  export http_proxy=[proxy url for http]
  export https_proxy=[proxy url for https]
  #Certificate Authority used by your proxy
  export NODE_EXTRA_CA_CERTS=[path_to_pem_certificate]
  ```

  Misalnya, jika URL proxy HTTP Anda `https://172.31.26.80:3128` dan URL proxy HTTP Anda`https://172.31.26.80:3129`, tambahkan baris berikut ke `~/.bashrc` file Anda dan atur

NODE_EXTRA_CA_CERTS ke jalur file otoritas sertifikat dalam format PEM. Untuk informasi lebih lanjut tentang variabel ini, lihathttps://nodejs.org/api/cli.html#node_extra_ca_certsfile.

```
export http_proxy=http://172.31.26.80:3128
export https_proxy=https://172.31.26.80:3129
export NODE_EXTRA_CA_CERTS=[path_to_pem_certificate]
```

- Jika Anda menggunakan EC2 instans Amazon tanpa masuk, Anda harus memastikan titik akhir VPC Amazon untuk Amazon S3 dikonfigurasi. Untuk informasi selengkapnya tentang ini, lihat Mengonfigurasi titik akhir Amazon VPC untuk dependensi unduhan Amazon S3.

## Kesalahan lingkungan SSH: “Python versi 3 diperlukan untuk menginstal pty.js”

Masalah: Setelah Anda membuka lingkungan pengembangan AWS Cloud9 SSH, terminal di AWS Cloud9 IDE menampilkan pesan yang dimulai dengan “Python versi 3 diperlukan untuk menginstal pty.js.”

Penyebab: Agar berfungsi seperti yang diharapkan, lingkungan SSH mengharuskan Python versi 3 diinstal.

Solusi: Instal Python versi 3 di lingkungan. Untuk memeriksa versi Anda, dari terminal server Anda, jalankan perintah **`python --version`** . Untuk menginstal Python 3 di server Anda, lihat salah satu dari berikut ini:

- Langkah 1: Pasang Python dalam Contoh Python.
- Unduh Python di situs web Python.

# AWS Cloud9 Lingkungan

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan Lingkungan. AWS Cloud9

## Kesalahan pembuatan lingkungan: “Kami tidak dapat membuat EC2 instance...”

Masalah: Saat Anda mencoba membuat lingkungan AWS Cloud9 pengembangan, sebuah pesan muncul dengan frasa “Kami tidak dapat membuat EC2 instance di akun Anda selama verifikasi dan aktivasi akun.”

Penyebab: saat AWS ini memverifikasi dan mengaktifkan Anda Akun AWS. Sebelum aktivasi selesai, yang dapat memakan waktu hingga 24 jam, Anda tidak dapat membuat lingkungan ini atau lainnya.

Solusi: Coba buat lingkungan lagi nanti. Jika Anda masih menerima pesan ini setelah 24 jam, hubungi dukungan. Selain itu, penting untuk diketahui bahwa, bahkan ketika upaya untuk membuat lingkungan gagal, AWS CloudFormation membuat tumpukan terkait di akun Anda. Tumpukan ini dihitung terhadap kuota pembuatan tumpukan untuk akun Anda. Untuk menghindari kehabisan kuota pembuatan tumpukan, Anda dapat menghapus tumpukan yang gagal ini. Untuk informasi selengkapnya, lihat Menghapus Tumpukan pada Konsol AWS CloudFormation di Panduan Pengguna AWS CloudFormation .

## Kesalahan pembuatan lingkungan: “Tidak diizinkan untuk melakukan sts:AssumeRole”

Masalah: Saat Anda mencoba membuat lingkungan baru, Anda melihat kesalahan ini: “Tidak diizinkan untuk melakukan sts:AssumeRole,” dan lingkungan tidak dibuat.

Kemungkinan penyebab: Peran AWS Cloud9 terkait layanan tidak ada di Anda. Akun AWS

Solusi yang disarankan: Buat peran AWS Cloud9 terkait layanan di Anda. Akun AWS Anda dapat melakukannya dengan menjalankan perintah berikut di AWS Command Line Interface (AWS CLI) atau AWS CloudShell.

```
aws iam create-service-linked-role --aws-service-name cloud9.amazonaws.com # For the
 AWS CLI.
iam create-service-linked-role --aws-service-name cloud9.amazonaws.com     # For the
 aws-shell.
```

Jika Anda tidak dapat melakukan ini, tanyakan kepada Akun AWS administrator Anda.

Setelah Anda menjalankan perintah ini, cobalah membuat lingkungan lagi.

## Identitas federasi tidak dapat menciptakan lingkungan

Masalah: Saat Anda mencoba menggunakan identitas AWS federasi untuk membuat lingkungan AWS Cloud9 pengembangan, pesan kesalahan akses akan ditampilkan, dan lingkungan tidak dibuat.

Penyebab:: AWS Cloud9 menggunakan peran terkait layanan. Peran terkait layanan dibuat pertama kali lingkungan dibuat di akun menggunakan panggilan. `iam:CreateServiceLinkedRole` Namun, pengguna federasi tidak dapat memanggil APIs IAM. Untuk informasi selengkapnya, lihat GetFederationToken di dalam Referensi API AWS Security Token Service .

Solusi: Minta Akun AWS administrator untuk membuat peran terkait layanan AWS Cloud9 baik di konsol IAM atau dengan menjalankan perintah ini dengan (): AWS Command Line Interface AWS CLI

```
aws iam create-service-linked-role --aws-service-name cloud9.amazonaws.com
```

Atau perintah ini dengan AWS-shell:

```
iam create-service-linked-role --aws-service-name cloud9.amazonaws.com
```

Untuk informasi selengkapnya, lihat Menggunakan Peran Tertaut Layanan di Panduan Pengguna IAM.

## Kesalahan konsol: "Pengguna tidak berwenang untuk melakukan tindakan pada sumber daya"

Masalah: Saat Anda mencoba menggunakan AWS Cloud9 konsol untuk membuat atau mengelola lingkungan AWS Cloud9 pengembangan, Anda melihat kesalahan yang berisi frasa yang mirip dengan “Pengguna tidak `arn:aws:iam::123456789012:user/MyUser` diizinkan untuk melakukan `cloud9:action` pada sumber daya`arn:aws:cloud9:us-east-2:123456789012:environment:12a34567b8cd9012345ef67abcd890e1`,” di mana:

- `arn:aws:iam::123456789012:user/MyUser` adalah Amazon Resource Name (ARN) dari pengguna yang meminta.
- `action` adalah nama operasi yang diminta oleh pengguna.
- `arn:aws:cloud9:us-east-2:123456789012:environment:12a34567b8cd9012345ef67abcd890e1` adalah ARN lingkungan yang pengguna diminta untuk menjalankan operasi.

Penyebab: Pengguna tempat Anda masuk ke AWS Cloud9 konsol tidak memiliki izin AWS akses yang benar untuk melakukan tindakan.

Solusi: Pastikan pengguna memiliki izin AWS akses yang benar, lalu coba lakukan tindakan lagi. Untuk informasi selengkapnya, lihat berikut ini:

- Langkah 2: Tambahkan izin AWS Cloud9 akses ke grup di Pengaturan Tim
- Langkah 6. Memungkinkan grup dan pengguna dalam organisasi untuk digunakan AWS Cloud9 di Pengaturan Korporasi
- Tentang peran akses anggota lingkungan di Bekerja dengan Lingkungan Bersama

## Tidak dapat terhubung ke lingkungan

Masalah: Pengguna tidak dapat terhubung ke lingkungan, dan terjebak pada tahap Menghubungkan.

Penyebab: Jika Anda mengubah izin `~/ .ssh/authorized_keys` file, menghapus AWS Cloud9 kunci dari file itu, atau menghapus file seluruhnya, masalah ini mungkin terjadi.

Solusi: Jangan hapus file ini. Jika Anda menghapusnya, Anda harus membuat ulang lingkungan Anda dan mungkin perlu melampirkan volume EBS dari lingkungan yang ada ke lingkungan baru EC2 . Ini untuk mengambil data Anda yang hilang. Jika ada izin yang hilang, pastikan file tersebut memiliki `Read-Write` izin. Ini untuk memungkinkan daemon SSH membacanya.

## Tidak dapat membuka lingkungan

Masalah: Saat Anda mencoba membuka lingkungan, IDE tidak ditampilkan selama lebih dari lima menit.

Kemungkinan penyebabnya:

- Pengguna IAM yang masuk ke AWS Cloud9 konsol tidak memiliki izin AWS akses yang diperlukan untuk membuka lingkungan.
- Jika lingkungan dikaitkan dengan instance komputasi AWS awan (misalnya, EC2 instans Amazon), kemungkinan itu mungkin benar:
  - VPC yang terkait dengan instance tidak disetel ke pengaturan yang benar untuk. AWS Cloud9
  - Instance sedang bertransisi antar status atau gagal dalam pemeriksaan status otomatis saat mencoba AWS Cloud9 terhubung ke instance.

- Jika lingkungan adalah lingkungan SSH, instance komputasi cloud terkait atau server Anda sendiri tidak diatur dengan benar untuk memungkinkan AWS Cloud9 mengaksesnya.

Solusi yang direkomendasikan:

- Pastikan pengguna IAM yang masuk ke AWS Cloud9 konsol memiliki izin AWS akses yang diperlukan untuk membuka lingkungan. Kemudian, coba buka lingkungan lagi. Untuk informasi selengkapnya, lihat yang berikut ini, atau hubungi Akun AWS administrator Anda:
  - Langkah 2: Tambahkan izin AWS Cloud9 akses ke grup di Pengaturan Tim
  - AWS kebijakan terkelola untuk AWS Cloud9 di Kontrol Otentikasi dan Akses
  - Contoh kebijakan terkelola pelanggan untuk tim, menggunakan AWS Cloud9 di Pengaturan Tim lanjutan
  - Contoh kebijakan yang dikelola pelanggan di Kontrol Otentikasi dan Akses
  - Mengubah Izin untuk pengguna IAM di Panduan Pengguna IAM
  - Memecahkan masalah Kebijakan IAM dalam Panduan Pengguna IAM

  Jika pengguna IAM yang masuk masih tidak dapat membuka lingkungan, coba keluar, lalu masuk kembali sebagai pengguna Akun AWS root atau pengguna administrator di akun. Kemudian, coba buka lingkungan lagi. Jika Anda tidak dapat membuka lingkungan dengan cara ini, kemungkinan besar ada masalah dengan izin akses pengguna IAM.
- Jika lingkungan dikaitkan dengan instance komputasi AWS awan (misalnya, EC2 instans Amazon), lakukan hal berikut:
  - Pastikan bahwa VPC yang terkait dengan instance diatur ke pengaturan yang benar untuk AWS Cloud9, dan kemudian coba buka lingkungan lagi. Untuk informasi selengkapnya, lihat Persyaratan Amazon VPC untuk AWS Cloud9.

    Jika VPC yang terkait dengan instans komputasi AWS awan disetel ke pengaturan yang benar AWS Cloud9 dan Anda masih tidak dapat membuka lingkungan, grup keamanan instans mungkin mencegah akses ke. AWS Cloud9Sebagai teknik pemecahan masalah saja, memeriksa grup keamanan untuk memastikan bahwa minimal, lalu lintas SSH masuk diizinkan melalui port 22 untuk semua alamat IP (`Anywhere` atau `0.0.0.0/0`). Untuk petunjuknya, lihat Menjelaskan Grup Keamanan Anda dan Memperbarui Aturan Grup Keamanan di Panduan EC2 Pengguna Amazon.

Untuk langkah pemecahan masalah VPC tambahan, tonton video 5 menit terkait Video Pusat AWS Pengetahuan: Apa yang dapat saya periksa jika saya tidak dapat terhubung ke instans di VPC? pada YouTube.

> **Warning**
>
> Setelah menyelesaikan pemecahan masalah, pastikan untuk mengatur aturan masuk ke rentang alamat yang sesuai. Untuk informasi selengkapnya, lihat the section called "Rentang alamat IP SSH inbound".

- Mulai ulang instance, pastikan instance berjalan dan lulus semua pemeriksaan sistem, lalu coba buka lingkungan lagi. Untuk informasi selengkapnya, lihat Memulai Ulang Instans Anda dan Melihat Pemeriksaan Status di Panduan EC2 Pengguna Amazon.

- Jika lingkungan adalah lingkungan SSH, pastikan instance komputasi awan yang terkait dengannya atau server Anda sendiri diatur dengan benar untuk memungkinkan AWS Cloud9 untuk mengaksesnya. Kemudian, coba buka lingkungan lagi. Untuk informasi selengkapnya, lihat Persyaratan host lingkungan SSH.

## Tidak dapat membuka AWS Cloud9 lingkungan: "Lingkungan ini saat ini tidak dapat diakses oleh kolaborator. Harap tunggu sampai penghapusan kredensial sementara terkelola selesai, atau kontak pemilik lingkungan ini."

Masalah: Jika kolaborator baru ditambahkan ke lingkungan oleh seseorang yang bukan pemilik lingkungan, kredensil sementara yang AWS dikelola akan dinonaktifkan. Kredensialnya dinonaktifkan saat Anda menghapus file. `~/.aws/credentials` Saat `~/.aws/credentials` file sedang dihapus, kolaborator baru tidak dapat mengakses AWS Cloud9 lingkungan.

Penyebab: Mencegah akses ke lingkungan saat kredensyal sementara yang AWS dikelola dihapus adalah tindakan pengamanan. Hal ini memungkinkan pemilik lingkungan mengonfirmasi bahwa hanya kolaborator tepercaya yang dapat mengakses kredenal terkelola. Jika mereka puas bahwa daftar kolaborator valid, pemilik lingkungan dapat mengaktifkan kembali kredensial terkelola sehingga mereka dapat dibagikan. Untuk informasi selengkapnya, lihat Mengontrol akses ke kredensial sementara terkelola AWS.

