



Panduan Developerr

# Amazon CloudSearch



Versi API 2013-01-01

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# Amazon CloudSearch: Panduan Developerr

Copyright © 2024 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 antara para pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon adalah milik dari pemiliknya masing-masing, yang mungkin berafiliasi atau tidak berafiliasi dengan, terkait, atau disponsori oleh Amazon.

---

# Table of Contents

Apa itu Amazon CloudSearch? .....	1
Apakah Anda baru mengenal Amazon CloudSearch? .....	2
Cara Kerja Penelusuran .....	3
Pengeindeksan .....	3
Faset .....	4
Pengolahan Teks .....	5
Menyortir Hasil .....	5
Permintaan Pencarian .....	5
Penskalaan Otomatis .....	6
Penskalaan untuk Data .....	7
Penskalaan untuk Lalu Lintas .....	8
Mengakses Amazon CloudSearch .....	9
Wilayah dan titik akhir .....	9
Menandatangani Permintaan .....	10
Memulai .....	11
Sebelum Anda Memulai .....	11
Langkah 1: Buat Domain Pencarian .....	12
Langkah 2: Unggah Data untuk Pengeindeksan .....	14
Langkah 3: Cari Domain Anda .....	15
Mencari dengan Search Tester .....	16
Mengirimkan Permintaan Pencarian dari Browser Web .....	17
Mencari Bidang Numerik .....	18
Menyortir Hasil Pencarian .....	19
Mendapatkan Informasi Facet .....	20
Mendapatkan Sorotan Pencarian .....	22
Langkah 4: Hapus Domain Film Anda .....	23
Migrasi ke API 2013-01-01 .....	25
Membuat 2013-01-01 Domain .....	25
Mengkonfigurasi 2013-01-01 Domain .....	25
Tindakan dan Opsi Layanan Konfigurasi Baru .....	28
Tindakan dan Opsi Layanan Konfigurasi Usang .....	29
Mengunggah Data ke Domain 2013-01-01 .....	29
Mencari 2013-01-01 Domain .....	30
Parameter dan Opsi Pencarian Baru .....	32

Parameter dan Opsi Pencarian Usang .....	33
Batas yang Diperbarui .....	34
Membuat dan Mengelola Domain Pencarian .....	36
Membuat Domain Pencarian .....	36
Membuat Domain Menggunakan Konsol .....	38
Membuat Domain Menggunakan AWS CLI .....	39
Membuat Domain Menggunakan AWS SDK .....	40
Mengkonfigurasi Akses .....	40
Menulis Kebijakan Akses untuk Amazon CloudSearch .....	41
Contoh CloudSearch Kebijakan Amazon .....	45
Mengkonfigurasi Akses untuk Amazon CloudSearch Menggunakan Konsol .....	51
Mengkonfigurasi Akses untuk Amazon CloudSearch dengan AWS CLI .....	52
Mengonfigurasi Akses ke Endpoint Domain Menggunakan AWS SDK .....	53
Mengkonfigurasi Opsi Penskalaan .....	54
Memilih Opsi Penskalaan .....	56
Mengkonfigurasi Opsi Penskalaan melalui Konsol .....	56
Mengkonfigurasi Opsi Penskalaan melalui AWS CLI .....	57
Mengkonfigurasi Opsi Penskalaan melalui AWS SDK .....	58
Mengkonfigurasi Opsi Ketersediaan .....	58
Mengkonfigurasi Opsi Ketersediaan melalui Konsol .....	60
Mengkonfigurasi Opsi Ketersediaan Menggunakan AWS CLI .....	60
Mengkonfigurasi Opsi Ketersediaan melalui AWS SDK .....	61
Mengkonfigurasi Opsi Endpoint Domain .....	61
Mengonfigurasi Opsi Titik Akhir Domain Menggunakan Konsol Amazon CloudSearch .....	61
Mengkonfigurasi Opsi Endpoint Domain Menggunakan AWS CLI .....	62
Mengonfigurasi Opsi Titik Akhir Domain Menggunakan AWS SDK .....	62
Memantau Domain Pencarian .....	62
Mendapatkan Informasi Domain .....	62
Memantau Domain dengan Amazon CloudWatch .....	68
Panggilan API Konfigurasi Pencatatan .....	71
Melacak CloudSearch Penggunaan dan Biaya Amazon Anda .....	74
Menghapus Domain .....	75
Menghapus Domain Menggunakan Konsol .....	75
Menghapus Domain Menggunakan AWS CLI .....	76
Menghapus Domain Menggunakan AWS SDK .....	76
Menandai Domain Amazon CloudSearch .....	76

Bekerja dengan Tag (Konsol) .....	77
Mengontrol Bagaimana Data Diindeks .....	79
Mempersiapkan Data Anda .....	79
Memetakan Data Dokumen ke Bidang Indeks .....	80
Membuat Batch Dokumen .....	80
Mengkonfigurasi Bidang Indeks .....	86
Mengkonfigurasi Bidang Indeks Individu .....	89
Mengkonfigurasi Bidang Indeks Menggunakan Konsol .....	90
Mengkonfigurasi Bidang Indeks Menggunakan AWS SDK .....	91
Menggunakan Bidang Dinamis .....	91
Mengkonfigurasi Bidang Dinamis .....	91
Mengabaikan Bidang Dokumen yang Tidak Dikenali .....	93
Mencari Bidang Dinamis .....	94
Mengkonfigurasi Skema Analisis .....	95
Memberangkai .....	96
Stopwords .....	98
Sinonim .....	99
Mengkonfigurasi Skema Analisis Menggunakan Konsol .....	100
Mengkonfigurasi Skema Analisis Menggunakan AWS CLI .....	101
Mengkonfigurasi Skema Analisis Menggunakan AWS SDK .....	103
Pengeindeksan Bigram untuk China, Jepang, dan Korea .....	103
Menyesuaikan Tokenisasi Jepang .....	104
Pengolahan Teks .....	109
Bahasa yang Didukung .....	110
Pengaturan Khusus Bahasa .....	111
Mengunggah dan Mengindeks Data .....	128
Mengunggah Data .....	128
Mengirimkan Permintaan Unggah Dokumen .....	130
Unggahan Massal .....	131
Mengunggah Data Menggunakan Konsol .....	132
Mengunggah Data Menggunakan AWS CLI .....	134
Memposting Dokumen ke Endpoint Layanan Dokumen melalui HTTP .....	135
Pengeindeksan Data Dokumen .....	135
Mengindeks Dokumen Menggunakan Konsol .....	136
Pengeindeksan Dokumen Menggunakan AWS CLI .....	137
Mengindeks Dokumen dengan AWS SDK .....	137

Mencari Data Anda .....	138
Mengirimkan Permintaan Pencarian .....	139
Mencari dengan Search Tester .....	142
Membangun Kueri Senyawa .....	143
Mencari Teks di Amazon CloudSearch .....	145
Mencari Persyaratan Individu .....	146
Mencari Frase .....	149
Mencari String Literal .....	150
Mencari Prefiks .....	150
Mencari Angka .....	152
Mencari Tanggal dan Waktu .....	153
Mencari Rentang Nilai .....	153
Mencari Rentang Tanggal .....	154
Mencari Rentang Lokasi .....	154
Mencari Rentang Teks .....	155
Pencarian dan Peringkat Hasil berdasarkan Lokasi Geografis .....	155
Mencari di dalam suatu area .....	155
Menyortir Hasil berdasarkan Jarak .....	156
Mencari DynamoDB Data .....	157
Mengkonfigurasi Domain untuk Mencari DynamoDB Data .....	158
Mengunggah Data dari DynamoDB .....	159
Menyinkronkan Domain Pencarian dengan Tabel DynamoDB .....	161
Memfilter Dokumen yang Cocokan .....	163
Tuning Permintaan Pencarian .....	163
Menganalisis Latensi Kueri .....	163
Meminta Informasi Lebih Lanjut .....	166
Mengambil Data dari Bidang Indeks .....	166
Mendapatkan Statistik untuk Bidang Numerik .....	168
Mendapatkan dan Menggunakan Informasi Facet .....	169
Mendapatkan Informasi Facet .....	169
Menggunakan Informasi Facet .....	171
Menyoroti Hits Pencarian .....	178
Mendapatkan Saran .....	180
Mengkonfigurasi Suggesters .....	180
Mengambil Saran .....	184
Mengontrol Hasil Pencarian .....	187

Menyortir Hasil .....	187
Menggunakan Pembobotan Bidang Relatif untuk Menyesuaikan Relevansi Teks .....	189
Mengkonfigurasi Ekspresi .....	190
Menulis Ekspresi .....	190
Mendefinisikan Ekspresi dalam Permintaan Pencarian .....	192
Mengkonfigurasi Ekspresi yang Dapat Digunakan Kembali .....	193
Membandingkan Ekspresi .....	195
Mendapatkan Hasil sebagai XHTML .....	197
Hasil Paginating .....	198
Paging Dalam Melampaui 10.000 Hits .....	198
Integrasi dengan API Gateway .....	201
Prasyarat .....	201
Membuat dan Mengonfigurasi API (Konsol) .....	204
Menguji API (Konsol) .....	205
Penanganan Kesalahan .....	207
Jenis Kesalahan di Amazon CloudSearch .....	207
Mencoba Kembali Permintaan di Amazon CloudSearch .....	208
Amazon CloudSearch API Referensi .....	210
Referensi API Konfigurasi .....	210
Mengirimkan Permintaan Konfigurasi .....	210
Tindakan .....	213
Jenis Data .....	266
Parameter Umum .....	312
Kesalahan Umum .....	315
Referensi API Layanan Dokumen .....	317
dokumen/batch .....	317
Cari Referensi API .....	329
Pencarian .....	329
Mengirimkan Permintaan Saran .....	353
Sarankan .....	354
Kesalahan Layanan Pencarian .....	356
Memecahkan masalah .....	358
Mengunggah Dokumen .....	358
Menghapus Semua Dokumen .....	359
Domain Tidak Menurun Setelah Menghapus Dokumen .....	360
Latensi Pembaruan Dokumen .....	360

Sejumlah Besar Kesalahan 5xx Saat Mengunggah Dokumen .....	360
Cari Latensi dan Timeout .....	360
Cari Latensi dan Timeout .....	361
Peningkatan Tiba-tiba dalam Kesalahan 5xx saat Mencari .....	361
Kegagalan Pengindeksan setelah Memperbarui Opsi Pengindeksan .....	362
Kesalahan Domain Tidak Ditemukan .....	362
Jumlah Dokumen yang Dapat D dicari Tidak Dikembalikan .....	362
Konfigurasi Kebijakan Akses Layanan Tidak Berfungsi .....	363
Kebijakan Akses Layanan Pencarian dan Dokumen Tidak Berfungsi .....	363
Kesalahan Izin CloudSearch Konsol Amazon .....	364
Menggunakan Wildcard untuk Mencari Bidang Teks Tidak Menghasilkan Hasil yang Diharapkan .....	364
Hasil yang Tidak Konsisten Saat Menggunakan Kursor untuk Deep Paging .....	365
Kesalahan Sertifikat Saat Menggunakan SDK .....	365
Batas .....	367
Sumber daya .....	372
Riwayat Dokumen .....	373
Glosarium AWS .....	381
.....	ccclxxxii



# Apa itu Amazon CloudSearch?

Amazon CloudSearch adalah layanan yang dikelola sepenuhnya di cloud yang memudahkan pengaturan, pengelolaan, dan skala solusi pencarian untuk situs web atau aplikasi Anda.

Dengan Amazon CloudSearch Anda dapat mencari koleksi besar data seperti halaman web, file dokumen, posting forum, atau informasi produk. Anda dapat dengan cepat menambahkan kemampuan pencarian tanpa harus menjadi ahli pencarian atau khawatir tentang penyediaan, persiapan, dan pemeliharaan perangkat keras. Saat volume data dan lalu lintas Anda berfluktuasi, Amazon CloudSearch menskalakan untuk memenuhi kebutuhan Anda.

## Note

Dokumen ini menjelaskan Amazon CloudSearch 2013-01-01 API. [Jika Anda memiliki domain pencarian 2011-02-01 dan perlu mereferensikan dokumentasi lama, Anda dapat mengunduh PDF dari Panduan Pengembang 2011-02-01.](#)

Anda dapat menggunakan Amazon CloudSearch untuk mengindeks dan mencari data terstruktur dan teks biasa. CloudSearch Fitur Amazon:

- Pencarian teks lengkap dengan pemrosesan teks khusus bahasa
- Pencarian Boolean
- Pencarian awalan
- Rentang pencarian
- Meningkatkan jangka waktu
- Faceting
- Menyoroti
- Saran pelengkapan otomatis

Anda bisa mendapatkan hasil pencarian dalam JSON atau XHTML, mengurutkan dan memfilter hasil berdasarkan nilai bidang, dan mengurutkan hasil menurut abjad, numerik, atau menurut ekspresi kustom.

Untuk membangun solusi pencarian dengan Amazon CloudSearch, Anda mengambil langkah-langkah berikut:

- Buat dan konfigurasi domain pencarian. Domain penelusuran mencakup data yang dapat dicari dan contoh penelusuran yang menangani permintaan penelusuran Anda. Jika Anda memiliki beberapa koleksi data yang ingin dicari, Anda dapat membuat beberapa domain penelusuran.
- Unggah data yang ingin Anda cari ke domain Anda. Amazon CloudSearch mengindeks data Anda dan menyebarkan indeks pencarian ke satu atau beberapa instance penelusuran.
- Cari domain Anda. Anda mengirim permintaan pencarian ke titik akhir pencarian domain Anda sebagai permintaan GET HTTP/HTTPS.

## Topik

- [Apakah Anda baru mengenal Amazon CloudSearch?](#)
- [Cara Kerja Penelusuran](#)
- [Penskalaan Otomatis di Amazon CloudSearch](#)
- [Mengakses Amazon CloudSearch](#)

## Apakah Anda baru mengenal Amazon CloudSearch?

Untuk ikhtisar tingkat tinggi Amazon CloudSearch, sorotan layanan, dan informasi harga, lihat [halaman CloudSearch detail Amazon](#). Jika Anda siap untuk mulai menggunakan Amazon CloudSearch, Anda harus mulai dengan [Memulai dengan Amazon CloudSearch](#).

Anda dapat berinteraksi dengan Amazon CloudSearch melalui AWS SDK, atau AWS CLI. AWS Management Console Meskipun Anda juga dapat mengirimkan permintaan API langsung ke Amazon CloudSearch, SDK dan AWS CLI secara otomatis menandatangani permintaan Anda sesuai kebutuhan dan menyediakan alat terpusat untuk berinteraksi dengan CloudSearch domain Amazon bersama dengan layanan AWS lainnya. Untuk informasi tentang AWS SDK, lihat [Alat untuk Amazon Web Services](#). Untuk informasi tentang menginstal dan menggunakan AWS CLI, lihat [Panduan Pengguna AWS Command Line Interface](#).

Untuk informasi selengkapnya tentang mengonfigurasi dan mengelola domain penelusuran, memasukkan data ke Amazon CloudSearch, mengirimkan permintaan penelusuran, dan memproses tanggapan, lihat:

- [Mempersiapkan Data Anda](#)—cara memformat data Anda sehingga Anda dapat mengunggahnya ke CloudSearch domain Amazon untuk pengindeksan
- [configure indexing options](#)—cara mengonfigurasi opsi pengindeksan untuk domain Amazon CloudSearch

- [Mencari Data Anda dengan Amazon CloudSearch](#)—cara menggunakan bahasa CloudSearch kueri Amazon
- [Mengontrol Hasil Pencarian](#)—cara mengurutkan, memfilter, dan membuat halaman hasil pencarian

## Cara Kerja Penelusuran

Pengumpulan data yang ingin Anda cari (kadang-kadang disebut sebagai korpus Anda) dapat terdiri dari dokumen teks lengkap yang tidak terstruktur, dokumen semi-terstruktur seperti yang diformat dalam bahasa mark-up seperti XHTML, atau data terstruktur yang sesuai dengan model data yang ketat. Setiap item yang ingin Anda cari (seperti posting forum atau halaman web) direpresentasikan sebagai dokumen. Setiap dokumen memiliki ID unik dan satu atau beberapa bidang yang berisi data yang ingin Anda cari dan sertakan dalam hasil.

Untuk membuat data Anda dapat dicari, Anda merepresentasikannya sebagai kumpulan dokumen baik dalam JSON atau XHTML dan mengunggah batch ke domain pencarian Anda. Amazon CloudSearch kemudian menghasilkan indeks pencarian dari data dokumen Anda sesuai dengan opsi konfigurasi domain Anda. Anda mengirimkan kueri terhadap indeks ini untuk menemukan dokumen yang memenuhi kriteria pencarian tertentu.

Saat data Anda berubah, Anda mengirimkan pembaruan untuk menambah, mengubah, atau menghapus dokumen dari indeks Anda. Pembaruan diterapkan terus menerus sesuai urutan penerimaannya.

Untuk informasi tentang cara memformat data Anda, lihat [Mempersiapkan Data Anda](#).

## Pengindeksan di Amazon CloudSearch

Untuk membuat indeks pencarian dari data Anda, Amazon CloudSearch memerlukan informasi berikut:

- Bidang dokumen mana yang ingin Anda cari?
- Nilai bidang dokumen mana yang ingin Anda ambil dengan hasil pencarian?
- Bidang dokumen mana yang mewakili kategori yang ingin Anda gunakan untuk menyempurnakan dan memfilter hasil pencarian?
- Bagaimana seharusnya teks dalam bidang tertentu diproses?

Anda menentukan metadata ini dalam konfigurasi domain Anda dengan mengonfigurasi opsi pengindeksan. Anda menggunakan opsi pengindeksan untuk menentukan bidang yang disertakan dalam indeks pencarian dan mengontrol bagaimana Anda dapat menggunakan bidang tersebut.

Anda harus mengonfigurasi bidang indeks yang sesuai untuk setiap bidang dokumen yang terjadi dalam data Anda—ada one-to-one pemetaan antara bidang dokumen dan bidang dalam indeks Amazon Anda. CloudSearch Selain nama bidang indeks, Anda menentukan yang berikut:

- Jenis bidang indeks
- Apakah bidang dapat dicari (`text` dan `text-array` bidang selalu dapat dicari)
- Apakah bidang dapat digunakan sebagai kategori (facet)
- Apakah nilai bidang dapat dikembalikan dengan hasil pencarian
- Apakah bidang dapat digunakan untuk mengurutkan hasil
- Apakah sorotan dapat dikembalikan untuk bidang
- Nilai default untuk digunakan jika tidak ada nilai yang ditentukan dalam data dokumen.

Untuk informasi tentang cara mengonfigurasi bidang indeks untuk Amazon CloudSearch, lihat [configure indexing options](#).

## Aspek di Amazon CloudSearch

Facet adalah bidang indeks yang mewakili kategori yang ingin Anda gunakan untuk menyaring dan memfilter hasil pencarian. Saat mengirimkan permintaan penelusuran ke Amazon CloudSearch, Anda dapat meminta informasi aspek untuk mengetahui berapa banyak klik yang memiliki nilai yang sama dalam satu aspek. Anda dapat menampilkan informasi ini bersama dengan hasil pencarian dan menggunakannya untuk memungkinkan pengguna untuk secara interaktif memperbaiki pencarian mereka. (Ini sering disebut sebagai navigasi segi atau pencarian segi.)

Sebuah faset dapat berupa bidang tanggal, literal, atau numerik yang mengaktifkan faset dalam konfigurasi domain Anda. Untuk setiap aspek, Amazon CloudSearch menghitung jumlah klik yang memiliki nilai yang sama. Anda dapat menentukan bucket untuk menghitung jumlah faset untuk himpunan bagian tertentu dari nilai faset. Hanya ember yang memiliki kecocokan yang termasuk dalam hasil facet.

Untuk informasi tentang mengonfigurasi aspek, lihat. [configure indexing options](#) Untuk informasi tentang penggunaan informasi facet untuk mendukung navigasi faset, lihat. [Mendapatkan dan Menggunakan Informasi Facet di Amazon CloudSearch](#)

## Pemrosesan Teks di Amazon CloudSearch

Selama pengindeksan, Amazon CloudSearch memproses isi `text` dan `text-array` bidang sesuai dengan skema analisis khusus bahasa yang dikonfigurasi untuk bidang tersebut. Skema analisis mengontrol bagaimana teks dinormalisasi, diberi token, dan bertangkai, dan menentukan setiap stopword atau sinonim untuk diperhitungkan selama pengindeksan. Amazon CloudSearch menyediakan skema analisis default untuk setiap bahasa yang didukung. Untuk informasi tentang mengonfigurasi skema analisis kustom, lihat [Mengkonfigurasi Skema Analisis](#). Untuk informasi tentang cara Amazon CloudSearch menormalkan dan memberi token teks serta menerapkan opsi teks yang dikonfigurasi saat mengindeks bidang teks dan memproses permintaan pencarian, lihat [Pemrosesan Teks di Amazon CloudSearch](#)

## Menyortir Hasil di Amazon CloudSearch

Anda dapat menyesuaikan peringkat hasil penelusuran dengan menentukan ekspresi yang menghitung nilai kustom untuk setiap dokumen yang cocok dengan kriteria penelusuran Anda. Misalnya, Anda dapat menentukan ekspresi yang memperhitungkan nilai dalam `popularity` bidang dokumen serta skor relevansi default yang dihitung oleh Amazon CloudSearch Expressions hanyalah ekspresi numerik yang menggunakan operator dan fungsi numerik standar. Ekspresi dapat referensi `int` dan `double` bidang, ekspresi lain, skor relevansi dokumen (`_score`), serta waktu epoch (`_time`). Saat Anda mengirimkan permintaan pencarian, Anda menentukan ekspresi yang ingin Anda gunakan untuk mengurutkan hasil pencarian. Anda juga dapat mereferensikan ekspresi dalam kriteria pencarian Anda.

Relevansi dokumen `_score` menunjukkan seberapa relevan hit pencarian tertentu dengan permintaan pencarian. Untuk menghitung skor relevansi, Amazon CloudSearch memperhitungkan berapa kali istilah pencarian muncul dalam dokumen relatif terhadap dokumen lain dalam indeks.

Untuk informasi tentang cara mengonfigurasi ekspresi untuk domain Anda, lihat [Mengkonfigurasi Ekspresi](#).

## Permintaan Pencarian di Amazon CloudSearch

Anda mengirimkan permintaan pencarian ke titik akhir pencarian domain Anda sebagai permintaan GET HTTP/HTTPS. Anda dapat menentukan berbagai opsi untuk membatasi pencarian Anda, meminta informasi aspek, mengontrol peringkat, dan menentukan apa yang ingin Anda kembalikan dalam hasil. Anda bisa mendapatkan hasil pencarian di JSON atau XHTML. Secara default, Amazon CloudSearch mengembalikan hasil di JSON.

Saat Anda mengirimkan permintaan pencarian, Amazon CloudSearch melakukan pemrosesan teks pada string pencarian. String pencarian diproses menjadi:

- Ubah semua karakter menjadi huruf kecil
- Pisahkan string menjadi istilah terpisah pada batas spasi dan tanda baca
- Hapus istilah yang ada di daftar stopword untuk bidang yang sedang dicari.
- Peta batang dan sinonim sesuai dengan opsi stemming dan sinonim yang dikonfigurasi untuk bidang yang dicari.

Setelah preprocessing ini selesai, Amazon CloudSearch mencari istilah pencarian dalam indeks dan mengidentifikasi semua dokumen yang cocok dengan permintaan. Untuk menghasilkan respons, Amazon CloudSearch memproses daftar klik pencarian ini untuk memfilter dan mengurutkan dokumen yang cocok dan aspek komputasi. Amazon CloudSearch kemudian mengembalikan respons dalam JSON atau XHTML.

Secara default, Amazon CloudSearch mengembalikan hasil pencarian yang diberi peringkat sesuai dengan `_score` relevansi hit. Atau, permintaan Anda dapat menentukan bidang indeks atau ekspresi yang ingin Anda gunakan untuk mengurutkan klik. Misalnya, Anda mungkin ingin mengurutkan hits berdasarkan bidang indeks yang berisi harga atau ekspresi yang menghitung popularitas.