Solusi yang disarankan: Tunggu `~/.aws/credentials` file dihapus sepenuhnya sebelum mencoba lagi untuk membuka AWS Cloud9 lingkungan. Waktu tunggu maksimum untuk kredensial

kedaluwarsa adalah 15 menit. Atau, meminta pemilik lingkungan untuk mengaktifkan kembali atau nonaktifkan kredensial sementara terkelola. Setelah kredensial diaktifkan kembali atau dinonaktifkan, kolaborator dapat langsung mengakses lingkungan. Dengan mengalihkan status kredensyal terkelola ke ENABLED atau DISABLED, pemilik lingkungan memastikan bahwa kredensyal tidak tetap dalam status perantara. Stat perantara dapat mencegah kolaborator mengakses lingkungan.

> **Note**
>
> Misalkan pemilik lingkungan dan kolaborator milik yang sama Akun AWS. Kemudian, kolaborator dapat mengidentifikasi pemilik lingkungan untuk dihubungi dengan meninjau kartu untuk lingkungan di halaman Lingkungan Anda di konsol. Pemilik lingkungan juga tercantum dalam halaman Detail lingkungan.

## Kesalahan penghapusan lingkungan: "Satu atau lebih lingkungan gagal untuk menghapus"

Masalah: Saat Anda mencoba menghapus satu atau beberapa lingkungan di AWS Cloud9 konsol, pesan akan ditampilkan yang bertuliskan “satu atau beberapa lingkungan gagal dihapus,” dan setidaknya salah satu lingkungan tidak dihapus.

Kemungkinan penyebabnya: AWS CloudFormation mungkin memiliki masalah saat menghapus satu atau lebih lingkungan. AWS Cloud9 bergantung pada AWS CloudFormation untuk membuat dan menghapus lingkungan.

Solusi yang disarankan: Coba gunakan AWS CloudFormation untuk menghapus setiap lingkungan yang tidak dihapus.

1. Buka AWS CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
2. Pada bilah AWS navigasi, pilih Wilayah AWS untuk lingkungan.
3. Dalam daftar AWS CloudFormation tumpukan, pilih entri di mana nama Stack berisi nama lingkungan yang tidak dihapus dan Status adalah DELETE_FAILED. Misalnya, jika nama lingkungannya**`my-demo-environment`**, pilih tumpukan yang dimulai dengan nama aws-cloud9 -. my-demo-environment (Memilih kotak atau opsi di samping nama lingkungan, bukan nama lingkungan itu sendiri.)
4. Memilih Tindakan, Hapus Tumpukan.
5. Jika diminta, memilih Ya, Hapus.

Proses menghapus tumpukan mungkin memakan waktu beberapa menit.

Jika tumpukan menghilang dari daftar, lingkungan sekarang dihapus.

Jika tumpukan masih menampilkan DELETE_FAILED setelah beberapa menit, lingkungan masih belum dihapus. Anda dapat mencoba menghapus secara manual setiap sumber daya tumpukan yang gagal.

> **Note**
>
> Menghapus sumber daya tumpukan yang gagal secara manual tidak menghapus tumpukan itu sendiri dari tumpukan Anda Akun AWS.

Untuk menghapus sumber daya ini secara manual, lakukan hal berikut. Di AWS CloudFormation konsol, pilih tumpukan yang gagal, lalu pilih bagian Sumber Daya. Buka konsol AWS untuk setiap sumber daya dalam daftar ini, lalu gunakan konsol itu untuk menghapus sumber daya.

## Mengubah waktu tunggu untuk lingkungan di IDE AWS Cloud9

Masalah: Pengguna ingin memperbarui waktu tunggu untuk EC2 lingkungan Amazon.

Penyebab: Waktu batas waktu default adalah 30 menit. Ini mungkin terlalu pendek untuk beberapa pengguna.

Solusi yang direkomendasikan:

1. Buka lingkungan yang ingin Anda konfigurasikan.
2. Di AWS Cloud9 IDE, pada bilah menu, pilih AWS Cloud9Preferensi.
3. Di jendela Preferensi, gulir ke bagian  EC2instans Amazon.
4. Pilih nilai batas waktu dari daftar yang tersedia dan perbarui.

## Kesalahan menjalankan aplikasi SAM secara lokal di AWS Toolkit karena AWS Cloud9 lingkungan tidak memiliki cukup ruang disk

Masalah: Kesalahan terjadi ketika Anda menggunakan AWS Toolkit untuk menjalankan perintah AWS SAM CLI untuk aplikasi yang ditentukan oleh template SAM.

Kemungkinan penyebab: Saat Anda menjalankan dan men-debug aplikasi tanpa server secara lokal dengan Toolkit, gunakan AWS AWS SAM Docker gambar. Gambar-gambar ini menyediakan lingkungan runtime dan membangun alat yang meniru lingkungan Lambda yang Anda rencanakan untuk diterapkan.

Namun, jika lingkungan Anda tidak memiliki ruang disk yang cukup, Docker image yang menyediakan fitur-fitur ini tidak dapat dibangun dan aplikasi SAM lokal Anda gagal dijalankan. Jika ini terjadi, Anda mungkin menerima kesalahan di tab Output yang mirip dengan berikut ini.

```
Error: Could not find amazon/aws-sam-cli-emulation-image-python3.7:rapid-1.18.1 image
 locally and failed to pull it from docker.
```

Kesalahan ini berhubungan dengan aplikasi SAM yang dibangun menggunakan runtime Python. Anda mungkin menerima pesan yang sedikit berbeda, tergantung pada runtime yang Anda pilih untuk aplikasi Anda.

Solusi yang disarankan: Kosongkan ruang disk di lingkungan Anda sehingga Docker Image dapat membangun Hapus yang tidak terpakai Docker gambar dengan menjalankan perintah berikut di terminal IDE.

```
docker image prune -a
```

Jika Anda berulang kali mengalami masalah dengan perintah SAM CLI karena pembatasan ruang disk, beralih ke lingkungan pengembangan menggunakan jenis instance yang berbeda.

(kembali ke atas)

## Tidak dapat memuat IDE menggunakan versi sebelumnya Microsoft Edge penjelajah

Masalah: `HTTP403: FORBIDDEN` kesalahan dikembalikan saat mencoba memuat AWS Cloud9 IDE menggunakan Microsoft Edge peramban web.

Kemungkinan penyebabnya: AWS Cloud9 IDE tidak mendukung versi lama tertentu Microsoft Edge.

Solusi yang disarankan: Untuk memperbarui browser, pilih tombol elipsis (...) di Microsoft Edge bilah alat. Dari menu, pilih Pengaturan dan kemudian pilih  Tentang Microsoft Edge. Jika pembaruan diperlukan, pembaruan akan diunduh dan diinstal secara otomatis.

(kembali ke atas)

## Tidak dapat membuat struktur sub-folder/ home/ec2-user/environment/ home/ec2-user/environment di AWS Cloud9 IDE File Explorer.

Masalah: Saat Anda membuat struktur home/ec2-user/environment/home/ec2-user/environmentsub-folder/di AWS Cloud9 IDE File Explorer, Anda mendapatkan pesan kesalahan yang menyatakan bahwa tidak mungkin membuka direktori ini.

Kemungkinan penyebab: Saat ini tidak mungkin untuk membuat struktur home/ec2-user/ environmentsub-folder/dalam folder dengan nama yang sama menggunakan Sistem File AWS Cloud9 IDE. Anda tidak akan dapat mengakses file apa pun dalam direktori ini dari AWS Cloud9 IDE File Explorer, tetapi Anda akan dapat mengaksesnya menggunakan baris perintah. Masalah ini hanya memengaruhi jalur file/home/ec2-user/environment/home/ec2-user/environment, jalur file seperti/test/home/ec2-user/environmentdan/home/ec2-user/environment/testseharusnya berfungsi. Ini adalah masalah yang diketahui dan hanya mempengaruhi AWS Cloud9 IDE File Explorer.

Solusi yang disarankan: Gunakan nama dan struktur file yang berbeda.

(kembali ke atas)

## Tidak dapat membuat struktur sub-folder /projects/projects dalam File Explorer IDE untuk. AWS Cloud9 CodeCatalyst

Masalah: Ketika Anda membuat struktur sub-folder /projects/projects di AWS Cloud9 IDE File Explorer untuk CodeCatalyst, Anda mendapatkan pesan kesalahan yang menyatakan bahwa tidak mungkin untuk membuka direktori ini.

Kemungkinan penyebab: Saat ini tidak mungkin untuk membuat struktur sub-folder/proyek dalam folder dengan nama yang sama menggunakan File Explorer dari AWS Cloud9 IDE untuk. CodeCatalyst Anda tidak akan dapat mengakses file apa pun dalam direktori ini dari AWS Cloud9 IDE File Explorer, tetapi Anda akan dapat mengaksesnya menggunakan baris perintah. Masalah ini hanya memengaruhi jalur file /projects/projects, jalur file seperti /test/projects dan /projects/test seharusnya berfungsi. Ini adalah masalah yang diketahui dan hanya memengaruhi AWS Cloud9 IDE File Explorer untuk CodeCatalyst.

Solusi yang disarankan: Gunakan nama dan struktur file yang berbeda.

(kembali ke atas)

# Tidak dapat berinteraksi dengan jendela terminal AWS Cloud9 karena kesalahan **`tmux`** sesi

Masalah: Saat Anda mencoba meluncurkan jendela terminal baru AWS Cloud9, antarmuka baris perintah yang diharapkan tidak tersedia. Tidak ada command prompt dan Anda tidak dapat memasukkan teks. Pesan kesalahan seperti `tmux: need UTF-8 locale (LC_CTYPE)` dan `invalid LC_ALL, LC_CTYPE or LANG` dikembalikan.

Kemungkinan penyebabnya: Terminal yang tidak responsif mungkin disebabkan oleh kesalahan tmux. AWS Cloud9 menggunakan utilitas tmux. Dengan cara ini, informasi yang ditampilkan di terminal tetap ada bahkan ketika halaman dimuat ulang atau Anda terhubung kembali ke lingkungan pengembangan Anda.

Dalam `tmux` sesi, apa yang ditampilkan di jendela terminal ditangani oleh klien. Klien berkomunikasi ke server yang dapat mengelola beberapa sesi. Server dan klien berkomunikasi melalui soket yang terletak di `tmp` folder. Jika `tmp` folder hilang dari lingkungan pengembangan Anda atau izin yang terlalu ketat diterapkan padanya, `tmux` sesi tidak dapat dijalankan. Jika ini terjadi, jendela terminal di IDE menjadi tidak responsif.

Solusi yang disarankan: Jika `tmux` kesalahan mencegah Anda berinteraksi dengan jendela terminal, gunakan cara alternatif untuk membuat `tmp` folder dengan izin yang tepat. Dengan begitu, `tmux` sesi bisa berjalan. Salah satu solusinya adalah mengekspor ke `LC_CTYPE` dalam `.bash_profile` atau di `.bashrc` file. Solusi lain yang disarankan adalah menggunakan AWS Systems Manager untuk mengatur konfigurasi manajemen host. Ini memungkinkan akses ke instance yang relevan melalui EC2 konsol Amazon.

Menyiapkan manajemen host

1. Pertama, di AWS Cloud9 konsol, temukan nama instance lingkungan Anda. Anda dapat melakukannya dengan memilih panel yang relevan di halaman Lingkungan Anda dan memilih Lihat detail. Di halaman Environment details, pilih Go to Instance. Di EC2 konsol Amazon, konfirmasikan nama instance yang perlu Anda akses.
2. Sekarang pergi ke AWS Systems Manager konsol, dan di panel navigasi, pilih Pengaturan Cepat.
3. Di halaman Quick Setup, pilih Create.
4. Untuk jenis Konfigurasi, buka Manajemen Host dan pilih Buat.
5. Untuk opsi konfigurasi Sesuaikan Manajemen Host, di bagian Target, pilih Manual.

6. Pilih EC2 instance yang ingin Anda akses dan kemudian pilih Buat.

Menghubungkan ke instance dan menjalankan perintah

> **Note**
>
> Langkah-langkah berikut adalah untuk EC2 konsol baru.

1. Di EC2 konsol Amazon, di panel navigasi, pilih Instans dan pilih instance yang ingin Anda sambungkan.
2. Pilih Hubungkan.

   Jika Connect tidak diaktifkan, Anda mungkin perlu memulai instans terlebih dahulu.
3. Di panel Connect to instance Anda, untuk metode Connection, pilih Session Manager lalu pilih Connect.
4. Di jendela sesi terminal yang muncul, masukkan perintah berikut. Perintah ini membuat `tmp` folder dengan izin yang tepat sehingga soket tmux tersedia.

   ```
   sudo mkdir /tmp
   sudo chmod 777 /tmp
   sudo rmdir /tmp/tmux-*
   ```

(kembali ke atas)

# Amazon EC2

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan Amazon. EC2

## EC2 Instans Amazon tidak diperbarui secara otomatis

Masalah: Pembaruan sistem terbaru tidak diterapkan secara otomatis ke EC2 instans Amazon yang terhubung ke lingkungan AWS Cloud9 pengembangan.

Penyebab: Menerapkan pembaruan sistem terbaru secara otomatis dapat menyebabkan kode Anda atau EC2 instans Amazon berperilaku dengan cara yang tidak terduga, tanpa sepengetahuan atau persetujuan Anda sebelumnya.

Solusi yang direkomendasikan:

Terapkan pembaruan sistem ke EC2 instans Amazon secara teratur dengan mengikuti petunjuk dalam Memperbarui Perangkat Lunak Instans di Panduan EC2 Pengguna Amazon.

Untuk menjalankan perintah pada instance, Anda dapat menggunakan sesi terminal di AWS Cloud9 IDE dari lingkungan yang terhubung ke instance.

Atau, Anda dapat menggunakan utilitas akses jarak jauh SSH seperti ssh atau PuTTY untuk terhubung ke instance. Untuk melakukan ini, dari komputer lokal Anda, gunakan utilitas pembuatan key pair SSH seperti ssh-keygen atau PuTTYgen. Gunakan AWS Cloud9 IDE dari lingkungan yang terhubung ke instance untuk menyimpan kunci publik yang dihasilkan pada instance. Kemudian gunakan utilitas akses jarak jauh SSH bersama dengan kunci pribadi yang dihasilkan untuk mengakses instance. Untuk informasi lebih lanjut, lihat dokumentasi khusus sistem Anda.