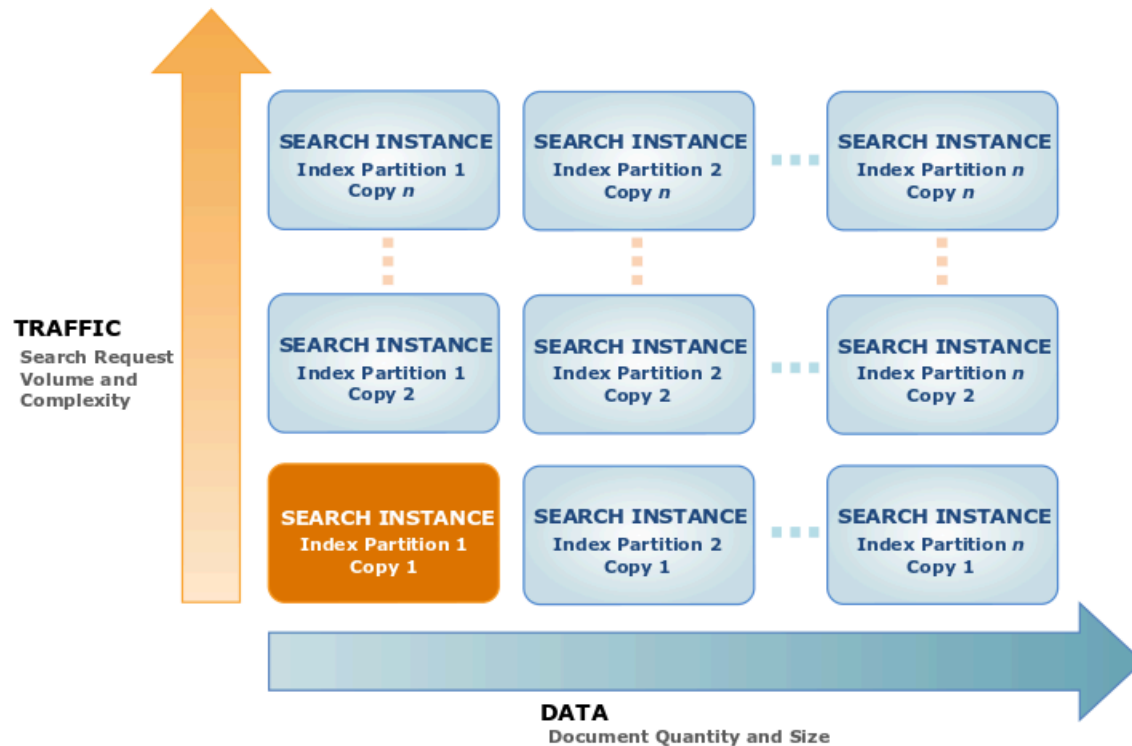
Untuk informasi selengkapnya tentang pencarian, peringkat, dan hasil paginasi, lihat [Mencari Data Anda dengan Amazon CloudSearch](#)

## Penskalaan Otomatis di Amazon CloudSearch

Sebuah domain pencarian memiliki satu atau lebih contoh pencarian, masing-masing dengan jumlah terbatas RAM dan sumber daya CPU untuk mengindeks data dan memproses permintaan. Berapa banyak contoh pencarian yang dibutuhkan domain tergantung pada dokumen dalam koleksi Anda dan volume serta kompleksitas permintaan pencarian Anda.

Amazon CloudSearch dapat menentukan ukuran dan jumlah instans penelusuran yang diperlukan untuk menghasilkan kinerja penelusuran dengan latensi rendah dan throughput tinggi. Saat mengunggah data dan mengonfigurasi indeks, Amazon CloudSearch membuat indeks dan memilih jenis instance penelusuran awal yang sesuai. Saat Anda menggunakan domain pencarian, Amazon CloudSearch dapat menskalakan untuk mengakomodasi jumlah data yang diunggah ke domain serta volume serta kompleksitas permintaan pencarian.

Saat Anda membuat domain penelusuran, satu instance akan digunakan untuk domain tersebut. Seperti yang ditunjukkan oleh ilustrasi berikut, Anda selalu memiliki setidaknya satu contoh untuk domain Anda. Amazon CloudSearch secara otomatis menskalakan domain dengan menambahkan instance saat volume data atau lalu lintas meningkat.



## Penskalaan untuk Data

Jika jumlah data yang Anda tambahkan ke domain melebihi kapasitas jenis instans penelusuran awal, Amazon CloudSearch menskalakan domain pencarian Anda ke jenis instans penelusuran yang lebih besar. Setelah domain melebihi kapasitas jenis instans penelusuran terbesar, Amazon CloudSearch mempartisi indeks pencarian di beberapa instance penelusuran. (Jumlah contoh pencarian yang diperlukan untuk menahan partisi indeks kadang-kadang disebut sebagai lebar domain.)

Ketika volume data dalam domain Anda menyusut, Amazon menurunkan CloudSearch skala domain Anda ke instance penelusuran yang lebih sedikit atau jenis instans penelusuran yang lebih kecil untuk meminimalkan biaya.

#### Note

Jika domain Anda telah ditingkatkan untuk mengakomodasi ukuran indeks Anda dan Anda menghapus sejumlah besar dokumen, skala domain akan turun saat berikutnya indeks penuh dibangun kembali. Meskipun indeks secara otomatis dibangun kembali secara berkala, untuk menurunkan skala secepat mungkin Anda dapat secara eksplisit [menjalankan pengindeksan](#) ketika Anda selesai menghapus dokumen.

## Penskalaan untuk Lalu Lintas

Saat volume atau kompleksitas permintaan pencarian Anda meningkat, dibutuhkan lebih banyak daya pemrosesan untuk menangani beban. Volume upload dokumen yang tinggi juga meningkatkan beban pada instance pencarian domain. Saat instance penelusuran mendekati pemuatan maksimumnya, Amazon CloudSearch akan menerapkan instance penelusuran duplikat untuk memberikan daya pemrosesan tambahan. (Jumlah contoh pencarian duplikat kadang-kadang disebut sebagai kedalaman domain.)

Saat lalu lintas turun, Amazon CloudSearch menghapus instance pencarian untuk meminimalkan biaya. Misalnya, domain baru mungkin ditingkatkan untuk menangani masuknya dokumen awal, dan mengurangi skala setelah Anda selesai mengunggah data Anda dan hanya mengirimkan pembaruan.

Jika domain Anda mengalami lonjakan lalu lintas yang tiba-tiba, Amazon akan CloudSearch menerapkan instance penelusuran tambahan. Namun, dibutuhkan beberapa menit untuk menyiapkan instance baru, sehingga Anda mungkin melihat peningkatan kesalahan 5xx hingga instance baru dapat mulai memproses permintaan. Untuk informasi selengkapnya tentang penanganan kesalahan 5xx, lihat [Penanganan Kesalahan](#).

Perlu diingat bahwa jenis dan kompleksitas permintaan pencarian Anda memengaruhi kinerja penelusuran secara keseluruhan dan dalam beberapa kasus meningkatkan jumlah instance pencarian yang diperlukan untuk mengoperasikan domain Anda. Mengirimkan sejumlah besar kumpulan dokumen kecil atau tunggal dapat memengaruhi kinerja domain penelusuran Anda. Untuk informasi selengkapnya, lihat [Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch](#).



# Mengakses Amazon CloudSearch

Anda dapat mengakses Amazon CloudSearch melalui CloudSearch konsol Amazon, AWS SDK, atau AWS CLI

- [CloudSearchKonsol Amazon](#) memungkinkan Anda membuat, mengonfigurasi, dan memantau domain pencarian, mengunggah dokumen, dan menjalankan pencarian pengujian dengan mudah. Menggunakan konsol adalah cara termudah untuk memulai dengan Amazon CloudSearch dan menyediakan pusat komando pusat untuk pengelolaan domain pencarian Anda yang sedang berlangsung.
- [AWS SDK](#) mendukung semua operasi Amazon CloudSearch API, sehingga memudahkan pengelolaan dan interaksi dengan domain penelusuran menggunakan teknologi pilihan Anda. SDK secara otomatis menandatangani permintaan sesuai kebutuhan menggunakan kredensial AWS Anda.
- Ini [AWS CLI](#) membungkus semua operasi Amazon CloudSearch API untuk menyediakan cara sederhana untuk membuat dan mengonfigurasi domain pencarian, mengunggah data yang ingin Anda telusuri, dan mengirimkan permintaan pencarian. AWS CLI secara otomatis menandatangani permintaan sesuai kebutuhan menggunakan kredensial AWS Anda.

## Wilayah dan Titik Akhir untuk Amazon CloudSearch

Amazon CloudSearch menyediakan titik akhir regional untuk mengakses layanan konfigurasi dan titik akhir khusus domain untuk mengakses layanan pencarian dan dokumen.

Anda menggunakan layanan konfigurasi untuk membuat dan mengelola domain pencarian Anda. Titik akhir layanan konfigurasi khusus wilayah berbentuk: `cloudsearch.region.amazonaws.com`. Misalnya, `cloudsearch.us-east-1.amazonaws.com`. Untuk daftar wilayah yang didukung saat ini, lihat [Wilayah dan Titik Akhir](#) di Referensi Umum AWS

Untuk mengakses layanan CloudSearch penelusuran dan dokumen Amazon, Anda menggunakan titik akhir khusus domain terpisah:

- `http://doc-domainname-domainid.us-east-1.cloudsearch.amazonaws.com` Endpoint layanan dokumen domain digunakan untuk mengunggah dokumen
- `http://search-domainname-domainid.us-east-1.cloudsearch.amazonaws.com`—titik akhir pencarian domain digunakan untuk mengirimkan permintaan pencarian

## Menandatangani CloudSearch Permintaan Amazon

Jika Anda menggunakan bahasa yang AWS menyediakan SDK, sebaiknya gunakan SDK untuk mengirimkan permintaan Amazon CloudSearch. Semua AWS SDK sangat menyederhanakan proses penandatanganan permintaan dan menghemat banyak waktu jika dibandingkan dengan menggunakan CloudSearch API Amazon secara langsung. SDK terintegrasi dengan mudah dengan lingkungan pengembangan Anda dan menyediakan akses mudah ke perintah terkait. Anda juga dapat menggunakan CloudSearch konsol Amazon dan AWS CLI untuk mengirimkan permintaan yang ditandatangani tanpa upaya tambahan.

Jika Anda memilih untuk memanggil Amazon CloudSearch API secara langsung, Anda harus menandatangani permintaan Anda sendiri. Permintaan layanan konfigurasi harus selalu ditandatangani. Permintaan unggah, cari, dan saran harus ditandatangani, kecuali jika Anda mengonfigurasi akses anonim untuk layanan tersebut. Untuk menandatangani permintaan, Anda menghitung tanda tangan digital menggunakan fungsi hash kriptografi, yang mengembalikan nilai hash berdasarkan input. Input termasuk teks permintaan Anda dan kunci akses rahasia Anda. Fungsi hash mengembalikan nilai hash yang Anda sertakan dalam permintaan sebagai tanda tangan Anda. Tanda tangan adalah bagian dari header Otorisasi permintaan Anda. Setelah menerima permintaan Anda, Amazon CloudSearch menghitung ulang tanda tangan menggunakan fungsi hash yang sama dan input yang Anda gunakan untuk menandatangani permintaan. Jika tanda tangan yang dihasilkan cocok dengan tanda tangan dalam permintaan, Amazon CloudSearch memproses permintaan tersebut. Jika tidak, permintaan ditolak.

Amazon CloudSearch mendukung otentikasi menggunakan AWS Signature Version 4. Untuk informasi selengkapnya, lihat [Proses Penandatanganan Signature Version 4](#).

# Memulai dengan Amazon CloudSearch

Untuk mulai mencari data Anda dengan Amazon CloudSearch, Anda cukup mengambil langkah-langkah berikut:

- Buat dan konfigurasi domain pencarian
- Unggah dan indeks data yang ingin Anda cari
- Kirim permintaan pencarian ke domain Anda

Tutorial ini menunjukkan kepada Anda cara memulai dan menjalankan menggunakan AWS Management Console untuk Amazon CloudSearch. Untuk membuatnya lebih mudah untuk memulai, kami telah membuat kumpulan data sampel 5.000 judul film populer yang dapat Anda unduh dan periksa, unggah ke domain pencarian Anda sendiri, dan kirimkan kueri penelusuran untuk melihat cara CloudSearch kerja Amazon.

Menggunakan AWS Management Console dan contoh data film, Anda akan memiliki domain pencarian sendiri dan berjalan dalam waktu sekitar setengah jam.

Untuk memulai, [Get Signed Up](#).

## Topik

- [Sebelum Anda Memulai dengan Amazon CloudSearch](#)
- [Langkah 1: Buat CloudSearch Domain Amazon](#)
- [Langkah 2: Unggah Data ke Amazon CloudSearch untuk Pengindeksan](#)
- [Langkah 3: Cari CloudSearch Domain Amazon Anda](#)
- [Langkah 4: Hapus Domain CloudSearch Film Amazon Anda](#)

## Sebelum Anda Memulai dengan Amazon CloudSearch

Untuk menggunakan Amazon CloudSearch, Anda memerlukan akun Amazon Web Services (AWS). Akun AWS memungkinkan Anda mengakses Amazon CloudSearch dan AWS layanan lainnya, seperti Amazon Simple Storage Service (Amazon S3) dan Amazon Elastic Compute Cloud (Amazon EC2). Seperti layanan AWS lainnya, Anda hanya membayar untuk CloudSearch sumber daya Amazon yang Anda gunakan. Tidak ada biaya pendaftaran dan biaya tidak dikeluarkan sampai Anda membuat domain pencarian.

Jika Anda sudah memiliki akun AWS, Anda secara otomatis mendaftar ke Amazon CloudSearch.

Untuk membuat akun AWS

1. Buka <https://aws.amazon.com> dan klik Daftar Sekarang.
2. Ikuti instruksi untuk mendaftar. Anda harus memasukkan informasi pembayaran sebelum dapat mulai menggunakan Amazon CloudSearch.

## Langkah 1: Buat CloudSearch Domain Amazon

CloudSearch Domain Amazon merangkul kumpulan data yang ingin Anda telusuri, instance penelusuran yang memproses permintaan penelusuran, dan konfigurasi yang mengontrol cara data Anda diindeks dan dicari. Anda membuat domain penelusuran terpisah untuk setiap kumpulan data yang ingin dicari. Untuk setiap domain, Anda mengonfigurasi opsi pengindeksan yang menjelaskan bidang yang ingin Anda sertakan dalam indeks dan cara Anda ingin menggunakannya, skema analisis yang menentukan opsi pemrosesan teks khusus bahasa untuk setiap bidang, ekspresi yang dapat Anda gunakan untuk menyesuaikan peringkat hasil penelusuran, dan kebijakan akses yang mengontrol akses ke dokumen domain dan titik akhir penelusuran.

Anda berinteraksi dengan domain pencarian untuk:

- Konfigurasi indeks dan opsi pencarian
- Kirim data untuk pengindeksan
- Lakukan pencarian

Setiap domain memiliki titik akhir unik di mana Anda mengirimkan permintaan pencarian ke domain. Misalnya, titik akhir untuk domain yang disebut film yang dibuat di wilayah AS Timur (Virginia N.) mungkin:

Example

```
search-movies-mtshfsu2rje7ywr66uit3dei4m.us-east-1.cloudsearch.amazonaws.com
```

Saat membuat domain pencarian, Anda menentukan nama unik untuk domain tersebut. Nama domain harus dimulai dengan huruf atau angka dan setidaknya 3 dan tidak lebih dari 28 karakter. Karakter yang diizinkan adalah: a-z, 0-9, dan tanda hubung (-). Secara default, domain baru dibuat di

wilayah AS Timur (Virginia N.). Untuk membuat domain di wilayah lain, Anda harus secara eksplisit menentukan wilayah saat membuat domain.

Untuk mengkonfigurasi domain baru, Anda harus menentukan:

- Opsi pengindeksan untuk data yang ingin Anda cari.
- Kebijakan akses untuk layanan dokumen domain dan titik akhir layanan pencarian.

Tutorial ini menunjukkan cara membuat dan berinteraksi dengan domain menggunakan CloudSearch konsol Amazon. Untuk mempelajari selengkapnya, lihat [Membuat Domain Pencarian](#).

#### Important

Domain yang akan Anda buat akan ditayangkan dan Anda akan dikenakan biaya CloudSearch penggunaan Amazon standar untuk domain tersebut hingga Anda menghapusnya. Untuk informasi selengkapnya tentang tarif CloudSearch penggunaan Amazon, buka [halaman CloudSearch detail Amazon](#).

Untuk membuat domain film Anda

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Pilih Create domain (Buat domain).
3. Masukkan nama untuk domain baru Anda. Nama domain harus dimulai dengan huruf atau angka dan minimal 3 dan tidak lebih dari 28 karakter. Nama domain dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung). Huruf besar dan garis bawah tidak diperbolehkan.
4. Biarkan pengaturan lain sebagai defaultnya dan pilih Berikutnya.
5. Pilih Sampel data dan pilih film IMDb (demo) dari dropdown. Anda juga dapat secara otomatis mengonfigurasi domain pencarian dengan menganalisis sampel data Anda.
6. Pilih Berikutnya.
7. Tinjau bidang indeks yang sedang dikonfigurasi. Sebelas bidang dikonfigurasi secara otomatis untuk data imdb-movie: aktor, sutradara, genre, image\_url, plot, peringkat, rating, release\_date, running\_time\_secs, judul, dan tahun.

**Note**

Secara default, semua opsi diaktifkan untuk setiap bidang. Meskipun ini nyaman untuk pengembangan dan pengujian, menyempurnakan opsi yang dikonfigurasi untuk setiap bidang sesuai dengan cara Anda menggunakan bidang tersebut dapat mengurangi ukuran indeks Anda. Jika domain Anda menggunakan lebih dari satu contoh penelusuran kecil, penyetelan dapat membantu meminimalkan biaya menjalankan domain Anda.

Ketika Anda selesai meninjau opsi pengindeksan, pilih Berikutnya.

8. Untuk kesederhanaan dalam tutorial ini, gunakan domain akses terbuka. Pilih Izinkan akses terbuka ke domain dan pilih Berikutnya.
9. Tinjau konfigurasi domain dan klik Buat untuk membuat domain Anda.

Amazon CloudSearch menginisialisasi sumber daya untuk domain, yang dapat memakan waktu sekitar sepuluh menit. Selama proses inisialisasi ini, status domain adalah Processing. Setelah status berubah menjadi Aktif, Anda dapat mengunggah data Anda dan mulai mencari.

## Langkah 2: Unggah Data ke Amazon CloudSearch untuk Pengindeksan

Anda mengunggah data yang ingin dicari ke domain Anda sehingga Amazon CloudSearch dapat membuat dan menerapkan indeks yang dapat dicari. Untuk diindeks oleh Amazon CloudSearch, data harus diformat dalam JSON atau XHTML. CloudSearch Konsol Amazon dapat secara otomatis mengonversi jenis file berikut ke format yang diperlukan:

- Batch dokumen diformat dalam JSON atau XML.json, .xml.
- Nilai Terpisah Koma (.csv)
- Dokumen Teks (.txt)

Saat Anda mengunggah file CSV, Amazon CloudSearch mem-parsing setiap baris secara terpisah. Baris pertama mendefinisikan bidang dokumen, dan setiap baris berikutnya menjadi dokumen terpisah. Untuk semua jenis file lainnya Amazon CloudSearch membuat satu dokumen dan isi file

dipetakan ke bidang teks tunggal. Jika metadata tersedia untuk file, metadata dipetakan ke bidang dokumen yang sesuai—bidang yang dihasilkan dari metadata dokumen bervariasi tergantung pada jenis file.

Sampel data film IMDb sudah diformat dalam JSON.

Tutorial ini menunjukkan cara mengirimkan data melalui CloudSearch konsol Amazon, tetapi Anda juga dapat [convert](#) dan [upload documents](#) dengan alat baris perintah, dan mengunggah dokumen menggunakan [dokumen/batch](#) sumber daya. (Untuk mengunggah lebih dari 5 MB data, Anda harus menggunakan alat baris perintah atau API.)

Untuk mengunggah data sampel ke domain film Anda

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih Domain. Pilih nama domain film Anda untuk melihat dasbor domain.
3. Pilih Tindakan, Unggah dokumen.
4. Pilih Sampel data dan pilih film IMDb (demo) dari dropdown.
5. Pilih Berikutnya.
6. Tinjau ringkasan unggahan dan pilih Unggah dokumen untuk mengirim data ke domain Anda untuk diindeks.

#### Note

Untuk melihat bagaimana data diformat, pilih Unduh kumpulan dokumen yang dihasilkan. Untuk informasi selengkapnya tentang menyiapkan data Anda sendiri, lihat [Mempersiapkan Data Anda](#).

Anda sekarang memiliki CloudSearch domain Amazon yang berfungsi penuh yang dapat Anda mulai mencari. Pembaruan diterapkan terus menerus dalam urutan mereka diterima, sehingga Anda dapat mulai mencari domain Anda segera.

## Langkah 3: Cari CloudSearch Domain Amazon Anda

Anda dapat menggunakan penguji penelusuran di CloudSearch konsol Amazon untuk mengirimkan contoh permintaan pencarian dan melihat hasilnya. Anda juga dapat mengirimkan contoh permintaan pencarian melalui browser Web atau menggunakan cURL. Dalam aplikasi Anda, Anda dapat

menggunakan perpustakaan HTTP apa pun untuk mengirim lalu lintas pencarian ke CloudSearch domain Amazon Anda.

## Mencari dengan Search Tester

Penguji penelusuran di CloudSearch konsol Amazon memungkinkan Anda mengirimkan contoh permintaan penelusuran menggunakan salah satu parser kueri yang didukung: sederhana, terstruktur, lucene, atau dismax. Secara default, permintaan diproses dengan parser kueri sederhana. Anda dapat menentukan opsi untuk parser yang dipilih, memfilter dan mengurutkan hasil, dan menelusuri aspek yang dikonfigurasi. Hit pencarian secara otomatis disorot dalam hasil pencarian. Untuk informasi tentang bagaimana hal ini dilakukan, lihat [Menyoroti Pencarian Hits di Amazon CloudSearch](#). Anda juga dapat memilih sugester untuk mendapatkan saran saat Anda memasukkan istilah di kolom Penelusuran. (Anda harus mengonfigurasi sugester sebelum Anda bisa mendapatkan saran. Untuk informasi lebih lanjut lihat [Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch](#).)

Secara default, hasil diurutkan menurut skor relevansi yang dihasilkan secara otomatis, `_score`. Untuk informasi tentang menyesuaikan bagaimana hasil diberi peringkat, lihat [Menyortir Hasil di Amazon CloudSearch](#).

Untuk mencari domain Anda

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih domain film Anda untuk membuka konfigurasinya.
3. Pilih Jalankan pencarian pengujian.
4. Untuk melakukan pencarian teks sederhana, masukkan kueri penelusuran dan pilih Jalankan. Secara default, semua `text` dan `text-array` bidang dicari.

Untuk mencari bidang tertentu, perluas Opsi dan masukkan daftar bidang yang dipisahkan koma yang ingin Anda cari di bidang bidang Penelusuran. Anda dapat menambahkan bobot ke setiap bidang dengan tanda sisipan (^) untuk mengontrol kepentingan relatif setiap bidang dalam hasil pencarian. Misalnya, menentukan hit `title^5`, `description` bobot di `title` lapangan lima kali lebih banyak daripada hit di `description` lapangan saat menghitung skor relevansi untuk setiap dokumen yang cocok.

Untuk menggunakan sintaks kueri terstruktur, pilih Structured dari menu Query parser. Setelah Anda memilih parser kueri terstruktur, masukkan kueri terstruktur Anda di bidang Penelusuran dan pilih Jalankan. Misalnya, untuk menemukan semua film dengan bintang dalam judul yang dirilis pada



tahun 2000 atau sebelumnya, Anda dapat memasukkan:(and title:'star' year:{,2000}). Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#). Untuk mengirimkan Lucene atau DisMax kueri, pilih parser kueri yang sesuai.

Anda dapat menentukan opsi tambahan untuk parser kueri yang dipilih untuk mengonfigurasi operator default dan mengontrol operator mana yang dapat digunakan dalam kueri. Untuk informasi selengkapnya, lihat [Parameter Permintaan Pencarian](#).

Anda dapat menyalin dan menempelkan URL permintaan untuk mengirimkan permintaan dan melihat respons dari browser Web. Permintaan dapat dikirim melalui HTTP atau HTTPS.

## Mengirimkan Permintaan Pencarian dari Browser Web

Anda dapat mengirimkan permintaan pencarian langsung ke titik akhir pencarian Anda dari browser Web apa pun. Anda dapat menggunakan salah satu parser kueri (sederhana, terstruktur, lucene, atau dismax) dan menentukan berbagai opsi untuk membatasi pencarian Anda, meminta informasi aspek, menyesuaikan peringkat, dan mengontrol informasi apa yang dikembalikan dalam hasil.

Misalnya, untuk mencari domain film Anda dan mendapatkan judul semua film Star Wars yang tersedia, tambahkan string pencarian berikut ke titik akhir pencarian Anda. (2013-01-01 adalah versi API dan harus ditentukan.)