## AWS CLI atau AWS-shell error: “Token keamanan yang disertakan dalam permintaan tidak valid” di lingkungan EC2

Masalah: Saat Anda mencoba menggunakan AWS Command Line Interface (AWS CLI) atau AWS-shell untuk menjalankan perintah di AWS Cloud9 IDE untuk EC2 lingkungan, kesalahan akan ditampilkan: “Token keamanan yang disertakan dalam permintaan tidak valid.”

Penyebab: Token keamanan yang tidak valid dapat terjadi jika Anda kredensial sementara terkelola AWS yang diaktifkan dan salah satu berikut terjadi:

- Anda mencoba menjalankan perintah yang tidak diizinkan oleh kredensi sementara yang AWS dikelola. Untuk daftar perintah yang diizinkan, lihat Tindakan yang didukung oleh kredensial sementara yang AWS dikelola.
- Kredensi sementara yang AWS dikelola secara otomatis kedaluwarsa setelah 15 menit.
- Kredensi sementara yang AWS dikelola untuk lingkungan bersama dinonaktifkan karena anggota baru ditambahkan oleh orang lain selain pemilik lingkungan.

Solusi yang direkomendasikan:

- Jalankan hanya perintah-perintah yang diizinkan oleh kredensi sementara yang AWS dikelola. Jika Anda perlu menjalankan perintah yang tidak diizinkan oleh kredensil sementara yang AWS dikelola, konfigurasikan AWS CLI atau AWS-shell di lingkungan dengan sekumpulan kredensil permanen. Ini menghilangkan batasan ini. Untuk petunjuk, silakan lihat Membuat dan menyimpan kredensial akses permanen di Lingkungan.

- Untuk kredensil yang dinonaktifkan atau kedaluwarsa, pastikan bahwa pemilik lingkungan membuka lingkungan sehingga AWS Cloud9 dapat menyegarkan kredensil sementara di lingkungan. Untuk informasi selengkapnya, lihat Mengontrol akses ke kredensial sementara terkelola AWS.

## Tidak dapat terhubung ke EC2 lingkungan karena alamat IP VPC digunakan oleh Docker

Masalah: Untuk EC2 lingkungan, jika Anda meluncurkan EC2 instance ke VPC Amazon yang menggunakan blok IPv4 Classless Inter-Domain Routing (CIDR)`172.17.0.0/16`, koneksi mungkin terhenti saat Anda mencoba membuka lingkungan tersebut.

Penyebab: Docker menggunakan perangkat lapisan tautan yang disebut jaringan jembatan yang memungkinkan kontainer yang terhubung ke jaringan jembatan yang sama untuk berkomunikasi. AWS Cloud9 membuat kontainer yang menggunakan jembatan default untuk komunikasi kontainer. Jembatan default biasanya menggunakan subnet `172.17.0.0/16` untuk jaringan kontainer.

Jika subnet VPC untuk instance lingkungan Anda menggunakan rentang alamat yang sama yang sudah digunakan oleh DockerKonflik alamat IP mungkin terjadi. Jadi, ketika AWS Cloud9 mencoba terhubung ke instance-nya, koneksi itu dirutekan oleh tabel rute gateway ke Docker jembatan. Ini AWS Cloud9 mencegah menghubungkan ke EC2 instance yang mendukung lingkungan pengembangan.

Solusi yang disarankan: Untuk mengatasi konflik alamat IP yang disebabkan oleh Amazon VPC dan Docker menggunakan blok alamat IPv4 CIDR yang sama, konfigurasikan VPC baru untuk instance yang mendukung EC2 lingkungan Anda. Untuk VPC baru ini, konfigurasikan blok CIDR yang berbeda dari `172.17.0.0/16`. (Anda tidak dapat mengubah rentang alamat IP dari VPC atau subnet yang ada.)

Untuk informasi selengkapnya, lihat VPC dan pengukuran subnet dalam Panduan Pengguna Amazon VPC.

## Tidak dapat membuat struktur sub-folder/ home/ec2-user/environment/ home/ec2-user/environment di AWS Cloud9 IDE File Explorer.

Masalah: Saat Anda membuat struktur home/ec2-user/environment/home/ec2-user/environmentsub-folder/di AWS Cloud9 IDE File Explorer, Anda mendapatkan pesan kesalahan yang menyatakan bahwa tidak mungkin membuka direktori ini.

Kemungkinan penyebab: Saat ini tidak mungkin untuk membuat struktur home/ec2-user/environmentsub-folder/dalam folder dengan nama yang sama menggunakan Sistem File AWS Cloud9 IDE. Anda tidak akan dapat mengakses file apa pun dalam direktori ini dari AWS Cloud9 IDE File Explorer, tetapi Anda akan dapat mengaksesnya menggunakan baris perintah. Masalah ini hanya memengaruhi jalur file/home/ec2-user/environment/home/ec2-user/environment, jalur file seperti/test/home/ec2-user/environmentdan/home/ec2-user/environment/testseharusnya berfungsi. Ini adalah masalah yang diketahui dan hanya mempengaruhi AWS Cloud9 IDE File Explorer.

Solusi yang disarankan: Gunakan nama dan struktur file yang berbeda.

## Tidak dapat diluncurkan AWS Cloud9 dari konsol saat konfigurasi AWS License Manager lisensi dikaitkan dengan EC2 instans Amazon

Masalah: Saat Anda mencoba meluncurkan AWS Cloud9 EC2 lingkungan dari konsol, pesan kesalahan akan `unable to access your environment` dikembalikan.

Kemungkinan penyebabnya: AWS License Manager merampingkan pengelolaan lisensi vendor perangkat lunak di seluruh. AWS Cloud Saat menyiapkan License Manager, Anda membuat konfigurasi lisensi, yang merupakan kumpulan aturan lisensi berdasarkan ketentuan perjanjian perusahaan Anda. Konfigurasi lisensi ini dapat dilampirkan ke mekanisme, seperti Amazon Machine Image (AMI) atau AWS CloudFormation. Anda dapat menggunakan salah satu mekanisme ini untuk meluncurkan EC2 instance.

Versi lama AWSCloud9ServiceRolePolicyuntuk peran AWSService RoleFor AWSCloud9 terkait layanan (SLR) saat ini tidak menyertakan kondisi sumber daya. `license-configuration` Karena AWS Cloud9 itu, tidak diperbolehkan untuk memulai dan menghentikan instance-nya. Jadi, AWS Cloud9 ditolak akses ke EC2 instance Amazon-nya, dan kesalahan dikembalikan.

Solusi yang disarankan: Jika Anda tidak dapat mengakses AWS Cloud9 lingkungan yang ada dan menggunakan License Manager, ganti peran AWSCloud9ServiceRolePolicyterkait layanan lama dengan versi SLR yang secara eksplisit mengizinkan EC2 tindakan saat berlaku untuk instance. `license-configuration` Anda dapat mengganti peran lama hanya dengan menghapusnya. Peran yang diperbarui kemudian dibuat secara otomatis.

## Tidak dapat menjalankan beberapa perintah atau skrip di lingkungan EC2

Masalah: Setelah Anda membuka lingkungan AWS Cloud9 EC2 pengembangan, Anda tidak dapat menginstal beberapa jenis paket, menjalankan perintah seperti `yum` atau`apt`, atau menjalankan skrip yang berisi perintah yang biasanya bekerja dengan sistem operasi Linux lainnya.

Penyebab: EC2 Instans Amazon yang AWS Cloud9 digunakan untuk EC2 lingkungan bergantung pada Amazon Linux (yang didasarkan pada Red Hat Enterprise Linux (RHEL)) atau Server Ubuntu.

Solusi: Jika Anda menginstal atau mengelola paket atau menjalankan perintah atau skrip di IDE untuk EC2 lingkungan, pastikan mereka kompatibel dengan RHEL (untuk Amazon Linux) atau Ubuntu Server, tergantung pada instance untuk lingkungan itu.

## Pelaporan pesan kesalahan " AWSCloud9SSMInstanceProfil Instans Profil tidak ada di akun" saat membuat EC2 lingkungan menggunakan AWS CloudFormation

Masalah: Saat menggunakan AWS CloudFormation sumber daya AWS::Cloud9::EnvironmentEC2 untuk membuat EC2 lingkungan, pengguna menerima pesan galat bahwa Profil AWSCloud9 SSMInstance profil Instance tidak ada di akun.

Penyebab: Saat membuat EC2 lingkungan no-ingress, Anda harus membuat peran layanan `AWSCloud9SSMAccessRole` dan profil instance. `AWSCloud9SSMInstanceProfile` Sumber daya IAM ini memungkinkan Systems Manager untuk mengelola EC2 instans yang mendukung lingkungan pengembangan Anda.

Jika Anda membuat lingkungan no-ingress dengan konsol tersebut, `AWSCloud9SSMAccessRole` dan `AWSCloud9SSMInstanceProfile` dibuat secara otomatis. Tetapi ketika menggunakan AWS CloudFormation atau AWS CLI untuk membuat lingkungan no-ingress pertama Anda, Anda harus membuat sumber daya IAM ini secara manual.

Solusi yang disarankan: Untuk informasi tentang mengedit AWS CloudFormation template Anda dan memperbarui izin IAM, lihat Menggunakan AWS CloudFormation untuk membuat lingkungan tanpa masuknya EC2

## Pelaporan pesan kesalahan "tidak diizinkan untuk **`perform: ssm:StartSession`** pada sumber daya" saat membuat EC2 lingkungan menggunakan AWS CloudFormation

Masalah: Saat menggunakan AWS CloudFormation sumber daya AWS::Cloud9::EnvironmentEC2 untuk membuat EC2 lingkungan, pengguna menerima `AccessDeniedException` dan diberi tahu bahwa mereka "tidak berwenang untuk melakukan: `ssm:StartSession` pada sumber daya."

Penyebab: Pengguna tidak memiliki izin untuk memanggil `StartSession` API yang diperlukan sebagai bagian dari konfigurasi untuk EC2 lingkungan yang menggunakan Systems Manager untuk instance no-ingress.

Solusi yang disarankan: Untuk informasi tentang mengedit AWS CloudFormation template Anda dan memperbarui izin IAM, lihat. Menggunakan AWS CloudFormation untuk membuat lingkungan tanpa masuknya EC2

## Pesan kesalahan melaporkan tidak ada otorisasi “untuk melakukan: **`iam:GetInstanceProfile`** pada sumber daya: profil instance**`AWSCloud9SSMInstanceProfile`**" saat membuat EC2 lingkungan menggunakan AWS CLI

Masalah: Saat menggunakan AWS CLIuntuk membuat EC2 lingkungan, pengguna menerima `AccessDeniedException` dan diberi tahu bahwa AWS Cloud9 lingkungan mereka tidak diizinkan “untuk melakukan iam: GetInstanceProfile on resource: instance profile`AWSCloud9SSMInstanceProfile`.”

Penyebab: AWS Cloud9 tidak memiliki izin untuk memanggil `StartSession` API yang diperlukan sebagai bagian dari konfigurasi untuk EC2 lingkungan yang menggunakan Systems Manager untuk instance tanpa masuk.

Solusi yang disarankan: Untuk informasi tentang menambahkan peran `AWSCloud9SSMAccessRole` layanan yang diperlukan dan `AWSCloud9SSMInstanceProfile` ke AWS Cloud9 lingkungan Anda, lihatMengelola profil instans untuk Systems Manager dengan AWS CLI.

## Kegagalan untuk membuat lingkungan ketika default enkripsi diterapkan ke Amazon EBS volume

Masalah: `Failed to create environments. The development environment '[environment-ID]' failed to create` kesalahan dikembalikan saat mencoba membuat EC2 lingkungan Amazon.

Kemungkinan penyebabnya: Jika AWS Cloud9 IDE Anda menggunakan volume Amazon EBS yang secara default dienkripsi, peran yang AWS Identity and Access Management ditautkan layanan AWS Cloud9 memerlukan akses ke volume EBS ini AWS KMS keys . Jika akses tidak disediakan, AWS Cloud9 IDE mungkin gagal diluncurkan, dan mungkin sulit untuk men-debug masalah.

Solusi yang disarankan: Untuk menyediakan akses, tambahkan peran terkait layanan untuk AWS Cloud9,`AWSServiceRoleForAWSCloud9`, ke kunci terkelola pelanggan yang digunakan oleh volume Amazon EBS Anda.

Untuk informasi selengkapnya tentang tugas ini, lihat Membuat AWS Cloud9 yang menggunakan volume Amazon EBS dengan enkripsi default di Pola Panduan AWS Preskriptif.

## Kesalahan VPC untuk akun EC2 -Classic: “Tidak dapat mengakses lingkungan Anda”

Masalah: EC2 -Classic diperkenalkan dalam rilis asli Amazon EC2. Jika Anda menggunakan Akun AWS yang telah disiapkan sebelum 4 Desember 2013, kesalahan ini mungkin terjadi jika Anda tidak mengonfigurasi VPC dan subnet Amazon saat membuat lingkungan AWS Cloud9 EC2 pengembangan.

Jika Anda menerima pengaturan VPC default, EC2 instans Amazon diluncurkan ke jaringan EC2 -Classic. Instance tidak diluncurkan ke subnet dari VPC default. Pesan berikut ditampilkan saat lingkungan gagal dibuat:

Kesalahan Lingkungan

Tidak dapat mengakses lingkungan Anda

Penciptaan lingkungan gagal dengan galat: Sumber daya berikut gagal membuat: [Instans]. . Rollback diminta oleh pengguna..

Anda dapat mengonfirmasi bahwa kesalahan disebabkan oleh EC2 instance yang tidak berada di VPC default. Gunakan AWS CloudFormation untuk melihat riwayat peristiwa tumpukan untuk lingkungan pengembangan.

1. Buka AWS CloudFormation konsol. Untuk informasi selengkapnya, lihat: Masuk ke konsol AWS CloudFormation .
2. Di AWS CloudFormation konsol, pilih Stacks.
3. Pada halaman Tumpukan, memilih nama lingkungan pengembangan yang gagal untuk membuat.
4. Pada halaman Stack details, pilih tab Events dan periksa entri berikut:

   Status: CREATE_FAILED

> Alasan status: AssociatePublicIpAddress Parameter ini hanya didukung oleh peluncuran VPC. [...]