### Example

```
/2013-01-01/search?q=star+wars&return=title
```

#### Note

Titik akhir pencarian domain Anda ditampilkan di dasbor domain. Anda juga dapat melakukan pencarian dari AWS Management Console, melihat permintaan dan respons mentah, dan menyalin URL permintaan dari bidang Permintaan Pencarian. Endpoint pencarian dan layanan dokumen domain tetap sama selama masa pakai domain.

Secara default, Amazon CloudSearch mengembalikan respons di JSON. Anda juga bisa mendapatkan hasil pencarian diformat dalam XHTML dengan menentukan parameter, `format=xhtml` (Perhatikan bahwa kesalahan dapat dikembalikan dalam JSON atau XHTML, tergantung dari mana kesalahan berasal.)

## Mencari Bidang Numerik

Anda dapat menggunakan sintaks kueri terstruktur, `q.parser=structured`, untuk menemukan dokumen yang memiliki atribut numerik tertentu. Anda dapat mencari nilai yang tepat atau rentang nilai dalam bidang numerik apa pun (`double`, `double-array`, `int`, `int-array`). Untuk mencari rentang, Anda menentukan batas atas dan bawah, dipisahkan oleh koma, dan melampirkan rentang dalam tanda kurung atau tanda kurung. Gunakan tanda kurung siku (`[,]`) saat Anda ingin menyertakan batas, dan kurawal kurawal (`{,}`) untuk mengecualikan batas. Sebagai contoh:

- `year:2000` cocok dengan dokumen yang bidang tahunnya berisi nilai 2000.
- `year:[2000, ]` cocok dengan dokumen yang bidang tahunnya berisi nilai lebih besar dari atau sama dengan 2000
- `year:{, 2000]` cocok dengan dokumen yang bidang tahunnya berisi nilai kurang dari atau sama dengan 2000
- `year:[2000, 2011]` cocok dengan dokumen yang bidang tahunnya berisi nilai antara 2000 dan 2011, inklusif.
- `year:{2000, 2011}` cocok dengan dokumen yang bidang tahunnya berisi nilai antara 2000 dan 2011, eksklusif.

Anda juga dapat mencari bidang tanggal untuk tanggal atau rentang tanggal tertentu, tetapi Anda harus melampirkan setiap string tanggal dalam tanda kutip tunggal: `release_date:[ '2000-01-01T00:00:00Z' , '2011-01-01T00:00:00Z' ]`.

Misalnya, kueri terstruktur berikut mencari “bintang” di bidang judul, menemukan semua film yang cocok yang dirilis sebelum tahun 2000, dan mengembalikan skor judul, tahun, dan relevansi untuk masing-masing film:

### Example

```
q=(and title:'star' year:{,2000})&q.parser=structured&return=title,year,_score
```

Respons menunjukkan status permintaan, jumlah dokumen yang cocok, dan bidang yang diminta untuk setiap klik.

```
{
  "status": {
```

```
    "rid": "hLPckLsoEQoELQo=",
    "time-ms": 2
  },
  "hits": {
    "found": 15,
    "start": 0,
    "hit": [
      {
        "id": "tt0076759",
        "fields": {
          "title": "Star Wars",
          "year": "1977",
          "_score": "5.7601414"
        }
      },
      .
      .
      .
      {
        "id": "tt0088170",
        "fields": {
          "title": "Star Trek III: The Search for Spock",
          "year": "1984",
          "_score": "4.2371693"
        }
      }
    ]
  }
}
```

Untuk informasi selengkapnya tentang membuat kueri penelusuran, lihat. [Mencari Data Anda dengan Amazon CloudSearch](#)

## Menyortir Hasil Pencarian

Secara default, Amazon CloudSearch mengurutkan hasil pencarian sesuai dengan relevansi `_score` yang dihasilkan secara otomatis. Anda dapat mengubah peringkat hasil dengan menggunakan parameter pengurutan dalam permintaan pencarian Anda untuk menentukan bidang atau ekspresi yang ingin Anda gunakan untuk peringkat. (Ekspresi adalah ekspresi numerik kustom yang dapat dievaluasi untuk setiap dokumen dalam kumpulan dokumen yang cocok. Untuk informasi tentang mendefinisikan ekspresi Anda sendiri, lihat [Mengkonfigurasi Ekspresi](#).)

Jika Anda menentukan bidang teks dengan `sort` parameter, hasilnya diurutkan menurut abjad sesuai dengan bidang itu. Misalnya, untuk mengurutkan hasil dari domain film menurut abjad menurut judul, tambahkan `&sort=title asc` ke string kueri Anda:

### Example

```
2013-01-01/search?q=(and genres:'Sci-Fi' year:
[,2000])&q.parser=structured&return=title,year&sort=title asc
```

Perhatikan bahwa Anda harus secara eksplisit menentukan arah pengurutan, `asc` (naik) atau `desc` (turun). Saat Anda mengurutkan menurut abjad, Amazon CloudSearch mengurutkan berdasarkan titik kode Unicode. Ini berarti angka datang sebelum huruf dan huruf besar datang sebelum huruf kecil. Angka diurutkan sebagai string; misalnya, 10 akan datang sebelum 2.

Demikian pula, Anda dapat menentukan bidang integer dengan `sort` parameter untuk mengurutkan hasil secara numerik.

Jika Anda menentukan daftar kolom atau ekspresi yang dipisahkan koma, bidang atau ekspresi pertama digunakan sebagai kriteria pengurutan primer, yang kedua digunakan sebagai kriteria pengurutan sekunder, dan seterusnya.

Untuk informasi selengkapnya tentang hasil peringkat, lihat [Menyortir Hasil di Amazon CloudSearch](#)

## Mendapatkan Informasi Facet

Facet adalah bidang indeks yang mewakili kategori yang ingin Anda gunakan untuk menyaring dan memfilter hasil pencarian. Saat mengirimkan permintaan penelusuran ke Amazon CloudSearch, Anda dapat meminta informasi aspek untuk mengetahui berapa banyak klik yang memiliki nilai yang sama dalam satu aspek. Anda dapat menampilkan informasi ini bersama dengan hasil pencarian dan menggunakannya untuk memungkinkan pengguna untuk secara interaktif menyempurnakan pencarian mereka. (Ini sering disebut sebagai navigasi segi atau pencarian segi.)

Sebuah faset dapat berupa bidang tanggal, literal, atau numerik yang mengaktifkan faset dalam konfigurasi domain Anda. Untuk setiap aspek, Amazon CloudSearch menghitung jumlah klik yang memiliki nilai yang sama. Anda dapat menentukan bucket untuk menghitung jumlah faset untuk himpunan bagian tertentu dari nilai faset. Hanya ember yang memiliki kecocokan yang termasuk dalam hasil facet.

Untuk mendapatkan jumlah aspek dengan hasil pencarian Anda

- Gunakan `facet.FIELD` opsi untuk menentukan bidang yang ingin Anda hitung aspeknya. Untuk contoh film IMDb, `faceting` data diaktifkan untuk bidang berikut: `genres`, `rank`, `rating`, `release_date`, `running_time_secs`, dan `year`. Opsi `facet` ditentukan sebagai objek JSON. Jika objek JSON kosong, `facet.FIELD={}`, jumlah faset dihitung untuk semua nilai bidang, faset diurutkan berdasarkan jumlah faset, dan 10 aspek teratas dikembalikan dalam hasil:

```
q=star&return=title&facet.genres={}
```

Aspek muncul di bawah hit dalam hasil.

```
facets": {  
  "genres": {  
    "buckets": [  
      {"value": "Comedy", "count": 41},  
      .  
      .  
      {"value": "Sport", "count": 7}  
    ]  
  }  
}
```

Anda dapat menentukan opsi untuk menghitung aspek untuk nilai bidang yang dipilih, menentukan jumlah maksimum nilai faset yang akan disertakan dalam hasil, dan mengontrol bagaimana aspek diurutkan.

Untuk menentukan bucket untuk menghitung jumlah faset untuk nilai bidang yang dipilih, Anda menentukan opsi `buckets`. Misalnya, permintaan berikut mengurutkan jumlah aspek untuk bidang tahun menurut dekade:

```
q=star&facet.year={buckets:["[1970,1979]","[1980,1989]","[1990,1999]"]}
```

Ini membatasi jumlah faset ke tiga rentang yang ditentukan:

```
"facets": {  
  "year": {
```

```
    "buckets": [
      {"value": "[1970,1979]", "count": 3},
      {"value": "[1980,1989]", "count": 7},
      {"value": "[1990,1999]", "count": 12}
    ]
  }
}
```

Untuk informasi selengkapnya tentang menentukan opsi facet, lihat. [Mendapatkan dan Menggunakan Informasi Facet di Amazon CloudSearch](#)

## Mendapatkan Sorotan Pencarian

Sorotan pencarian adalah kutipan dari bidang teks atau teks-array yang menunjukkan di mana istilah pencarian terjadi di dalam bidang.

Untuk mendapatkan informasi sorotan dengan hasil pencarian Anda

- Gunakan `highlight.FIELD` opsi untuk menentukan bidang teks atau teks-array yang ingin Anda soroti. Bidang harus disorot diaktifkan dalam opsi pengindeksan domain Anda. Untuk contoh penyorotan data film IMDb diaktifkan untuk bidang berikut: `actors`, `directorsplot`, dan `title`. Opsi sorotan ditentukan sebagai objek JSON. Jika objek JSON kosong `highlight.FIELD={}`, Amazon CloudSearch menyoroti semua kemunculan istilah pencarian dengan melampirkannya dalam tag penekanan HTML `<em>term</em>`, dan kutipan dikembalikan sebagai HTML.

```
q=title:'star'&q.parser=structured&return=_no_fields&highlight.title={}
```

Informasi sorotan disertakan dengan setiap hit pencarian.

```
hits": {
  "found": 29,
  "start": 0,
  "hit": [
    {
      "id": "tt0796366",
      "highlights": {
        "title": "<em>Star</em> Trek"
      }
    },
  ],
}
```

```
.
.
.
{
  "id": "tt2488496",
  "highlights": {
    "title": "<em>Star</em> Wars: Episode VII"
  }
}
]
```

Untuk informasi selengkapnya tentang menentukan opsi sorotan, lihat [Menyoroti Pencarian Hits di Amazon CloudSearch](#).

## Langkah 4: Hapus Domain CloudSearch Film Amazon Anda

Setelah selesai bereksperimen dengan domain film Anda, Anda harus menghapusnya agar tidak dikenakan biaya penggunaan tambahan.

### Important

Menghapus domain menghapus indeks yang terkait dengan domain dan mengambil dokumen domain dan titik akhir pencarian offline secara permanen.

Untuk menghapus domain imdb-movies

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home> dan arahkan ke daftar domain.
2. Pilih kotak centang untuk domain film, lalu pilih Hapus dan konfirmasi penghapusan.

### Note

Diperlukan waktu sekitar 15 menit untuk menghapus domain dan sumber dayanya.

Ingin tahu ke mana harus pergi selanjutnya? [Apakah Anda baru mengenal Amazon CloudSearch?](#) memiliki panduan untuk dokumentasi CloudSearch pengembang Amazon lainnya.

Untuk informasi selengkapnya tentang bahasa CloudSearch kueri Amazon, lihat [Mencari Data Anda dengan Amazon CloudSearch](#). Jika Anda siap menyiapkan domain dengan data Anda sendiri, lihat [Mempersiapkan Data Anda](#) dan [upload documents](#).



# Migrasi ke Amazon CloudSearch 2013-01-01 API

Amazon CloudSearch 2013-01-01 API menawarkan beberapa fitur baru, termasuk dukungan untuk berbagai bahasa, menyoroti istilah penelusuran dalam hasil, dan mendapatkan saran. Untuk menggunakan fitur-fitur ini, Anda membuat dan mengonfigurasi domain pencarian 2013-01-01 baru, memodifikasi pipeline data Anda untuk mengisi domain baru menggunakan format data 2013-01-01, dan memperbarui pipeline kueri Anda untuk mengirimkan permintaan dalam format permintaan 2013-01-01. Panduan migrasi ini merangkum perubahan API dan menyoroti perubahan yang paling mungkin memengaruhi aplikasi Anda.

## Membuat 2013-01-01 Domain Amazon CloudSearch

Jika Anda membuat CloudSearch domain Amazon sebelum peluncuran API 2013-01-01, Anda dapat memilih versi API mana yang akan digunakan saat membuat domain baru. Untuk membuat domain 2013-01-01 melalui konsol, pilih versi 2013-01-01 di Create Domain Wizard. Untuk membuat domain 2013-01-01 dari baris perintah, unduh dan instal AWS CLI dan jalankan perintah. `aws cloudsearch create-domain`

### Note

Untuk membuat dan berinteraksi dengan domain 2013-01-01, Anda harus menggunakan alat AWS CLI. Untuk membuat dan berinteraksi dengan domain 2011-02-01, Anda harus menggunakan alat v1.

## Mengkonfigurasi 2013-01-01 Domain Amazon CloudSearch

Anda dapat mengonfigurasi domain 2013-01-01 melalui konsol, alat baris perintah, atau AWS SDK. 2013-01-01 domain mendukung beberapa opsi konfigurasi baru:

- **Skema Analisis** —Anda mengonfigurasi skema analisis untuk menentukan opsi pemrosesan teks khusus bahasa untuk `text` dan bidang `text-array` Amazon CloudSearch sekarang mendukung 33 bahasa, serta opsi untuk bidang multi-bahasa. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Skema Analisis](#). Untuk daftar lengkap bahasa yang didukung, lihat [Bahasa yang Didukung](#).

- Opsi Ketersediaan — Anda dapat mengaktifkan opsi Multi-AZ untuk memperluas domain ke zona ketersediaan kedua untuk memastikan ketersediaan jika terjadi gangguan layanan. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Ketersediaan](#).
- Opsi Penskalaan — Anda dapat mengatur jenis instans yang diinginkan dan jumlah replikasi yang diinginkan untuk meningkatkan kapasitas unggahan atau pencarian, mempercepat permintaan pencarian, dan meningkatkan toleransi kesalahan. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).
- Penyaran—Anda dapat mengonfigurasi sugester untuk mengimplementasikan fungsionalitas pelengkapan otomatis. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Suggesters untuk Amazon CloudSearch](#).

Akses ke layanan CloudSearch konfigurasi Amazon dikelola melalui IAM dan sekarang memungkinkan Anda untuk mengontrol akses ke tindakan konfigurasi tertentu. Perhatikan bahwa Amazon CloudSearch ARN juga telah berubah. Akses ke dokumen domain dan titik akhir penelusuran dikelola melalui layanan CloudSearch konfigurasi Amazon. Untuk informasi selengkapnya, lihat [configure access policies](#).

2013-01-01 domain juga mendukung serangkaian opsi pengindeksan yang diperluas:

- Skema Analisis — Anda mengonfigurasi pemrosesan teks khusus bahasa berdasarkan per bidang dengan menentukan skema analisis untuk masing-masing bidang dan bidang. `text text-array` Untuk informasi selengkapnya, lihat [Mengkonfigurasi Skema Analisis](#).
- Jenis Bidang — Amazon CloudSearch sekarang mendukung 11 jenis bidang:
  - tanggal-berisi stempel waktu. Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut IETF RFC3339: `T 00:00:00 Z. yyyy-mm-dd Di UTC`, misalnya, 17:00 23 Agustus 1970 adalah: `1970-08-23T 17:00:00 Z`.
  - `date-array`—bidang tanggal yang dapat berisi beberapa nilai.
  - `double`—berisi nilai floating point 64-bit presisi ganda.
  - `double-array` — bidang ganda yang dapat berisi beberapa nilai.
  - `int`—berisi nilai integer bertanda 64-bit.
  - `int-array`—bidang integer yang dapat berisi beberapa nilai.
  - `latlon`—berisi lokasi yang disimpan sebagai pasangan nilai lintang dan bujur.
  - `literal`—berisi pengenalan atau data lain yang ingin Anda cocokkan dengan tepat.
  - `literal-array`—bidang literal yang dapat berisi beberapa nilai.

- teks-berisi data alfanumerik arbitrer.
- `text-array`—bidang teks yang dapat berisi beberapa nilai.
- Sorot —saat Anda mengaktifkan opsi sorotan untuk bidang, Anda dapat mengambil kutipan yang menunjukkan di mana istilah penelusuran muncul dalam bidang tersebut. Untuk informasi selengkapnya, lihat [Menyoroti Pencarian Hits di Amazon CloudSearch](#).
- Sumber —Anda dapat menentukan sumber untuk bidang untuk menyalin data dari satu bidang ke bidang lainnya, memungkinkan Anda menggunakan data sumber yang sama dengan cara yang berbeda dengan mengonfigurasi opsi yang berbeda untuk bidang tersebut.

Saat mengonfigurasi domain 2013-01-01 Anda, ada beberapa hal yang perlu diingat:

- Secara default, saat Anda menambahkan bidang, semua opsi yang valid untuk jenis bidang tersebut diaktifkan. Meskipun ini berguna untuk pengembangan dan pengujian, menonaktifkan opsi yang tidak Anda perlukan dapat mengurangi ukuran indeks Anda dan meningkatkan kinerja.
- Anda harus menggunakan bidang tipe array terpisah untuk bidang multi-nilai.
- Hanya bidang nilai tunggal yang dapat diurutkan diaktifkan.
- Hanya `text` dan `text-array` bidang yang dapat disorot diaktifkan.
- Semua bidang kecuali `text` dan `text-array` bidang dapat diaktifkan facet.
- Bidang literal sekarang peka huruf besar/kecil.
- Anda tidak lagi harus menyimpan nilai floating point sebagai bilangan bulat — gunakan bidang `double`.
- Anda dapat menyimpan lokasi menggunakan jenis `latlon` bidang baru. Untuk informasi selengkapnya, lihat [location-based searching and sorting](#).
- `int` Bidang adalah integer bertanda 64-bit.
- Alih-alih mengonfigurasi bidang pencarian default, Anda dapat menentukan bidang mana yang akan dicari dengan `q.options` parameter dalam permintaan pencarian Anda. `q.options` Parameter ini juga memungkinkan Anda untuk menentukan bobot untuk masing-masing bidang.
- Saat menyortir dan mengonfigurasi ekspresi, Anda mereferensikan skor relevansi default dengan nama. `_score` Karena perubahan dalam algoritma relevansi, skor yang dihitung akan berbeda dari yang berada di bawah API 2011-02-01. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Ekspresi](#).

- Ekspresi sekarang mendukung `logn`, `atan2`, dan `haversin` fungsi serta variabel `_score` (skor relevansi teks) dan `_time` (waktu zaman). Jika Anda menyimpan lokasi di `latlon` bidang, Anda dapat mereferensikan nilai lintang dan bujur sebagai `FIELD.latitude` dan `FIELD.longitude`. Anda juga dapat mereferensikan keduanya `int` dan `double` bidang dalam ekspresi. Fungsi-fungsi berikut tidak lagi didukung: `cs.text_relevance`, `erf`, `lgamma`, `rand`, dan `time`. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Ekspresi](#).

Untuk informasi selengkapnya tentang mengonfigurasi opsi pengindeksan untuk domain 2013-01-01, lihat [configure indexing options](#). Untuk informasi selengkapnya tentang mengonfigurasi opsi ketersediaan, opsi penskalaan, opsi pemrosesan teks, pemberi saran, dan ekspresi, lihat [Membuat dan Mengelola Domain Pencarian](#).

## Tindakan dan Opsi Layanan CloudSearch Konfigurasi Amazon Baru

Tindakan berikut telah ditambahkan ke API Layanan Konfigurasi 2013-01-01:

- `DefineAnalysisScheme`
- `DefineExpression`
- `DefineSuggester`
- `DeleteAnalysisScheme`
- `DeleteExpression`
- `DeleteSuggester`
- `DescribeAnalysisSchemes`
- `DescribeAvailabilityOptions`
- `DescribeExpressions`
- `DescribeScalingParameters`
- `DescribeSuggesters`
- `ListDomainNames`
- `UpdateAvailabilityOptions`
- `UpdateScalingParameters`

`deployedOpsi` telah ditambahkan ke tindakan deskripsi untuk bidang indeks, kebijakan akses, dan pemberi saran. Setel `deployed` opsi ke `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda.

## Tindakan dan Opsi Layanan CloudSearch Konfigurasi Amazon yang Usang

Tindakan berikut tidak didukung di API Layanan Konfigurasi 2013-01-01:

- DefineRankExpression
- DescribeRankExpression
- DeleteRankExpression
- DescribeDefaultSearchField
- DescribeStemmingOptions
- DescribeStopwordOptions
- DescribeSynonymOptions
- UpdateDefaultSearchField
- UpdateStemmingOptions
- UpdateStopwordOptions
- UpdateSynonymOptions

## Mengunggah Data ke 2013-01-01 Domain Amazon CloudSearch

Dengan API 2013-01-01, Anda tidak perlu lagi menentukan versi dokumen—pembaruan diterapkan sesuai urutan penerimaannya. Anda juga tidak lagi menentukan `lang` atribut untuk setiap dokumen—Anda mengontrol pemrosesan teks khusus bahasa dengan mengonfigurasi skema analisis untuk masing-masing dan bidang. `text text-array`

Untuk mengunggah data Anda ke domain 2013-01-01, Anda perlu:

- Hilangkan `lang` atribut `version` dan dari kumpulan dokumen Anda.
- Pastikan semua bidang dokumen sesuai dengan bidang indeks yang dikonfigurasi untuk domain Anda. Bidang yang tidak dikenal tidak lagi diabaikan, mereka akan menghasilkan kesalahan.
- Posting batch dokumen ke titik akhir dokumen domain 2013-01-01 Anda. Perhatikan bahwa Anda harus menentukan versi API 2013-01-01. Misalnya, permintaan berikut memposting batch yang terkandung di `data1.json` `doc-movies-123456789012.us-east-1.cloudsearch.amazonaws.com` titik akhir.

```
curl -X POST --upload-file data1.json doc-movies-123456789012.us-east-1.  
cloudsearch.amazonaws.com/2013-01-01/documents/batch --header "Content-Type:
```

```
application/json"
```

API 2013-01-01 mendukung prescaling domain Anda untuk meningkatkan kapasitas upload. Jika Anda memiliki sejumlah besar data untuk diunggah, konfigurasi opsi penskalaan domain Anda dan pilih jenis instans yang lebih besar yang diinginkan. Pindah ke jenis instans yang lebih besar memungkinkan Anda mengunggah batch secara paralel dan mengurangi waktu yang diperlukan untuk data yang akan diindeks. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

Untuk informasi selengkapnya tentang memformat data Anda, lihat [Mempersiapkan Data Anda](#).

## Mencari 2013-01-01 Domain Amazon CloudSearch

Sebagian besar upaya yang diperlukan untuk memigrasikan domain CloudSearch penelusuran Amazon yang ada ke API 2013-01-01 memperbarui pipeline kueri Anda untuk mengirimkan permintaan penelusuran yang kompatibel dengan 2013-01-01.

- Gunakan versi API 2013-01-01 di semua permintaan.
- Gunakan `q` parameter untuk menentukan kriteria pencarian untuk semua permintaan. `bqParameter` tidak lagi didukung. Untuk menggunakan sintaks pencarian terstruktur (Boolean), tentukan `q.parser=structured` dalam permintaan.
- Parameter tidak dapat diulang dalam permintaan pencarian.
- Karakter wildcard (\*) hanya didukung saat menggunakan parser kueri sederhana. Gunakan prefix operator untuk melakukan pencocokan awalan dengan parser kueri terstruktur. Sebagai contoh, `q=(prefix 'oce')&q.parser=structured`.
- Gunakan nama bidang `_id` untuk mereferensikan bidang ID dokumen dalam permintaan pencarian. Nama `docid` bidang tidak lagi didukung.
- Gunakan range operator mencari bidang untuk nilai dalam rentang yang ditentukan. `filterOperator` tidak lagi didukung.
- Gunakan sintaks rentang baru untuk mencari rentang nilai, termasuk tanggal dan lokasi yang disimpan di `latlon` bidang. Notasi double dot (..) tidak lagi didukung. Pisahkan batas atas dan bawah dengan koma (,), dan lampirkan rentang dalam tanda kurung atau kawat gigi. Braket persegi ([,]) menunjukkan bahwa ikatan disertakan, penjepit keriting ({,}) tidak termasuk batas. Misalnya, `year:2008..2011` sekarang dinyatakan sebagai `year:[2008,2011]`. Rentang terbuka seperti `year:..2011` sekarang dinyatakan sebagai `year:{,2011]`.