Penyebab: Lingkungan AWS Cloud9 pengembangan harus dikaitkan dengan VPC Amazon yang memenuhi persyaratan VPC tertentu. Untuk akun dengan EC2 -Classic diaktifkan, menerima pengaturan jaringan default saat membuat EC2 lingkungan berarti EC2 instans yang diperlukan tidak diluncurkan ke VPC. Sebagai gantinya, instance diluncurkan ke jaringan EC2 -Classic.

Solusi yang disarankan: Dengan akun EC2 -Classic, Anda harus memilih VPC dan subnet saat membuat lingkungan. EC2 Pada halaman Konfigurasi pengaturan, di bagian Pengaturan jaringan (lanjutan), pilih VPC dan subnet tempat Anda dapat meluncurkan instance Anda EC2 .

# AWS Layanan lainnya

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan layanan lain. AWS

## Tidak dapat membuat struktur sub-folder /projects/projects dalam File Explorer IDE untuk. AWS Cloud9 CodeCatalyst

Masalah: Ketika Anda membuat struktur sub-folder /projects/projects di AWS Cloud9 IDE File Explorer untuk CodeCatalyst, Anda mendapatkan pesan kesalahan yang menyatakan bahwa tidak mungkin untuk membuka direktori ini.

Kemungkinan penyebab: Saat ini tidak mungkin untuk membuat struktur sub-folder/proyek dalam folder dengan nama yang sama menggunakan File Explorer dari AWS Cloud9 IDE untuk. CodeCatalyst Anda tidak akan dapat mengakses file apa pun dalam direktori ini dari AWS Cloud9 IDE File Explorer, tetapi Anda akan dapat mengaksesnya menggunakan baris perintah. Masalah ini hanya memengaruhi jalur file /projects/projects, jalur file seperti /test/projects dan /projects/test seharusnya berfungsi. Ini adalah masalah yang diketahui dan hanya memengaruhi AWS Cloud9 IDE File Explorer untuk CodeCatalyst.

Solusi yang disarankan: Gunakan nama dan struktur file yang berbeda.

## Tidak dapat menampilkan aplikasi yang sedang berjalan di luar IDE

Masalah: Ketika Anda atau orang lain mencoba untuk menampilkan aplikasi Anda berjalan di tab peramban web di luar IDE, tab peramban web menampilkan kesalahan, atau tab kosong.

Kemungkinan penyebabnya:

- Aplikasi tidak berjalan di IDE.
- Pastikan bahwa aplikasi berjalan dengan IP `127.0.0.1` atau `localhost`.
- Aplikasi ini berjalan di lingkungan AWS Cloud9 EC2 pengembangan. Selain itu, satu atau beberapa grup keamanan yang terkait dengan EC2 instans Amazon terkait tidak mengizinkan lalu lintas masuk melalui protokol, port, atau alamat IP yang dibutuhkan aplikasi.
- Aplikasi ini berjalan di lingkungan pengembangan AWS Cloud9 SSH untuk instance komputasi AWS awan (misalnya, EC2 instance Amazon). Selain itu, ACL jaringan untuk subnet di virtual private cloud (VPC) yang terkait dengan instance yang sesuai tidak memungkinkan lalu lintas masuk melalui protokol, port, atau alamat IP yang dibutuhkan aplikasi.
- URL tidak benar.
- URL di tab pratinjau aplikasi diminta alih-alih alamat IP publik instans.
- Anda mencoba untuk pergi ke alamat yang berisi IP dari `127.0.0.1` atau`localhost`. IPs Upaya ini untuk mengakses sumber daya di komputer lokal Anda alih-alih sumber daya di lingkungan.
- Alamat IP publik instans telah berubah.
- Permintaan web berasal dari jaringan pribadi virtual (VPN) yang mem-blok lalu lintas melalui protokol, port, atau alamat IP yang dibutuhkan aplikasi.
- Aplikasi ini berjalan di lingkungan SSH. Namun, server Anda atau jaringan terkait tidak mengizinkan lalu lintas melalui protokol, port, atau alamat IP yang dibutuhkan aplikasi.

Solusi yang direkomendasikan:

- Pastikan bahwa aplikasi berjalan di IDE.
- Pastikan bahwa aplikasi tidak berjalan dengan IP `127.0.0.1` atau`localhost`. Untuk contoh di Node.js dan Python, lihat Menjalankan aplikasi.
- Misalkan aplikasi berjalan pada instance komputasi AWS awan (misalnya, EC2 instance Amazon). Kemudian, pastikan semua grup keamanan yang terkait dengan instance yang sesuai memungkinkan lalu lintas masuk melalui protokol, port, dan alamat IP yang dibutuhkan aplikasi. Untuk instruksi, lihat Langkah 2: Siapkan grup keamanan untuk instans di Bagikan aplikasi yang sedang berjalan melalui internet. Lihat juga Grup Keamanan untuk VPC Anda di Panduan Pengguna Amazon VPC.
- Misalkan aplikasi berjalan pada instance komputasi AWS awan. Selain itu, ACL jaringan ada untuk subnet di VPC yang terkait dengan instance yang sesuai. Kemudian, pastikan bahwa ACL jaringan

memungkinkan lalu lintas masuk melalui protokol, port, dan alamat IP yang dibutuhkan aplikasi. Untuk instruksi, lihat Langkah 3: Siapkan subnet untuk instans di Bagikan aplikasi yang sedang berjalan melalui internet. Lihat juga Jaringan ACLs di Panduan Pengguna VPC Amazon.

- Pastikan URL yang meminta, termasuk protokol (dan port, jika harus ditentukan), benar. Untuk informasi selengkapnya, lihat Langkah 4: Bagikan URL aplikasi yang sedang berjalan di Bagikan aplikasi yang sedang berjalan melalui internet.
- Kami tidak menyarankan meminta URL dengan format `https://12a34567b8cd9012345ef67abcd890e1.vfs.cloud9.us-east-2.amazonaws.com/` (di `12a34567b8cd9012345ef67abcd890e1` mana ID yang AWS Cloud9 ditetapkan ke lingkungan, dan `us-east-2` merupakan ID AWS Wilayah untuk lingkungan). URL ini bekerja hanya ketika IDE untuk lingkungan terbuka dan aplikasi berjalan di peramban web yang sama.
- Misalkan Anda mencoba untuk pergi ke alamat yang berisi IP dari `127.0.0.1` atau`localhost`. Cobalah pergi ke alamat non-lokal yang benar untuk aplikasi yang sedang berjalan sebagai gantinya. Untuk informasi selengkapnya, lihat Membagikan aplikasi yang sedang berjalan melalui internet.
- Misalkan aplikasi berjalan pada instance komputasi AWS awan. Tentukan apakah alamat IP publik instans telah berubah. Alamat IP publik instans mungkin berubah kapan pun saat instans dimulai ulang. Untuk mencegah alamat IP ini berubah, Anda dapat mengalokasikan alamat IP elastis dan beri tugas ke instans berjalan. Untuk informasi selengkapnya, lihat Langkah 4: Bagikan URL aplikasi yang sedang berjalan di Bagikan aplikasi yang sedang berjalan melalui internet.
- Jika permintaan web berasal dari VPN, pastikan bahwa VPN mengizinkan lalu lintas melalui protokol, port, dan alamat IP yang diperlukan aplikasi. Jika Anda tidak dapat membuat perubahan pada VPN Anda, lihat administrator jaringan Anda. Atau, buat permintaan web dari jaringan yang berbeda jika memungkinkan.
- Misalkan aplikasi berjalan di lingkungan SSH untuk server Anda sendiri. Pastikan server Anda dan jaringan terkait memungkinkan lalu lintas melalui protokol, port, dan alamat IP yang dibutuhkan aplikasi. Jika Anda tidak dapat membuat perubahan pada server atau jaringan terkait, lihat server atau administrator jaringan Anda.
- Coba jalankan aplikasi dari terminal di lingkungan dengan menjalankan perintah `curl`, diikuti oleh URL. Jika perintah ini menampilkan pesan kesalahan, mungkin ada beberapa masalah lain yang tidak terkait dengannya AWS Cloud9.

## Kesalahan saat menjalankan AWS Toolkit: “Lingkungan Anda kehabisan inode, harap tingkatkan batas 'fs.inotify.max_user_watches'.”

Masalah: Utilitas pengamat file yang digunakan AWS Toolkit mendekati batas saat ini atau kuota file yang dapat ditonton.

Penyebab: AWS Toolkit menggunakan utilitas pengamat file yang memantau perubahan pada file dan direktori. Ketika utilitas hampir mencapai kuota file saat ini yang dapat ditonton, pesan peringatan muncul.

Solusi yang Direkomendasikan:  Untuk meningkatkan jumlah maksimum file yang dapat ditangani oleh file watcher, lakukan hal berikut:

1. Untuk meluncurkan sesi terminal baru, memilih Jendela, Terminal Baru pada bilah menu.
2. Masukkan perintah berikut.

```
sudo bash -c 'echo "fs.inotify.max_user_watches=524288" >> /etc/sysctl.conf' &&
 sudo sysctl -p
```

## Kesalahan menjalankan fungsi lokal Lambda: Tidak dapat menginstal SAM Local

Masalah: Setelah Anda mencoba menjalankan versi lokal AWS Lambda fungsi di AWS Cloud9 IDE, kotak dialog akan ditampilkan. Kotak dialog menyatakan bahwa AWS Cloud9 mengalami kesulitan menginstal SAM Local. AWS Cloud9 membutuhkan SAM Local untuk menjalankan versi AWS Lambda fungsi lokal di IDE. Sampai SAM Local diinstal, Anda tidak dapat menjalankan versi lokal fungsi Lambda di IDE.

Penyebab: tidak AWS Cloud9 dapat menemukan SAM Local di jalur yang diharapkan di lingkungan, yaitu`~/.c9/bin/sam`. Ini karena SAM Local belum diinstal, atau jika diinstal, tidak AWS Cloud9 dapat menemukannya di lokasi itu.

Solusi yang disarankan: Anda dapat menunggu AWS Cloud9 untuk mencoba menyelesaikan instalasi SAM Local, atau Anda dapat menginstalnya sendiri.

Untuk melihat bagaimana AWS Cloud9 dengan mencoba menginstal SAM Local, pilih Window, Installer pada menu bar.

Untuk menginstal SAM Local sendiri, ikuti instruksi dalam Menginstal AWS SAM CLI di Linux di Panduan AWS Serverless Application Model Pengembang.

## AWS Control Tower kesalahan saat mencoba membuat EC2 lingkungan Amazon menggunakan AWS Cloud9: “Pembuatan lingkungan gagal dengan kesalahan: Kait berikut gagal: [: :GuardControlTower: :Hook].”

Masalah: Masalah kompatibilitas ada dengan AWS Cloud9 dan CT kontrol AWS Control Tower proaktif. EC2.PR.8. Jika kontrol ini diaktifkan, Anda tidak dapat membuat EC2 lingkungan di AWS Cloud9.

Penyebab: AWS Control Tower mengharapkan AssociatePublicIpAddressparameter berada di AWS CloudFormation template. Parameter ini tidak dapat ditambahkan saat ini.

Solusi yang disarankan: Nonaktifkan kontrol CT. EC2.PR.8 dari AWS Control Tower konsol dan buat ulang lingkungan di. AWS Cloud9

## Kegagalan untuk membuat lingkungan ketika default enkripsi diterapkan ke Amazon EBS volume

Masalah: `Failed to create environments. The development environment '[environment-ID]' failed to create` kesalahan dikembalikan saat mencoba membuat EC2 lingkungan Amazon.

Kemungkinan penyebabnya: Jika AWS Cloud9 IDE Anda menggunakan volume Amazon EBS yang secara default dienkripsi, peran yang AWS Identity and Access Management ditautkan layanan AWS Cloud9 memerlukan akses ke volume EBS ini AWS KMS keys . Jika akses tidak disediakan, AWS Cloud9 IDE mungkin gagal diluncurkan, dan mungkin sulit untuk men-debug masalah.

Solusi yang disarankan: Untuk menyediakan akses, tambahkan peran terkait layanan untuk AWS Cloud9,`AWSServiceRoleForAWSCloud9`, ke kunci terkelola pelanggan yang digunakan oleh volume Amazon EBS Anda.

Untuk informasi selengkapnya tentang tugas ini, lihat Membuat AWS Cloud9 yang menggunakan volume Amazon EBS dengan enkripsi default di Pola Panduan AWS Preskriptif.

(kembali ke atas)

## Tidak dapat diluncurkan AWS Cloud9 dari konsol saat konfigurasi AWS License Manager lisensi dikaitkan dengan EC2 instans Amazon

Masalah: Saat Anda mencoba meluncurkan AWS Cloud9 EC2 lingkungan dari konsol, pesan kesalahan akan `unable to access your environment` dikembalikan.

Kemungkinan penyebabnya: AWS License Manager merampingkan pengelolaan lisensi vendor perangkat lunak di seluruh. AWS Cloud Saat menyiapkan License Manager, Anda membuat konfigurasi lisensi, yang merupakan kumpulan aturan lisensi berdasarkan ketentuan perjanjian perusahaan Anda. Konfigurasi lisensi ini dapat dilampirkan ke mekanisme, seperti Amazon Machine Image (AMI) atau AWS CloudFormation. Anda dapat menggunakan salah satu mekanisme ini untuk meluncurkan EC2 instance.

Versi lama AWSCloud9ServiceRolePolicyuntuk peran AWSService RoleFor AWSCloud9 terkait layanan (SLR) saat ini tidak menyertakan kondisi sumber daya. `license-configuration` Karena AWS Cloud9 itu, tidak diperbolehkan untuk memulai dan menghentikan instance-nya. Jadi, AWS Cloud9 ditolak akses ke EC2 instance Amazon-nya, dan kesalahan dikembalikan.

Solusi yang disarankan: Jika Anda tidak dapat mengakses AWS Cloud9 lingkungan yang ada dan menggunakan License Manager, ganti peran AWSCloud9ServiceRolePolicyterkait layanan lama dengan versi SLR yang secara eksplisit mengizinkan EC2 tindakan saat berlaku untuk instance. `license-configuration` Anda dapat mengganti peran lama hanya dengan menghapusnya. Peran yang diperbarui kemudian dibuat secara otomatis.

(kembali ke atas)

# Pratinjau aplikasi

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan pratinjau aplikasi.