- Gunakan term operator untuk mencari bidang untuk nilai tertentu. `fieldOperator` tidak lagi didukung.
- Gunakan `q.options` parameter untuk menentukan bobot bidang. `cs.text_relevance` Fungsi ini tidak lagi didukung. Sebagai contoh, `q.options={fields:['title^2','plot^0.5']}`.
- Gunakan `fq` parameter untuk memfilter hasil tanpa mempengaruhi bagaimana dokumen yang cocok dinilai dan diurutkan.
- Gunakan titik (.) sebagai pemisah daripada tanda hubung (-) dalam parameter awalan: `expr.NAME facet.FIELD highlight.FIELD`
- Gunakan `facet.FIELD` parameter untuk menentukan semua opsi facet. `facet-FIELD-constraints` Parameter `facet-FIELD-top-Nfacet-FIELD-sort`, dan tidak lagi didukung.
- Gunakan `sort` parameter untuk menentukan bidang atau ekspresi yang ingin Anda gunakan untuk menyortir. Anda harus secara eksplisit menentukan arah pengurutan dalam parameter pengurutan. Sebagai contoh, `sort=rank asc, date desc`. `rank` Parameter tidak lagi didukung.
- Gunakan `expr.NAME` untuk menentukan ekspresi dalam permintaan pencarian. `rank-RANKNAME` Parameter tidak lagi didukung.
- Gunakan `format=xml` untuk mendapatkan hasil sebagai XHTML. `result-type` Parameter tidak lagi didukung.

API pencarian 2013-01-01 juga mendukung beberapa fitur baru:

- Penguatan istilah — gunakan `boost` opsi dalam kueri terstruktur untuk meningkatkan pentingnya satu bagian kueri relatif terhadap bagian lainnya. Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).
- Pencarian frase ceroboh—gunakan `near` operator dalam kueri terstruktur untuk mencari `text` atau `text-array` bidang untuk beberapa istilah dan temukan dokumen yang berisi istilah dalam jarak yang ditentukan satu sama lain. Anda juga dapat melakukan pencarian frase ceroboh dengan parser kueri sederhana dengan menambahkan `~` operator dan nilai ke frasa. Untuk informasi selengkapnya, lihat [Mencari Frase](#).
- Pencarian fuzzy—gunakan `~` operator untuk melakukan pencarian fuzzy dengan parser kueri sederhana. Tambahkan `~` operator dan nilai ke istilah untuk menunjukkan berapa banyak istilah yang dapat berbeda dan masih dianggap cocok. Untuk informasi selengkapnya, lihat [Mencari Persyaratan Individu](#).
- Menyorot—Gunakan `highlight.FIELD` parameter untuk menyorot kecocokan di bidang tertentu. Untuk informasi selengkapnya, lihat [Menyoroti Pencarian Hits di Amazon CloudSearch](#).

- Pelengkapan otomatis—Konfigurasi dan kirimkan permintaan ke `suggester` sumber daya untuk mendapatkan daftar penyelesaian kueri dan dokumen tempat mereka ditemukan. Untuk informasi selengkapnya, lihat [Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch](#).
- Hasil pencarian parsial—gunakan `partial=true` parameter untuk mengambil sebagian hasil ketika satu atau beberapa partisi indeks tidak tersedia. Secara default Amazon CloudSearch hanya mengembalikan hasil jika setiap partisi dapat ditanyakan.
- Deep paging—gunakan `cursor` parameter untuk membuat paginasi hasil ketika Anda memiliki kumpulan hasil yang besar. Untuk informasi selengkapnya, lihat [Paginate the results](#).
- Cocokkan semua dokumen—gunakan operator kueri `matchall` terstruktur untuk mengambil semua dokumen dalam indeks.
- Parser kueri baru—gunakan `q.parser` parameter untuk memilih Lucene atau DisMax parser alih-alih parser sederhana atau terstruktur, atau. `q.parser=lucene` `q.parser=dismax`

Anda juga akan melihat beberapa perubahan perilaku saat mencari:

- String tidak lagi diberi token pada batas kasus dan periode yang tidak diikuti oleh spasi dianggap sebagai bagian dari istilah. Untuk informasi selengkapnya, lihat [Pemrosesan Teks di Amazon CloudSearch](#).
- Bidang literal sekarang peka huruf besar/kecil.
- Respons pencarian tidak lagi mencakup peringkat, ekspresi kecocokan, atau waktu CPU. Satu-satunya informasi status yang dikembalikan adalah ID sumber daya (`rid`) dan waktu pemrosesan (`waktu-ms`).
- Ketika Anda mendapatkan informasi facet untuk `int` bidang, `min` dan `max` nilai tidak lagi dikembalikan.

Untuk informasi selengkapnya tentang mencari data Anda, lihat [Mencari Data Anda dengan Amazon CloudSearch](#) dan [Search API](#).

## Parameter dan Opsi Baru di Amazon CloudSearch 2013-01-01 Search API

Parameter berikut telah ditambahkan ke API Pencarian 2013-01-01:

- `cursor.FIELD`
- `expr.NAME`



- facet.FIELD
- format
- fq
- highlight.FIELD
- partial
- pretty
- q.options
- q.parser
- return
- sort

~Operator telah ditambahkan ke bahasa kueri sederhana untuk mendukung pencarian fuzzy dan pencarian frase yang ceroboh.

Operator berikut telah ditambahkan ke bahasa kueri terstruktur:

- boost
- matchall
- near
- phrase
- prefix
- range
- term

## Parameter dan Opsi CloudSearch Pencarian Amazon yang Usang

Parameter berikut tidak lagi didukung di API pencarian 2013-01-01:

- bq
- Segi-bidang-TOP-N
- Facet-field-sort
- Kendala aspek-bidang
- peringkat

- Rank-rankName
- bidang kembali
- tipe hasil
- Bidang T

Operator dan pintasan berikut tidak lagi didukung dalam kueri terstruktur:

- bidang
- filter
- -
- |
- +
- \*

## Batas Diperbarui di Amazon CloudSearch 2013-01-01

Tabel ini merangkum perubahan dan penambahan batas Amazon CloudSearch . Untuk daftar lengkap CloudSearch batas Amazon, lihat [Batas](#).

Perubahan	Ringkasan
Nama yang dipesan	Skor adalah satu-satunya nama yang dicadangkan.
Tidak ada batasan data pengembalian	Data yang dikembalikan dari bidang teks tidak lagi terpotong pada 2 KB. Namun, perlu diingat bahwa ukuran dokumen maksimum adalah 1 MB.
Tidak ada batasan pada stemming, stopword, atau kamus sinonim.	Kamus stemming, stopword, dan sinonim dikonfigurasi dalam skema analisis dan tidak ada batasan ukuran definisi skema analisis.
Jumlah maksimum nilai bidang	Bidang tipe array dapat berisi hingga 1000 nilai.
Ukuran bidang	Ukuran maksimum <code>literal</code> bidang adalah 4096 poin kode Unicode.

Perubahan	Ringkasan
Rentang bidang <code>int</code>	<code>intBidang</code> dapat berisi nilai dalam kisaran -9.223.372.036,854.775.808 - 9.223.372.036.854.775.807 (inklusif).
Jumlah maksimum sorotan	Jumlah maksimum kemunculan istilah pencarian yang dapat disorot adalah 5.
Jumlah maksimum sugester	Jumlah maksimum sugester yang dapat Anda konfigurasi untuk domain adalah 10.
Jumlah maksimum hit yang dapat Anda ambil sekaligus	Jumlah maksimum hit yang dapat Anda ambil sekaligus adalah 10.000. <code>sizeParameter</code> dapat berisi nilai dalam kisaran 0 - 10000.

# Membuat dan Mengelola CloudSearch Domain Amazon

Domain pencarian merangkum data yang ingin Anda cari, opsi pengindeksan yang mengontrol cara Anda dapat mencari data dan informasi apa yang dapat Anda ambil dari domain pencarian Anda, dan contoh pencarian yang mengindeks data Anda dan memproses permintaan pencarian. Anda dapat [membuat](#), [memantau](#), dan [menghapus](#) domain menggunakan CloudSearch konsol AmazonAWS CLI, atau AWS SDK. Semua tindakan manajemen domain diimplementasikan oleh layanan CloudSearch konfigurasi Amazon. Lihat informasi yang lebih lengkap di [Referensi API Konfigurasi untuk Amazon CloudSearch](#).

## Topik

- [Membuat CloudSearch Domain Amazon](#)
- [Mengkonfigurasi Akses untuk Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi Ketersediaan di Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi Titik Akhir Domain di Amazon CloudSearch](#)
- [Memantau CloudSearch Domain Amazon](#)
- [Menghapus Domain Amazon CloudSearch](#)
- [Menandai Domain Amazon CloudSearch](#)

## Membuat CloudSearch Domain Amazon

Untuk mencari data Anda dengan Amazon CloudSearch, hal pertama yang perlu Anda lakukan adalah membuat domain pencarian. Jika Anda memiliki beberapa koleksi data yang ingin dicari, Anda dapat membuat beberapa domain penelusuran. Sebelum Anda dapat [mengirim permintaan pencarian](#) ke domain baru, Anda juga harus [mengonfigurasi kebijakan akses](#), [mengonfigurasi bidang indeks](#), dan [mengunggah data yang ingin Anda cari](#).

Saat Anda membuat domain pencarian, Anda harus memberinya nama yang unik. Nama domain harus dimulai dengan huruf atau angka dan setidaknya 3 dan tidak lebih dari 28 karakter. Karakter yang diizinkan adalah: a-z, 0-9, dan tanda hubung (-). Huruf besar, garis bawah (\_), dan karakter khusus lainnya tidak diperbolehkan dalam nama domain.

Secara default, semua domain baru dibuat menggunakan versi API 2013-01-01. Jika sebelumnya Anda telah membuat domain penelusuran dengan versi API 2011-02-01, Anda dapat memilih untuk

menggunakan API lama untuk domain baru Anda. Namun, sebaiknya gunakan API 2013-01-01 untuk semua kasus penggunaan baru. Semua domain harus bermigrasi ke API 2013-01-01 saat API 2011-02-01 dihentikan.

Anda dapat memilih wilayah AWS tempat Anda ingin membuat domain pencarian. Biasanya, Anda harus memilih wilayah yang paling dekat dengan operasi Anda. Misalnya, jika Anda tinggal di Eropa, buat domain pencarian Anda di wilayah Eropa (Irlandia) (eu-barat-1). Untuk daftar wilayah dan titik akhir yang didukung saat ini, lihat [Wilayah dan Titik Akhir](#). Untuk informasi selengkapnya tentang memilih wilayah, lihat [Wilayah dan Titik Akhir untuk Amazon CloudSearch](#).

#### Note

CloudSearch Domain Amazon di berbagai wilayah sepenuhnya independen. Misalnya, jika Anda membuat domain pencarian yang disebut my-domain di us-east-1, dan domain lain yang disebut my-domain di eu-west-1, mereka sepenuhnya independen dan tidak berbagi data apa pun.

Setiap domain pencarian memiliki titik akhir unik di mana Anda mengunggah data untuk pengindeksan dan mengirimkan permintaan pencarian. Dokumen domain dan titik akhir pencarian tetap sama untuk masa pakai domain. Misalnya, titik akhir untuk domain yang disebut imdb-movies mungkin:

```
doc-imdb-movies-nypdffbzrzkoudsurkxvbwpi4.us-east-1.cloudsearch.amazonaws.com  
search-imdb-movies-nypdffbzrzkoudsurkxvbwpi4.us-east-1.cloudsearch.amazonaws.com
```

#### Important

Secara default, akses ke dokumen domain baru dan titik akhir pencarian diblokir untuk semua alamat IP. Anda harus mengonfigurasi kebijakan akses untuk domain agar dapat mengirimkan permintaan pencarian ke titik akhir pencarian domain dan mengunggah data dari baris perintah atau melalui titik akhir dokumen domain. Anda dapat mengunggah dokumen dan mencari domain melalui CloudSearch konsol Amazon tanpa mengonfigurasi kebijakan akses.

Anda dapat membuat domain pencarian dari [Amazon CloudSearch console](#), menggunakan `aws cloudsearch create-domain` perintah, atau menggunakan salah satu AWS SDK.

## Topik

- [Membuat Domain Menggunakan CloudSearch Konsol Amazon](#)
- [Membuat Domain Menggunakan AWS CLI](#)
- [Membuat CloudSearch Domain Amazon Menggunakan AWS SDK](#)

## Membuat Domain Menggunakan CloudSearch Konsol Amazon

CloudSearch Konsol Amazon memungkinkan Anda membuat domain pencarian baru dengan mudah dan menyediakan berbagai opsi untuk mengonfigurasi opsi pengindeksan.

Untuk membuat domain

1. Masuk ke AWS Management Console dan buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Pilih Create domain (Buat domain).
3. Masukkan nama untuk domain baru Anda. Nama domain harus dimulai dengan huruf atau angka dan setidaknya 3 dan tidak lebih dari 28 karakter. Nama domain dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung). Huruf besar, garis bawah (\_), dan karakter khusus lainnya tidak diperbolehkan dalam nama domain.