## Setelah memuat ulang lingkungan, Anda harus me-refresh pratinjau aplikasi

Masalah: Setelah Anda memuat ulang lingkungan yang menampilkan tab pratinjau aplikasi, tab tidak menampilkan pratinjau aplikasi.

Penyebab: Terkadang pengguna menulis kode yang dapat menjalankan loop tak terbatas. Atau kode mereka dapat menggunakan begitu banyak memori sehingga AWS Cloud9 IDE mungkin berhenti sejenak atau berhenti saat pratinjau aplikasi sedang berjalan. Agar hal ini AWS Cloud9 tidak terjadi, jangan memuat ulang tab pratinjau aplikasi setiap kali lingkungan dimuat ulang.

Solusi: Setelah Anda memuat ulang lingkungan yang menampilkan tab pratinjau aplikasi, untuk menampilkan pratinjau aplikasi, memilih Klik untuk memuat halaman pada tab.

## Pemberitahuan pratinjau aplikasi atau pratinjau file: "Cookie pihak ke tiga dinonaktifkan"

Masalah: Ketika Anda mencoba untuk melakukan pratinjau aplikasi atau file, pemberitahuan akan ditampilkan dengan olahpesan berikut: "Fungsi pratinjau dinonaktifkan karena peramban Anda memiliki cookie pihak ke tiga yang dinonaktifkan."

Penyebab: Cookie pihak ketiga tidak diperlukan untuk membuka AWS Cloud9 IDE. Namun, Anda harus mengaktifkan cookie pihak ketiga untuk menggunakan fitur Pratinjau Aplikasi atau Pratinjau File.

Solusi: Mengaktifkan cookie pihak ke tiga di peramban web Anda, muat ulang IDE Anda, dan kemudian coba buka pratinjau lagi.

- Apple Safari: Kelola cookie dan data situs web di Safari di situs web Dukungan Apple.
- Google Chrome: Ubah setelan cookie Anda di Hapus, aktifkan, dan kelola cookie di Chrome di situs web Bantuan Google Chrome.
- Internet Explorer: Blokir atau izinkan cookie di Hapus dan kelola cookie di situs web Dukungan Microsoft.
- Microsoft Edge: Memblokir cookie pihak ketiga di situs web Dukungan Microsoft.
- Mozilla Firefox: Terima pengaturan cookie pihak ketiga di Aktifkan dan nonaktifkan cookie yang digunakan situs web untuk melacak preferensi Anda di situs web Dukungan Mozilla.
- Browser web lainnya: lihat dokumentasi browser web itu.

Jika browser web Anda memungkinkan perincian ini, Anda dapat mengaktifkan cookie pihak ketiga hanya untuk. AWS Cloud9 Untuk melakukan ini, tentukan domain berikut, tergantung pada yang didukung di Wilayah AWS mana Anda ingin menggunakan AWS Cloud9.

| AWS Wilayah | Domain |
| --- | --- |
| US East (N. Virginia) | `*.vfs.cloud9.us-east-1.amazonaws.com` |

| AWS Wilayah | Domain |
| --- | --- |
| | `vfs.cloud9.us-east-1.amazonaws.com` |
| AS Timur (Ohio) | `*.vfs.cloud9.us-east-2.amazonaws.com`<br>`vfs.cloud9.us-east-2.amazonaws.com` |
| AS Barat (California Utara) | `*.vfs.cloud9.us-west-1.amazonaws.com`<br>`vfs.cloud9.us-west-1.amazonaws.com` |
| AS Barat (Oregon) | `*.vfs.cloud9.us-west-2.amazonaws.com`<br>`vfs.cloud9.us-west-2.amazonaws.com` |
| Afrika (Cape Town) | `*.vfs.cloud9.af-south-1.amazonaws.com`<br>`vfs.cloud9.af-south-1.amazonaws.com` |
| Asia Pasifik (Hong Kong) | `*.vfs.cloud9.ap-east-1.amazonaws.com`<br>`vfs.cloud9.ap-east-1.amazonaws.com` |
| Asia Pasifik (Mumbai) | `*.vfs.cloud9.ap-south-1.amazonaws.com`<br>`vfs.cloud9.ap-south-1.amazonaws.com` |

| AWS Wilayah | Domain |
| --- | --- |
| Asia Pasifik (Osaka) | `*.vfs.cloud9.ap-northeast-3.amazonaws.com`<br>`vfs.cloud9.ap-northeast-3.amazonaws.com` |
| Asia Pasifik (Seoul) | `*.vfs.cloud9.ap-northeast-2.amazonaws.com`<br>`vfs.cloud9.ap-northeast-2.amazonaws.com` |
| Asia Pasifik (Singapura) | `*.vfs.cloud9.ap-southeast-1.amazonaws.com`<br>`vfs.cloud9.ap-southeast-1.amazonaws.com` |
| Asia Pasifik (Sydney) | `*.vfs.cloud9.ap-southeast-2.amazonaws.com`<br>`vfs.cloud9.ap-southeast-2.amazonaws.com` |
| Asia Pasifik (Tokyo) | `*.vfs.cloud9.ap-northeast-1.amazonaws.com`<br>`vfs.cloud9.ap-northeast-1.amazonaws.com` |
| Kanada (Pusat) | `*.vfs.cloud9.ca-central-1.amazonaws.com`<br>`vfs.cloud9.ca-central-1.amazonaws.com` |

| AWS Wilayah | Domain |
| --- | --- |
| Eropa (Frankfurt) | `*.vfs.cloud9.eu-central-1.amazonaws.com`<br>`vfs.cloud9.eu-central-1.amazonaws.com` |
| Eropa (Irlandia) | `*.vfs.cloud9.eu-west-1.amazonaws.com`<br>`vfs.cloud9.eu-west-1.amazonaws.com` |
| Eropa (London) | `*.vfs.cloud9.eu-west-2.amazonaws.com`<br>`vfs.cloud9.eu-west-2.amazonaws.com` |
| Eropa (Milan) | `*.vfs.cloud9.eu-south-1.amazonaws.com`<br>`vfs.cloud9.eu-south-1.amazonaws.com` |
| Eropa (Paris) | `*.vfs.cloud9.eu-west-3.amazonaws.com`<br>`vfs.cloud9.eu-west-3.amazonaws.com` |
| Eropa (Stockholm) | `*.vfs.cloud9.eu-north-1.amazonaws.com`<br>`vfs.cloud9.eu-north-1.amazonaws.com` |

| AWS Wilayah | Domain |
| --- | --- |
| Timur Tengah (Bahrain) | `*.vfs.cloud9.me-south-1.amazonaws.com`<br><br>`vfs.cloud9.me-south-1.amazonaws.com` |
| Amerika Selatan (Sao Paulo) | `*.vfs.cloud9.sa-east-1.amazonaws.com`<br><br>`vfs.cloud9.sa-east-1.amazonaws.com` |

## Tab pratinjau aplikasi menampilkan kesalahan atau kosong

Masalah: Pada bilah menu di IDE, bila Anda memilih Pratinjau, Pratinjau Aplikasi Berjalan atau Alat, Pratinjau, Preview Aplikasi Berjalan untuk mencoba menampilkan aplikasi Anda pada tab pratinjau di IDE, tab menampilkan kesalahan, atau tab kosong.

Kemungkinan penyebabnya:

- Aplikasi Anda tidak berjalan di IDE.
- Aplikasi Anda tidak berjalan menggunakan HTTP.
- Aplikasi Anda berjalan di lebih dari satu port.
- Aplikasi Anda berjalan melalui port selain `8080`, `8081`, atau `8082`.
- Aplikasi Anda berjalan dengan IP selain `127.0.0.1`, `localhost`, atau `0.0.0.0`.
- Port (`8080`,`8081`, atau`8082`) tidak ditentukan dalam URL pada tab pratinjau.
- Jaringan Anda memblokir lalu lintas masuk ke port`8080`,`8081`, atau`8082`.
- Anda mencoba untuk pergi ke alamat yang berisi IP dari`127.0.0.1`,`localhost`, atau`0.0.0.0`. Secara default, AWS Cloud9 IDE mencoba untuk pergi ke komputer lokal Anda. Itu tidak mencoba untuk pergi ke instance atau server Anda sendiri yang terhubung ke lingkungan.

Solusi yang direkomendasikan:

- Pastikan bahwa aplikasi berjalan di IDE.

- Pastikan bahwa aplikasi berjalan menggunakan HTTP. Untuk contoh di Node.js dan Python, lihat Menjalankan aplikasi.
- Pastikan bahwa aplikasi berjalan di atas hanya satu port. Untuk contoh di Node.js dan Python, lihat Menjalankan aplikasi.
- Pastikan bahwa aplikasi berjalan di atas port `8080`, `8081`, atau `8082`. Untuk contoh di Node.js dan Python, lihat Menjalankan aplikasi.
- Pastikan bahwa aplikasi berjalan dengan IP `127.0.0.1`, `localhost`, atau `0.0.0.0`. Untuk contoh di Node.js dan Python, lihat Menjalankan aplikasi.
- Menambahkan `:8080`, `:8081`, atau `:8082` ke URL pada tab pratinjau.
- Pastikan bahwa jaringan Anda mengizinkan lalu lintas masuk melalui port `8080`, `8081`, atau `8082`. Jika Anda tidak dapat membuat perubahan pada jaringan Anda, lihat administrator jaringan Anda.
- Jika Anda mencoba untuk pergi ke alamat yang berisi IP dari`127.0.0.1`,, atau `localhost0.0.0.0`, coba pergi ke alamat berikut sebagai gantinya:`https://12a34567b8cd9012345ef67abcd890e1.vfs.cloud9.us-east-2.amazonaws.com/`. Di alamat ini, `12a34567b8cd9012345ef67abcd890e1` adalah ID yang AWS Cloud9 diberikan ke lingkungan. `us-east-2`adalah ID Wilayah AWS untuk lingkungan. Anda juga dapat mencoba pergi ke alamat ini di luar IDE. Namun, ini hanya berfungsi ketika IDE untuk lingkungan terbuka dan aplikasi berjalan di browser web yang sama.
- Setelah Anda yakin bahwa semua kondisi sebelumnya terpenuhi, coba hentikan aplikasi dan kemudian mulai lagi.
- Jika Anda menghentikan aplikasi dan kemudian memulainya lagi, coba memilih Pratinjau, Pratinjau Aplikasi Berjalan atau Alat, Pratinjau, Pratinjau Aplikasi Berjalan pada bilah menu lagi. Atau coba memilih tombol Refresh (panah melingkar) pada tab pratinjau aplikasi yang sesuai, jika tab sudah terlihat.

## Tidak dapat melihat pratinjau konten web di IDE karena koneksi ke situs tidak aman

Masalah: Saat Anda mencoba mengakses konten web seperti WordPress situs yang dihosting di AWS Cloud9 EC2 lingkungan, jendela pratinjau IDE tidak dapat menampilkannya.

Kemungkinan penyebabnya: Secara default, semua halaman web yang Anda akses di tab pratinjau aplikasi AWS Cloud9 IDE secara otomatis menggunakan protokol HTTPS. Jika URI halaman memiliki fitur `http` protokol yang tidak aman, ia secara otomatis digantikan oleh `https`. Dan Anda tidak dapat mengakses konten yang tidak aman dengan mengubah `https` kembali ke `http` secara manual.

Solusi yang disarankan: Hapus skrip atau konten HTTP yang tidak aman dari situs web yang Anda coba pratinjau di IDE. Ikuti petunjuk untuk server web atau sistem manajemen konten Anda untuk panduan penerapan HTTPS.

## Pratinjau file mengembalikan kesalahan 499

Masalah: Saat Anda mencoba menggunakan AWS Cloud9 IDE untuk melihat pratinjau file yang berisi `<script>` elemen yang berisi `src` atribut dan dengan `type` atribut disetel ke`module`, kesalahan 499 terjadi dan skrip tidak berjalan seperti yang diharapkan.

Penyebab: Permintaan pengambilan pratinjau file di AWS Cloud9 IDE mengharuskan cookie dikirim oleh browser web untuk mengautentikasi. Secara default, browser web mengirim cookie untuk permintaan skrip reguler. Mereka tidak mengirim cookie untuk permintaan skrip modul kecuali Anda menambahkan `crossorigin` atribut.

Solusi: Menambahkan atribut `crossorigin` ke elemen `<script>`. Misalnya, `<script type="module" src="index.js" crossorigin></script>`. Kemudian, simpan file yang diubah, dan coba pratinjau lagi.

# Kinerja

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan kinerja.

## AWS Cloud9 Pembekuan IDE untuk waktu yang signifikan

Masalah: Selama start-up, dan saat melakukan penyegaran, terminal AWS Cloud9 IDE membeku untuk waktu yang signifikan dan menjadi tidak dapat digunakan.

Penyebab: Anda mungkin memiliki sejumlah besar file di lingkungan Anda yang sedang ditonton secara rekursif oleh modul menonton file. AWS Cloud9

Solusi yang disarankan: Anda dapat mengurangi kedalaman menonton file (nilai minimum adalah 1) dan mempertimbangkan untuk menambahkan folder atau folder besar yang tidak terkait dengan kode sumber (build output/artefak, paket pihak ke-3) ke pola yang diabaikan. Untuk melakukan ini, navigasikan ke Preferences **>** User Settings **>** File Watching. Ketahuilah bahwa ini akan menyebabkan CodeLenses AWS Toolkit tidak berfungsi dengan benar.

Solusi lain yang mungkin adalah mempertimbangkan mengabaikan file besar dan folder yang tidak terkait dengan kode sumber dengan mengurangi jumlah maksimum file untuk dicari. Untuk

melakukan ini, navigasikan ke Preferences > Project Settings > Find in Files. Ketahuilah bahwa ini akan menyebabkan folder yang diabaikan tidak muncul dalam pencarian file.

## Peringatan konsol: "Beralih ke mesin penyelesaian kode minimal..."

Masalah: Saat bekerja di AWS Cloud9 konsol (misalnya, saat membuka IDE atau menyegarkan halaman web IDE), Anda melihat pesan ini: “Satu atau beberapa sesi atau kolaborator aktif di lingkungan ini. Beralih ke mesin penyelesaian kode minimal untuk menghemat memori." Dalam korelasi dengan olahpesan ini, perilaku penyelesaian kode mungkin lambat atau terputus-putus.

Penyebab: Menjalankan mesin penyelesaian kode mengambil memori dan CPU siklus dari lingkungan. Selain itu, mesin penyelesaian kode terpisah diperlukan untuk setiap kolaborator dan setiap sesi tambahan. Untuk menghindari penggunaan terlalu banyak sumber daya, terutama pada ukuran instance kecil seperti t2.nano and t2.micro, AWS Cloud9 beralih ke mesin penyelesaian kode minimal.

Solusi yang disarankan: Jika Anda berencana untuk sering berkolaborasi dan untuk jangka waktu yang lama, pilih EC2 instans Amazon yang lebih besar saat membuat EC2 lingkungan Anda. Atau, sebagai alternatif, hubungkan lingkungan SSH Anda ke instans dengan kapasitas lebih besar.

> **Note**
>
> Memilih EC2 instans Amazon yang lebih besar dapat Akun AWS menyebabkan Anda dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat EC2 Harga Amazon.

## Peringatan IDE: "Lingkungan ini kehabisan memori" atau "Lingkungan ini memiliki beban CPU yang tinggi"

Masalah: Sementara IDE berjalan, Anda melihat olahpesan yang berisi frase "lingkungan ini kehabisan memori" atau "lingkungan ini memiliki beban CPU yang tinggi."

Penyebab: IDE mungkin tidak memiliki sumber daya komputasi yang cukup untuk terus berjalan tanpa penundaan atau hang.

Solusi yang direkomendasikan:

- Hentikan satu atau lebih proses yang berjalan untuk membebaskan memori yang tersedia. Untuk melakukan ini, pada bar menu di IDE untuk lingkungan, memilih Alat, Daftar Proses. Untuk setiap proses yang ingin Anda hentikan, memilih prosesnya, lalu memilih Bunuh Paksa.

- Buat file swap di lingkungan. file swap adalah file di lingkungan yang dapat digunakan sistem operasi sebagai memori virtual.

  Untuk mengonfirmasi bahwa lingkungan saat ini menggunakan memori swap, jalankan **`top`**perintah dalam sesi terminal di lingkungan. Jika memori swap sedang digunakan, output menampilkan statistik memori `Swap` non-zero (misalnya, `Swap: 499996k total, 1280k used, 498716 free, 110672k cached`). Untuk berhenti menampilkan informasi memori waktu nyata, tekan `Ctrl + C`.

  Untuk membuat file swap, jalankan perintah seperti berikut di lingkungan.

  ```
  sudo fallocate --length 512MB /var/swapfile && sudo chmod 600 /var/swapfile && sudo
   mkswap /var/swapfile && echo '/var/swapfile swap swap defaults 0 0' | sudo tee -a /
  etc/fstab > /dev/null
  ```

  Perintah sebelumnya melakukan hal berikut:

  1. Membuat file 512 MB yang diberi nama `swapfile` dalam `/var` direktori.
  2. Mengubah izin akses untuk file `swapfile` untuk baca-tulis untuk pemilik saja.
  3. Menyiapkan file `swapfile` sebagai file swap.
  4. Menulis informasi ke`/etc/fstab file`. Ini membuat file swap ini tersedia setiap kali sistem reboot.

  Setelah Anda menjalankan perintah sebelumnya, untuk membuat file swap ini segera tersedia, jalankan perintah berikut.

  ```
  sudo swapon /var/swapfile
  ```

- Memindahkan atau mengubah ukuran lingkungan ke instans atau server dengan lebih sumber daya komputasi. Untuk memindahkan atau mengubah ukuran EC2 instans Amazon, lihat. Memindahkan AWS Cloud9 IDE dari volume Amazon EBS Untuk jenis server atau instans lainnya, lihat dokumentasi instans atau server Anda.

## Tidak dapat mengunggah file di AWS Cloud9 IDE

Masalah: Pengguna tidak dapat mengunggah file besar di AWS Cloud9 IDE. Unggahan ini gagal.

Penyebab: AWS Cloud9 membatasi kecepatan unggah ke AWS Cloud9 IDE, dan akibatnya waktu permintaan unggahan file habis.

Solusi yang disarankan: Sebaiknya unggah file ke Amazon S3, lalu gunakan Amazon S3 untuk mengunduh file ke lingkungan dengan CLI di IDE. AWS Cloud9 Untuk informasi selengkapnya tentang mengunggah objek ke Amazon S3, lihat Mengunggah objek di Panduan Pengguna Amazon S3.

## Kecepatan unduh lambat di AWS Cloud9 IDE

Masalah: Pengguna berurusan dengan kecepatan unduh yang lambat ketika mencoba mengunduh file dari AWS Cloud9 IDE.

Penyebab: Saat Anda mengunduh file dari IDE ke sistem file lokal, kecepatan transfer akan dibatasi hingga kecepatan 0,1 megabit/detik.

Solusi yang disarankan: Untuk meningkatkan kecepatan mentransfer file, gunakan CLI di IDE AWS Cloud9 Anda untuk mengunggah file ke Amazon S3 dan kemudian gunakan Amazon S3 untuk mengunduh file dari sana.

## Tidak dapat melihat pratinjau konten web di IDE karena koneksi ke situs tidak aman

Masalah: Saat Anda mencoba mengakses konten web seperti WordPress situs yang dihosting di AWS Cloud9 EC2 lingkungan, jendela pratinjau IDE tidak dapat menampilkannya.

Kemungkinan penyebabnya: Secara default, semua halaman web yang Anda akses di tab pratinjau aplikasi AWS Cloud9 IDE secara otomatis menggunakan protokol HTTPS. Jika URI halaman memiliki fitur `http` protokol yang tidak aman, ia secara otomatis digantikan oleh `https`. Dan Anda tidak dapat mengakses konten yang tidak aman dengan mengubah `https` kembali ke `http` secara manual.

Solusi yang disarankan: Hapus skrip atau konten HTTP yang tidak aman dari situs web yang Anda coba pratinjau di IDE. Ikuti petunjuk untuk server web atau sistem manajemen konten Anda untuk panduan penerapan HTTPS.

(kembali ke atas)

# Aplikasi dan layanan pihak ketiga

Bagian berikut menguraikan masalah pemecahan masalah yang terkait dengan aplikasi dan layanan pihak ketiga.

# Tidak dapat berinteraksi dengan jendela terminal AWS Cloud9 karena kesalahan **`tmux`** sesi

Masalah: Saat Anda mencoba meluncurkan jendela terminal baru AWS Cloud9, antarmuka baris perintah yang diharapkan tidak tersedia. Tidak ada command prompt dan Anda tidak dapat memasukkan teks. Pesan kesalahan seperti `tmux: need UTF-8 locale (LC_CTYPE)` dan `invalid LC_ALL, LC_CTYPE or LANG` dikembalikan.

Kemungkinan penyebabnya: Terminal yang tidak responsif mungkin disebabkan oleh kesalahan tmux. AWS Cloud9 menggunakan utilitas tmux. Dengan cara ini, informasi yang ditampilkan di terminal tetap ada bahkan ketika halaman dimuat ulang atau Anda terhubung kembali ke lingkungan pengembangan Anda.

Dalam `tmux` sesi, apa yang ditampilkan di jendela terminal ditangani oleh klien. Klien berkomunikasi ke server yang dapat mengelola beberapa sesi. Server dan klien berkomunikasi melalui soket yang terletak di `tmp` folder. Jika `tmp` folder hilang dari lingkungan pengembangan Anda atau izin yang terlalu ketat diterapkan padanya, `tmux` sesi tidak dapat dijalankan. Jika ini terjadi, jendela terminal di IDE menjadi tidak responsif.

Solusi yang disarankan: Jika `tmux` kesalahan mencegah Anda berinteraksi dengan jendela terminal, gunakan cara alternatif untuk membuat `tmp` folder dengan izin yang tepat. Dengan begitu, `tmux` sesi bisa berjalan. Salah satu solusinya adalah mengekspor ke `LC_CTYPE` dalam `.bash_profile` atau di `.bashrc` file. Solusi lain yang disarankan adalah menggunakan AWS Systems Manager untuk mengatur konfigurasi manajemen host. Ini memungkinkan akses ke instance yang relevan melalui EC2 konsol Amazon.

Menyiapkan manajemen host

1. Pertama, di AWS Cloud9 konsol, temukan nama instance lingkungan Anda. Anda dapat melakukannya dengan memilih panel yang relevan di halaman Lingkungan Anda dan memilih Lihat detail. Di halaman Environment details, pilih Go to Instance. Di EC2 konsol Amazon, konfirmasikan nama instance yang perlu Anda akses.
2. Sekarang pergi ke AWS Systems Manager konsol, dan di panel navigasi, pilih Pengaturan Cepat.
3. Di halaman Quick Setup, pilih Create.
4. Untuk jenis Konfigurasi, buka Manajemen Host dan pilih Buat.
5. Untuk opsi konfigurasi Sesuaikan Manajemen Host, di bagian Target, pilih Manual.

6. Pilih EC2 instance yang ingin Anda akses dan kemudian pilih Buat.

Menghubungkan ke instance dan menjalankan perintah

> **Note**
>
> Langkah-langkah berikut adalah untuk EC2 konsol baru.

1. Di EC2 konsol Amazon, di panel navigasi, pilih Instans dan pilih instance yang ingin Anda sambungkan.
2. Pilih Hubungkan.

   Jika Connect tidak diaktifkan, Anda mungkin perlu memulai instans terlebih dahulu.
3. Di panel Connect to instance Anda, untuk metode Connection, pilih Session Manager lalu pilih Connect.
4. Di jendela sesi terminal yang muncul, masukkan perintah berikut. Perintah ini membuat `tmp` folder dengan izin yang tepat sehingga soket tmux tersedia.

   ```
   sudo mkdir /tmp
   sudo chmod 777 /tmp
   sudo rmdir /tmp/tmux-*
   ```

## Tidak dapat memuat IDE menggunakan versi sebelumnya Microsoft Edge penjelajah

Masalah: `HTTP403: FORBIDDEN` kesalahan dikembalikan saat mencoba memuat AWS Cloud9 IDE menggunakan Microsoft Edge peramban web.

Kemungkinan penyebabnya: AWS Cloud9 IDE tidak mendukung versi lama tertentu Microsoft Edge.

Solusi yang disarankan: Untuk memperbarui browser, pilih tombol elipsis (...) di Microsoft Edge bilah alat. Dari menu, pilih Pengaturan dan kemudian pilih  Tentang Microsoft Edge. Jika pembaruan diperlukan, pembaruan akan diunduh dan diinstal secara otomatis.

# Kesalahan **gdb** saat men-debug C++ memproyeksikan

Masalah: Kesalahan dilaporkan untuk debugger gdb ketika mencoba untuk debug proyek C++ di IDE.

Kemungkinan penyebab: Misalkan AWS Cloud9 lingkungan Anda menggunakan tipe EC2 instance tertentu (misalnya, `t3.small` ataum`5.large`). Kemudian, kesalahan debug mungkin terjadi saat Anda mencoba menjalankan dan men-debug C++ proyek menggunakan runner bawaan IDE. Kesalahan ini dapat terjadi karena versi gdb (GNU Project Debugger) yang telah diinstal sebelumnya untuk lingkungan Anda tidak bekerja pada platform prosesor tertentu. Anda mungkin melihat kode kesalahan berikut.

```
GDB server terminated with code 1
```

Solusi yang Direkomendasikan: Masalah dengan gdb yang tidak men-support platform prosesor tertentu telah diperbaiki dari versi 3.0 dan seterusnya. Copot pemasangan debugger versi lama dan tingkatkan ke versi yang lebih baru dari: gdb

1. Hapus versi debugger yang ada dengan menjalankan perintah berikut di terminal. AWS Cloud9

   ```
   sudo yum -y remove gdb
   ```

2. Ambil arsip untukgdb, buka kemasannya, dan kemudian arahkan ke direktori yang berisi file yang diekstrak dengan menjalankan perintah berikut.

   ```
   wget "http://ftp.gnu.org/gnu/gdb/gdb-8.3.tar.gz"
   tar xzf gdb-8.3.tar.gz
   cd gdb-8.3
   ```

3. Bangun debugger dengan menjalankan perintah berikut. Untuk melakukan ini, salin dan tempel teks berikut sebagai satu blok dan tekan Kembali untuk menjalankan`make`.

   ```
   ./configure --prefix=/usr \
               --with-system-readline \
               --with-python=/usr/bin/python3 &&
   make
   ```

4. Instal debugger.

   ```
   sudo make -C gdb install
   ```

5. Konfirmasikan bahwa versi debugger yang diperbarui telah diinstal.

```
gdb --version
```

## Masalah dengan PHP runner di AWS Cloud9

Masalah: Pengguna tidak dapat melihat output apa pun di terminal runner PHP CLI.

Penyebab: CLI runner perlu diatur ke PHP dan mode debugger perlu diaktifkan.

Solusi yang disarankan: Atur runner CLI ke PHP dan pastikan mode debugger diaktifkan.

## Kesalahan GLIBC terkait dengan Node.js

Masalah: Pengguna tidak dapat menjalankan Node.js dan mendapatkan kesalahan GLIBC. Contoh pesan kesalahan ini diuraikan di bawah ini:

```
node: /lib64/libm.so.6: version `GLIBC_2.27' not found (required by node)
node: /lib64/libc.so.6: version `GLIBC_2.28' not found (required by node)
```

Penyebab: Kemungkinan itu bisa menjadi masalah versi Node.js yang terkait dengan instance yang digunakan.

Solusi yang disarankan: Lihat Langkah 1: Pasang alat yang diperlukan bagian untuk informasi tentang cara menginstal Node.js untuk AWS Cloud9.

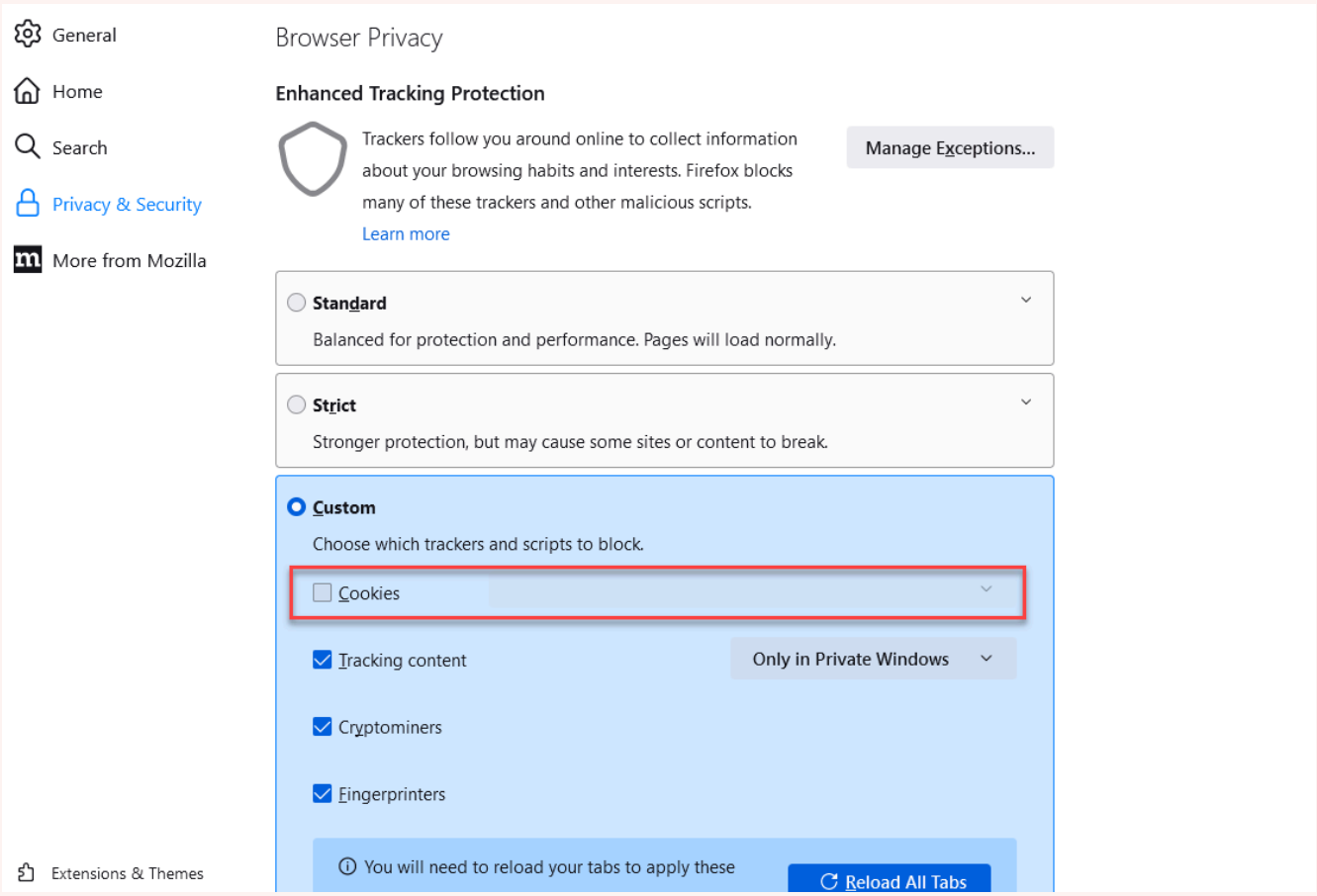
# Browser yang didukung untuk AWS Cloud9

Tabel berikut mencantumkan browser yang didukung untuk AWS Cloud9.

| Browser | Versi |
| --- | --- |
| Google Chrome | Tiga versi terakhir |
| Mozilla Firefox | Tiga versi terakhir |
| Microsoft Edge | Tiga versi terakhir |
| Apple Safari untuk macOS | Dua versi terakhir |

**Warning**

Jika Anda menggunakan Mozilla Firefox sebagai browser pilihan Anda dengan AWS Cloud9 IDE, ada pengaturan cookie pihak ketiga yang mencegah AWS Cloud9 tampilan web dan AWS Toolkit berfungsi dengan benar di browser. Sebagai solusi untuk masalah ini, Anda harus memastikan bahwa Anda belum memblokir Cookie di bagian Privasi & Keamanan pengaturan browser Anda, seperti yang ditampilkan pada gambar di bawah ini.


General
Home
Search
Privacy & Security
More from Mozilla
Extensions & Themes
Browser Privacy
Enhanced Tracking Protection
Trackers follow you around online to collect information about your browsing habits and interests. Firefox blocks many of these trackers and other malicious scripts.
Learn more
Manage Exceptions...
Standard
Balanced for protection and performance. Pages will load normally.
Strict
Stronger protection, but may cause some sites or content to break.
Custom
Choose which trackers and scripts to block.
Cookies
Tracking content
Only in Private Windows
Cryptominers
Fingerprinters
You will need to reload your tabs to apply these
Reload All Tabs

# Kuota layanan untuk AWS Cloud9

Tabel berikut mencantumkan kuota AWS Cloud9 dan AWS layanan terkait.

- AWS Cloud9 kuota layanan
- Kuota AWS Layanan Terkait

## AWS Cloud9 kuota

Tabel berikut menyediakan kuota default AWS Cloud9 untuk AWS akun. Kecuali dinyatakan lain, setiap batas berlaku untuk Wilayah tertentu. Anda dapat meminta peningkatan menggunakan Konsol AWS Manajemen atau AWS CLI. Untuk meminta peningkatan kuota, lihat Meminta kenaikan kuota pada Panduan Pengguna Service Quotas.

Peningkatan ini tidak diberikan segera, sehingga mungkin diperlukan beberapa hari agar peningkatan Anda menjadi efektif.

| Sumber Daya | Batas Default | Dapat Disesuaikan |
| --- | --- | --- |
| Jumlah maksimum lingkungan AWS Cloud9 EC2 pengembangan | • 100 per pengguna<br>• 200 per akun | Ya |
| Jumlah maksimum lingkungan SSH | • 100 per pengguna<br>• 200 per akun | Ya |
| Jumlah maksimum anggota dalam suatu lingkungan | Jumlah maksimum default anggota sama dengan memori instans untuk lingkungan tersebut dibagi 60 MB, dengan hasil yang dibulatkan ke bawah. Misalnya, sebuah instans dengan memori 1 GiB dapat memiliki maksimal 17 anggota (yaitu 1 GiB dibagi 60 MB, dibulatkan ke bawah). | Tidak[1] |

| Sumber Daya | Batas Default | Dapat Disesuaikan |
| --- | --- | --- |
| | Jika AWS Cloud9 tidak dapat menentukan memori dari sebuah instance, maka defaultnya maksimal 8 pengguna untuk setiap lingkungan yang terkait dengan instance tersebut.<br><br>Jumlah maksimum absolut anggota untuk suatu lingkunga n adalah 25. | |
| Maksimum ukuran file yang dapat diedit | 8 MB | Tidak |

[1] Anda dapat memindahkan lingkungan untuk mencoba meningkatkan jumlah anggota maksimum default. Namun, jumlah maksimum absolut anggota untuk suatu lingkungan tetap 25.

## AWS Cloud9 Kuota Unduh IDE

Saat Anda mengunduh file dari AWS Cloud9 IDE ke sistem file lokal, kecepatan transfer akan dibatasi hingga kecepatan 0,1 megabit/detik. Untuk meningkatkan kecepatan mentransfer file, gunakan CLI AWS Cloud9 di IDE untuk mengunggah file ke Amazon S3, lalu gunakan Amazon S3 untuk mengunduh file dari sana.

# Kuota AWS Layanan Terkait

| | |
| --- | --- |
| Jumlah maksimum volume Amazon Elastic Block Store (Amazon EBS) | 5.000<br><br>Untuk informasi selengkapnya, lihat titik akhir Amazon Elastic Block Store dan kuota di. Referensi Umum Amazon Web Services |
| Jumlah AWS CloudFormation tumpukan maksimum | 200 |

| | |
|---|---|
| | Untuk informasi selengkapnya, lihat Memahami CloudFormation kuota di Panduan AWS CloudFormation Pengguna. |
| EC2 Kuota Amazon | Lihat EC2titik akhir dan kuota Amazon di. Referensi Umum Amazon Web Services |

# Riwayat dokumen untuk Panduan AWS Cloud9 Pengguna

Topik ini berisi daftar perubahan signifikan pada Panduan Pengguna AWS Cloud9 . Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan ke umpan RSS.

Update terbaru

Tabel berikut menjelaskan perubahan penting pada Panduan Pengguna AWS Cloud9 setelah Maret 2019.

| Perubahan | Deskripsi | Tanggal |
| --- | --- | --- |
| AWS Cloud9 tidak lagi tersedia untuk pelanggan baru | AWS Cloud9 tidak lagi tersedia untuk pelanggan baru. Pelanggan yang sudah ada AWS Cloud9 dapat terus menggunakan layanan seperti biasa. Pelajari selengkapnya | Juli 25, 2024 |
| Support untuk Amazon Linux 2023 untuk AWS Cloud9 ditambahkan. | AWS Cloud9 sekarang mendukung Amazon Linux 2023. | 15 Desember 2023 |
| Pembaruan dibuat untuk tutorial Node.js. | Pembaruan telah dilakukan untuk tutorial Node.js terkait dengan dukungan untuk Amazon Linux 2 dan Node.js 18. | 23 Oktober 2023 |
| Memperbarui bagian tentang membuat VPC Amazon menggunakan dasbor Amazon VPC | Memperbarui bagian tentang membuat VPC Amazon menggunakan dasbor Amazon VPC. | Juli 27, 2023 |
| Bagian tentang bekerja dengan EventBridge Skema Amazon | Menambahkan bagian tentang bekerja dengan Amazon EventBridge Schemas | 15 Desember 2022 |

| | | |
|---|---|---|
| | menggunakan AWS Toolkit untuk. AWS Cloud9 | |
| CodeCatalyst Bagian Ditambahkan | Bagian tentang CodeCatalyst layanan Amazon baru telah ditambahkan. | Desember 2, 2022 |
| AWS IoT Konten Ditambahkan | Bagian tentang penggunaan AWS IoT ditambahkan. | 1 November 2022 |
| Ikhtisar layanan Amazon ECS untuk IDE AWS Cloud9 | Menambahkan ikhtisar dan panduan untuk fitur dan fungsi layanan Amazon ECS yang dapat diakses di IDE. AWS Cloud9 | 20 Oktober 2022 |
| Bekerja dengan AWS CDK lingkungan pengembangan AWS Cloud9 terintegrasi (IDE) | Bagian tentang bekerja dengan AWS CDK lingkungan pengembangan AWS Cloud9 terintegrasi (IDE) ditambahk an. | 5 Oktober 2022 |
| Konten Amazon ECR Ditambahkan | Bagian tentang penggunaan AWS Amazon ECR ditambahk an. | 4 Oktober 2022 |
| Validasi kepatuhan | Daftar program kepatuhan yang diperbarui AWS Cloud9 yang cakupannya. | 4 Maret 2022 |

| | | |
|---|---|---|
| Dukungan Java yang disempurnakan | Dukungan bahasa tambahan untuk meningkatkan pengalaman pengembangan Anda saat bekerja dengan Java. Fitur produktivitas utama termasuk penyelesaian kode, linting untuk kesalahan , tindakan spesifik konteks, dan opsi debugging seperti breakpoint dan stepping. | 18 Januari 2022 |
| Diperbarui AWSService RoleFor AWSCloud9 | Peran terkait layanan yang diperbarui untuk mendukung EC2 instance menggunakan License Manager. | 12 Januari 2022 |
| Dukungan dokumentasi Step Functions | Menambahkan konten yang menjelaskan penggunaa n Step Functions untuk membuat, mengedit, dan menjalankan state machine. | Desember 20, 2021 |
| AWS Systems Manager dukungan dokumentasi | Menambahkan konten yang menjelaskan dokumen otomatisasi Systems Manager. | Desember 20, 2021 |
| Panduan pengguna yang dibuat untuk Amazon Elastic Container Service Exec | Ini adalah ikhtisar dari Amazon ECS Exec. | 13 Desember 2021 |
| Panduan pengguna yang dibuat untuk layanan AWS IoTAWS Cloud9 IDE | Panduan pengguna ini mencakup bagaimana Anda dapat mulai menggunakan AWS IoT layanan untuk AWS Cloud9 IDE. | 22 November 2021 |

| | | |
|---|---|---|
| Support untuk AWS sumber daya | Menambahkan dukungan untuk mengakses jenis sumber daya bersama dengan opsi antarmuka untuk melihat sumber daya dan dokumentasi terkait. | November 5, 2021 |
| Ikhtisar layanan Amazon ECR untuk IDE AWS Cloud9 | Menambahkan ikhtisar dan panduan untuk fitur dan fungsi layanan Amazon ECR yang dapat diakses di IDE AWS Cloud9 | Oktober 14, 2021 |
| Dukungan App Runner | Menambahkan dukungan AWS App Runner untuk AWS Toolkit. | 30 September 2021 |
| AWS Cloud9 juga tersedia di Wilayah Afrika (Cape Town) dan Asia Pasifik (Osaka) | AWS Cloud9 sekarang juga tersedia di wilayah berikut: Afrika (Cape Town) dan Asia Pasifik (Osaka). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 1 September 2021 |
| CloudWatch Log dan Amazon S3 di Toolkit AWS | Menambahkan dukungan untuk CloudWatch Log ke AWS Toolkit untuk AWS Cloud9. Fitur baru untuk memungkinkan pengunggahan file saat ini ke bucket Amazon S3. | 16 Juli 2021 |

| | | |
|---|---|---|
| Titik akhir VPC untuk Amazon S3 | Menambahkan dukungan untuk mengonfigurasi VPC endpoint untuk Amazon S3 agar dependensi dapat diunduh. | 22 April 2021 |
| Kontrol sumber visual tersedia melalui panel Git | Sebagai developer, Anda dapat menggunakan panel Git untuk menjalankan perintah Git dalam antarmuka pengguna. | 1 Februari 2021 |
| Luncurkan instance lingkungan ke subnet pribadi | Menambahkan dukungan untuk EC2 instans yang diakses melalui Systems Manager untuk diluncurkan ke subnet pribadi. | 21 Januari 2021 |
| Integrasi untuk AWS Toolkit | Anda sekarang dapat menavigasi dan berinteraksi dengan Layanan AWS menggunakan AWS Toolkit melalui jendela AWS Explorer. | 11 Desember 2020 |
| AWS CloudFormation dan lingkungan tanpa masuknya EC2 | Dokumentasi yang diperluas tentang pembuatan EC2 lingkungan tanpa masuknya menggunakan AWS CloudFormation templat. | 29 Oktober 2020 |
| Lingkungan berbasis Amazon Linux 2 EC2 | Saat Anda membuat EC2 lingkungan di konsol, Anda dapat memilih Amazon Linux 2 AMI untuk EC2 instance. | 7 Oktober 2020 |

| | | |
|---|---|---|
| EC2 Instans tanpa masuk dengan Systems Manager | Menambahkan dukungan untuk mengakses EC2 instance pribadi dengan. AWS Systems Manager | 12 Agustus 2020 |
| Debugging lokal yang disempurnakan dari aplikasi AWS Tanpa Server | Menambahkan dukungan untuk fitur debugging lokal baru untuk Aplikasi Tanpa AWS Server. | 30 Juli 2020 |
| AWS Cloud9 juga tersedia di Wilayah Eropa (Milan) | AWS Cloud9 sekarang juga tersedia di Wilayah Eropa (Milan). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 29 Juli 2020 |
| Enkripsi Amazon EBS | Bagian yang menjelaskan cara mengenkripsi volume Amazon EBS untuk EC2 instans yang digunakan oleh lingkungan pengembangan. AWS Cloud9 | 3 Juli 2020 |

| | | |
|---|---|---|
| Menambahkan dukungan Wilayah ke AWS Cloud9 | AWS Cloud9 sekarang juga tersedia di Wilayah berikut: AS Barat (California N.), Asia Pasifik (Hong Kong), Eropa (Paris), Timur Tengah (Bahrain), dan Amerika Selatan (São Paulo). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 7 Mei 2020 |
| Keamanan | Bab keamanan ditambahkan ke Panduan AWS Cloud9 Pengguna. | 30 April 2020 |
| Tanda | Gunakan tag untuk membantu Anda mengontrol akses ke AWS Cloud9 sumber daya dan membantu Anda mengelola informasi penagihan. | 22 Januari 2020 |

| | | |
|---|---|---|
| Menambahkan dukungan Wilayah ke AWS Cloud9 | AWS Cloud9 sekarang juga tersedia di Wilayah berikut: Asia Pasifik (Mumbai), Asia Pasifik (Seoul), Asia Pasifik (Sydney), Kanada (Tengah), Eropa (London), dan Eropa (Stockholm). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 18 Desember 2019 |
| Diperbarui: Pemecahan Masalah, Tidak Dapat Membuka Lingkungan | Cookie pihak ke tiga tidak lagi diperlukan untuk membuka IDE. | 6 November 2019 |
| Ditambahkan: Pemecahan masalah, cookie pihak ketiga dinonaktifkan | Cookie pihak ketiga tidak lagi diperlukan untuk membuka IDE. Namun, mereka diperlukan untuk fitur Pratinjau Aplikasi atau Pratinjau File. Anda dapat menemukan informasi tentang ini di topik Pemecahan Masalah. | 6 November 2019 |
| Organisasi dokumen | Perubahan organisasi diterapkan ke panduan pengguna untuk membantu navigasi, terutama untuk pengguna pertama kali. | 15 Agustus 2019 |

| | | |
|---|---|---|
| AWS Cloud9 juga tersedia di Wilayah Eropa (Frankfurt) | AWS Cloud9 sekarang juga tersedia di Wilayah Eropa (Frankfurt). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 15 Mei 2019 |
| Sampel LAMP ditambahkan | Menambahkan sampel baru yang menunjukkan cara menggunakan AWS Cloud9 LAMP (Linux, Apache HTTP Server, MySQL, dan PHP). Untuk informasi selengkap nya, lihat Sampel LAMP untuk AWS Cloud9. | 10 Mei 2019 |
| WordPress sampel ditambahk an | Menambahkan sampel baru yang menunjukkan bagaimana menggunakan AWS Cloud9 dengan WordPress. Untuk informasi selengkapnya, lihat WordPress Sampel untuk AWS Cloud9. | 19 April 2019 |
| AWS Cloud9 juga tersedia di Wilayah Asia Pasifik (Tokyo) | AWS Cloud9 sekarang juga tersedia di Wilayah Asia Pasifik (Tokyo). Untuk informasi selengkapnya tentang titik akhir layanan dan kuota layanan yang terkait dengan ini dan AWS Wilayah lainnya, lihat AWS Cloud9di. Referensi Umum Amazon Web Services | 4 April 2019 |

| | | |
|---|---|---|
| Informasi tentang dukungan untuk Ubuntu Server di EC2 lingkungan ditambahkan | Petunjuk untuk menggunakan AWS Cloud9 konsol untuk membuat lingkungan AWS Cloud9 EC2 pengembangan yang terhubung ke Server Ubuntu ditambahkan. Untuk informasi selengkapnya, lihat Membuat EC2 Lingkungan.<br><br>Perhatikan bahwa saat ini Anda tidak dapat menggunakan kode untuk membuat lingkungan AWS Cloud9 EC2 pengembangan yang terhubung ke Server Ubuntu AWS CLI AWS CloudFormation, misalnya dengan menggunakan AWS SDKs,, Alat untuk Windows PowerShell, atau AWS Cloud9 API. Dukungan untuk metode ini diharapkan di masa mendatang. | 2 April 2019 |

Pembaruan sebelumnya

Tabel berikut menjelaskan perubahan penting pada Panduan Pengguna AWS Cloud9 sebelum April 2019.

| Ubah | Deskripsi | Tanggal Berubah |
|---|---|---|
| Instruksi memulai ditambahkan untuk siswa, pendidik, dan perusahaan | Instruksi untuk memulai AWS Cloud9 telah diperluas untuk mencakup langkah-langkah untuk siswa, pendidik, dan perusahaan. Untuk informasi | 7 Februari 2019 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| | selengkapnya, lihat Menyiapkan AWS Cloud9. | |
| AWS CloudTrail dukungan ditambahkan | AWS CloudTrail sekarang mendukung AWS Cloud9. Untuk informasi selengkapnya, lihat Pencatatan panggilan AWS Cloud9 API dengan AWS CloudTrail. | 21 Januari 2019 |
| VPCs Dukungan bersama ditambahkan | AWS Cloud9 sekarang mendukung dibagikan VPCs di Amazon VPC. Untuk informasi selengkapnya, lihat Persyaratan Amazon VPC untuk AWS Cloud9. | 7 Desember 2018 |
| AWS RoboMaker integrasi ditambahkan | AWS Cloud9 sekarang mendukung AWS RoboMaker, layanan yang memudahkan untuk mengembangkan, menguji, dan menyebarkan aplikasi robotika cerdas dalam skala besar. Untuk informasi selengkapnya, lihat Memulai AWS RoboMaker dan Mengembangkan dengan AWS Cloud9 di Panduan AWS RoboMaker Pengembang. | 26 November 2018 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Informasi tentang fitur produktivitas tambahan untuk proyek bahasa ditambahkan | AWS Cloud9 IDE sekarang menyediakan fitur produktiv itas tambahan untuk beberapa bahasa dalam konteks proyek bahasa. Untuk informasi selengkapnya, lihat TypeScrip t Dukungan dan fitur yang ditingkatkan. | 2 Oktober 2018 |
| Jendela Pergi ditambahkan; jendela Navigasi dan Perintah dihapus | Jendela Go ditambahkan ke AWS Cloud9 IDE untuk lingkungan yang dibuat pada atau setelah 2 Oktober 2018. Jendela baru ini menggantikan jendela Navigasi dan Perintah, yang keduanya dihapus dari IDE untuk lingkungan yang dibuat pada atau setelah 2 Oktober 2018. Untuk informasi selengkapnya, lihat Langkah 10: jendela Pergi di Tur IDE. | 2 Oktober 2018 |
| AWS CDK sampel ditambahk an | Menambahkan contoh baru yang menunjukkan cara menggunakan AWS Cloud9 AWS Cloud Development Kit (AWS CDK). Untuk informasi selengkapnya, lihat AWS CDK tutorial untuk AWS Cloud9. | 30 Agustus 2018 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Informasi tentang pembatasan alamat IP SSH secara otomatis ditambahkan ke EC2 lingkungan yang ditambahkan | Untuk lingkungan AWS Cloud9 EC2 pengembangan yang dibuat pada atau setelah 31 Juli 2018, AWS Cloud9 sekarang secara otomatis membatasi lalu lintas SSH yang masuk hanya ke rentang alamat IP yang AWS Cloud9 digunakan untuk terhubung melalui SSH. Untuk informasi selengkapnya, lihat Rentang alamat IP SSH masuk untuk AWS Cloud9. | 31 Juli 2018 |
| Sampel Docker ditambahkan | Menambahkan sampel baru yang menunjukkan cara menggunakan AWS Cloud9 dengan Docker. Untuk informasi selengkapnya, lihat Tutorial Docker untuk AWS Cloud9. | 19 Juni 2018 |
| Sampel ditambahkan untuk Java, .NET Core, dan TypeScript | Menambahkan sampel baru yang menunjukkan cara menggunakan AWS Cloud9 dengan Java, .NET Core, dan TypeScript. Untuk informasi lebih lanjut, lihat Tutorial Java untuk AWS Cloud9, Tutorial. NET untuk AWS Cloud9 dan TypeScript tutorial untuk AWS Cloud9. | 29 Mei 2018 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Daftar peramban yang didukung ditambahkan | Menambahkan informasi tentang browser yang didukung untuk AWS Cloud9. Untuk informasi selengkapnya, lihat Browser yang didukung untuk AWS Cloud9. | 23 Mei 2018 |
| Informasi pembatasan lalu lintas IP SSH ditambahkan | Menambahkan informasi tentang cara membatasi lalu lintas masuk hanya rentang alamat IP yang AWS Cloud9 digunakan untuk terhubung ke host melalui SSH. Untuk informasi selengkapnya, lihat Rentang alamat IP SSH masuk untuk AWS Cloud9. | 19 April 2018 |
| Pemecah masalah ditambahkan untuk melihat pratinjau aplikasi dan berbagi aplikasi yang sedang berjalan | Menambahkan pemecah masalah baru untuk melihat pratinjau aplikasi dan berbagi aplikasi yang sedang berjalan. Untuk informasi selengkapnya, lihat Tab pratinjau aplikasi menampilkan kesalahan atau kosong dan Tidak dapat menampilkan aplikasi yang sedang berjalan di luar IDE. | 19 April 2018 |
| Informasi Riwayat Revisi File ditambahkan | Menambahkan informasi tentang cara menggunakan panel Riwayat Revisi File di IDE. Untuk informasi selengkapnya, lihat Bekerja dengan Revisi File di IDE AWS Cloud9. | 19 April 2018 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Pemecah masalah ditambahkan untuk membuka lingkungan | Menambahkan pemecah masalah baru untuk membuka AWS Cloud9 lingkungan pengembangan. Untuk informasi selengkapnya, lihat Tidak dapat membuka lingkungan. | 19 Maret 2018 |
| Pemecah masalah ditambahkan untuk Installer AWS Cloud9 | Menambahkan pemecah masalah baru untuk Installer. AWS Cloud9 Untuk informasi selengkapnya, lihat AWS Cloud9 Pemasang hang atau gagal. | 19 Maret 2018 |
| AWS CodePipeline informasi ditambahkan | Menambahkan informasi tentang cara menggunakan AWS Cloud9 dengan AWS CodePipeline. Untuk informasi selengkapnya, lihat Bekerja dengan AWS CodePipeline di AWS Cloud9 IDE. | 13 Februari 2018 |
| AWS CloudShell informasi ditambahkan | Menambahkan informasi tentang cara menggunakan AWS Cloud9 dengan AWS CloudShell. Untuk informasi selengkapnya, lihat AWS CLI dan tutorial aws-shell untuk AWS Cloud9. | 19 Januari 2018 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Ketersediaan dokumentasi GitHub ditambahkan | Panduan ini sekarang tersedia di GitHub. Anda juga dapat menggunakan GitHub untuk mengirimkan umpan balik dan mengubah permintaan untuk konten panduan ini. Untuk informasi selengkapnya, pilih GitHub ikon Edit on di bilah navigasi panduan, atau lihat repositori awsdocs/aws-cloud9-user-guide di situs web. GitHub | 10 Januari 2018 |
| Ketersediaan format Kindle | Panduan ini kini tersedia dalam format Amazon Kindle. Untuk informasi selengkapnya, pilih ikon Open Kindle di bilah navigasi panduan. | 2 Januari 2018 |
| Informasi Amazon Lightsail ditambahkan | Menambahkan informasi tentang cara menggunakan AWS Cloud9 Amazon Lightsail. Untuk informasi selengkapnya, lihat Bekerja dengan instans Amazon Lightsail di IDE AWS Cloud9. | 19 Desember 2017 |
| Menambahkan deskripsi pengaturan lingkungan untuk AWS | Menambahkan deskripsi AWS pengaturan khusus untuk lingkungan AWS Cloud9 pengembangan. Untuk informasi selengkapnya, lihat Bekerja dengan pengaturan AWS proyek dan pengguna di AWS Cloud9 IDE. | 7 Desember 2017 |

| Ubah | Deskripsi | Tanggal Berubah |
| --- | --- | --- |
| Petunjuk memulai ditambahkan untuk pengguna root AWS akun dan langkah-langkah penyiapan lanjutan untuk tim | Menambahkan langkah-langkah pengaturan untuk menggunakan AWS Cloud9 dengan pengguna root AWS akun. Menambahkan langkah-langkah pengaturan lanjutan untuk digunakan AWS Cloud9 dengan tim. Untuk informasi selengkapnya, lihat Menyiapkan AWS Cloud9. | 5 Desember 2017 |
| Cakupan diperluas untuk persyaratan lingkungan | Cakupan persyaratan yang diperluas untuk EC2 instans Amazon atau server Anda sendiri untuk terhubung ke lingkungan pengembangan AWS Cloud9 SSH. Untuk informasi selengkapnya, lihat Persyaratan host lingkungan SSH. | 4 Desember 2017 |
| Rilis dokumentasi awal | Ini adalah rilis awal Panduan Pengguna AWS Cloud9 . | 30 November 2017 |