Secara opsional, Anda dapat mengatur jenis instans yang diinginkan dan jumlah replikasi yang diinginkan untuk menskalakan domain Anda. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

4. Pilih Berikutnya.
5. Dalam opsi konfigurasi, pilih Konfigurasi manual dan pilih Berikutnya.
6. Konfigurasi bidang indeks untuk domain. Untuk petunjuk, lihat [configure indexing options](#).
7. Pilih Berikutnya.
8. Konfigurasi kebijakan akses domain. Untuk petunjuk, lihat [configure access policies](#).

### Note

Sampai Anda mengonfigurasi kebijakan akses, Anda hanya dapat mengunggah dokumen dan mengirimkan kueri pencarian melalui konsol. Secara default, dokumen dan titik akhir pencarian dikonfigurasi untuk memblokir semua alamat IP.

9. Pilih Berikutnya.

## 10. Tinjau konfigurasi domain dan pilih Buat.

Dokumen domain dan titik akhir layanan pencarian ditampilkan di dasbor domain saat domain menjadi aktif. Pada saat itu, Anda dapat mengunggah dokumen untuk pengindeksan dan mulai mencari data Anda.

## Membuat Domain Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch create-domain` perintah untuk membuat domain pencarian. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk membuat domain

- Jalankan `aws cloudsearch create-domain` perintah dan tentukan nama domain yang ingin Anda buat dengan `--domain-name` opsi. Misalnya, untuk membuat domain yang disebut film:

### Example

```
aws cloudsearch create-domain --domain-name movies
{
  "DomainStatus": {
    "DomainId": "965407640801/movies",
    "Created": true,
    "Deleted": false,
    "SearchInstanceCount": 0,
    "DomainName": "movies",
    "SearchService": {},
    "RequiresIndexDocuments": false,
    "Processing": false,
    "DocService": {},
    "ARN": "arn:aws:cloudsearch:us-east-1:965407640801:domain/movies",
    "SearchPartitionCount": 0
  }
}
```

`aws cloudsearch create-domain` Perintah segera kembali. Dibutuhkan sekitar sepuluh menit untuk membuat endpoint untuk domain baru. Anda dapat menggunakan `aws cloudsearch describe-domains` perintah untuk melihat ringkasan status dan konfigurasi domain. Untuk informasi selengkapnya, lihat [Mendapatkan Informasi Tentang CloudSearch Domain Amazon](#).

**⚠ Important**

Setelah titik akhir domain aktif, mereka tetap sama selama masa pakai domain. Anda harus men-cache titik akhir — tidak perlu melakukan kueri untuk titik akhir sebelum mengirimkan dokumen atau permintaan layanan pencarian dan melakukannya kemungkinan akan mengakibatkan permintaan Anda dibatasi.

## Membuat CloudSearch Domain Amazon Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [CreateDomain](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Mengkonfigurasi Akses untuk Amazon CloudSearch

Anda menggunakan kebijakan akses AWS Identity and Access Management (IAM) untuk mengontrol akses ke layanan CloudSearch konfigurasi Amazon dan setiap layanan dokumen, pencarian, dan saran domain pencarian. Kebijakan akses IAM adalah dokumen JSON yang secara eksplisit mencantumkan izin yang menentukan tindakan apa yang diizinkan dilakukan orang atau proses. Untuk pengenalan kebijakan akses IAM, lihat [Gambaran Umum Kebijakan AWS IAM](#).

Anda mengontrol akses ke API layanan CloudSearch konfigurasi Amazon dan API layanan domain secara independen. Misalnya, Anda dapat memilih untuk membatasi siapa yang dapat memodifikasi konfigurasi domain produksi Anda, tetapi mengizinkan anggota tim untuk membuat dan mengelola domain mereka sendiri untuk pengembangan dan pengujian. Demikian pula, Anda dapat mengonfigurasi domain pengembangan dan pengujian untuk menerima permintaan anonim ke layanan upload, pencarian, dan saran, tetapi mengunci domain produksi Anda sehingga hanya menerima permintaan yang diautentikasi dari aplikasi Anda.

Ketika AWS menerima permintaan, AWS mengautentikasi bahwa permintaan tersebut berasal dari pengguna AWS yang dikenal, dan kemudian memeriksa kebijakan yang relevan untuk menentukan apakah pengguna berwenang untuk melakukan tindakan yang diminta menggunakan sumber daya yang diminta. Jika pengguna belum secara eksplisit diberikan izin untuk melakukan tindakan, permintaan ditolak. Selama evaluasi kebijakan, jika AWS menemukan penolakan eksplisit, efek penolakan lebih diutamakan daripada efek izin eksplisit apa pun yang berlaku.



**⚠ Important**

Untuk mengaktifkan otentikasi, CloudSearch permintaan Amazon harus ditandatangani dengan kunci akses. Satu-satunya pengecualian adalah jika Anda mengizinkan akses anonim ke layanan upload, pencarian, atau saran domain. Untuk informasi selengkapnya, lihat [Menandatangani Permintaan](#).

**Topik**

- [Menulis Kebijakan Akses untuk Amazon CloudSearch](#)
- [Contoh CloudSearch Kebijakan Amazon](#)
- [Mengonfigurasi Akses untuk Amazon CloudSearch Menggunakan AWS Management Console](#)
- [Mengkonfigurasi Akses untuk Amazon CloudSearch dengan AWS CLI](#)
- [Mengonfigurasi Akses ke Endpoint Domain Menggunakan AWS SDK](#)

## Menulis Kebijakan Akses untuk Amazon CloudSearch

Amazon CloudSearch mendukung kebijakan berbasis pengguna dan kebijakan berbasis sumber daya:

- Kebijakan berbasis pengguna dilampirkan ke peran, grup, atau pengguna IAM tertentu. Kebijakan berbasis pengguna menentukan domain pencarian akun mana yang dapat diakses seseorang atau proses dan tindakan apa yang dapat mereka lakukan. Untuk melampirkan kebijakan berbasis pengguna ke pengguna, grup, atau peran, Anda menggunakan konsol IAM, AWS CLI atau AWS SDK. Anda harus menentukan kebijakan berbasis pengguna untuk mengontrol akses ke tindakan layanan CloudSearch konfigurasi Amazon. (Pengguna dalam konteks ini belum tentu seseorang, itu hanya identitas dengan izin terkait. Misalnya, Anda dapat membuat pengguna untuk mewakili aplikasi yang perlu memiliki kredensial untuk mengirimkan permintaan penelusuran ke domain Anda.)
- Kebijakan berbasis sumber daya untuk Amazon CloudSearch dilampirkan ke domain pencarian tertentu. Kebijakan berbasis sumber daya menentukan siapa yang memiliki akses ke domain pencarian dan layanan domain mana yang dapat mereka gunakan. Kebijakan berbasis sumber daya hanya mengontrol akses ke dokumen, pencarian, dan layanan saran domain tertentu; kebijakan tersebut tidak dapat digunakan untuk mengonfigurasi akses ke tindakan layanan konfigurasi Amazon CloudSearch . Untuk melampirkan kebijakan berbasis sumber daya ke domain, Anda menggunakan konsol Amazon CloudSearch , atau AWS CLI AWS SDK.

Secara umum, kami menyarankan untuk mengelola akses ke Amazon CloudSearch API dengan mengonfigurasi kebijakan berbasis pengguna. Ini memungkinkan Anda untuk mengelola semua izin Anda di satu tempat dan setiap perubahan yang perlu Anda lakukan segera berlaku. Namun, untuk mengizinkan akses publik ke layanan pencarian domain atau membatasi akses berdasarkan alamat IP, Anda harus mengonfigurasi kebijakan berbasis sumber daya untuk domain tersebut. (Sebaiknya ganti kebijakan akses berbasis IP lama Anda dengan kebijakan berbasis pengguna secepatnya.) Anda juga dapat menggunakan kebijakan berbasis sumber daya untuk mengizinkan akun lain mengakses domain dengan mudah. Perlu diingat bahwa memproses perubahan pada kebijakan berbasis sumber daya domain membutuhkan waktu lebih lama daripada menerapkan perubahan pada kebijakan berbasis pengguna.

Konsol IAM dapat membantu Anda menulis kebijakan berbasis pengguna dan sumber daya untuk Amazon CloudSearch. Untuk informasi selengkapnya, lihat [Mengelola Kebijakan IAM](#).

## Isi Kebijakan Akses untuk Amazon CloudSearch

Anda menentukan informasi berikut dalam kebijakan akses Anda untuk Amazon CloudSearch:

- **Version** menentukan versi bahasa kebijakan yang kompatibel dengan pernyataan tersebut. Versi selalu diatur ke 2012-10-17.
- **Resource** adalah ARN (Nama Sumber Daya Amazon) untuk domain tempat kebijakan berbasis pengguna berlaku. **Resource** tidak ditentukan dalam kebijakan berbasis sumber daya yang dikonfigurasi melalui layanan CloudSearch konfigurasi Amazon, karena kebijakan dilampirkan langsung ke sumber daya. Untuk informasi selengkapnya tentang Amazon CloudSearch ARN, lihat [Amazon CloudSearch ARN](#).
- **Effect** menentukan apakah pernyataan mengotorisasi atau memblokir akses ke tindakan tertentu. Itu harus **Allow** atau **Deny**.
- **SideEffect** adalah string opsional yang dapat Anda gunakan untuk memberikan nama deskriptif untuk pernyataan kebijakan.
- **Action** menentukan CloudSearch tindakan Amazon mana yang berlaku untuk pernyataan tersebut. Untuk tindakan yang didukung, lihat [CloudSearch Tindakan Amazon](#). Anda dapat menggunakan wildcard (\*) sebagai tindakan untuk mengonfigurasi akses untuk semua tindakan saat Anda perlu memberikan akses administratif untuk memilih pengguna. (Dalam hal ini, Anda mungkin juga ingin mengaktifkan otorisasi multi-faktor untuk keamanan tambahan. Untuk informasi selengkapnya, lihat [Mengonfigurasi Akses API yang dilindungi MFA](#).) Wildcard juga didukung dalam nama tindakan. Misalnya, "Action": ["cloudsearch:Describe\*"]

cocok dengan semua Describe tindakan layanan konfigurasi, seperti DescribeDomains dan DescribeServiceAccessPolicies.

- **Condition** menentukan kondisi kapan kebijakan berlaku. Saat mengonfigurasi akses berbasis IP anonim, Anda akan menentukan alamat IP yang diterapkan aturan akses, misalnya. 

```
"IpAddress": {"aws:SourceIp": ["192.0.2.0/32"]}
```
- **Principal** menentukan siapa yang diizinkan mengakses domain dalam kebijakan berbasis sumber daya. **Principal** tidak ditentukan dalam kebijakan berbasis pengguna yang dikonfigurasi melalui IAM. **Principal** Nilai untuk kebijakan berbasis sumber daya dapat menentukan akun AWS atau pengguna lain di akun Anda sendiri. Misalnya, untuk memberikan akses ke akun 555555555555, Anda akan menentukan. 

```
"Principal":{"AWS":["arn:aws:iam::555555555555:root"]}
```

 Menentukan wildcard (\*) memungkinkan akses anonim ke domain. Akses anonim tidak disarankan. Jika Anda mengaktifkan akses anonim, Anda setidaknya harus menentukan kondisi untuk membatasi alamat IP mana yang dapat mengirimkan permintaan ke domain. Untuk informasi selengkapnya, lihat [Memberikan Akses ke Domain dari Alamat IP Terpilih](#).

Untuk contoh kebijakan akses untuk Amazon CloudSearch, lihat [Contoh CloudSearch Kebijakan Amazon](#).

## Amazon CloudSearch ARN

Nama Sumber Daya Amazon (ARN) kebijakan secara unik menentukan domain tempat kebijakan tersebut berlaku. ARN adalah format standar yang digunakan AWS untuk mengidentifikasi sumber daya. Nomor 12 digit di ARN adalah ID akun AWS Anda. Amazon CloudSearch ARN adalah dari bentuk `arn:aws:cloudsearch:REGION:ACCOUNT-ID:domain/DOMAIN-NAME`.

Daftar berikut menjelaskan elemen variabel dalam ARN:

- **REGION** adalah wilayah AWS tempat CloudSearch domain Amazon tempat Anda mengonfigurasi izin berada. Anda dapat menggunakan wildcard (\*) **REGION** untuk semua wilayah.
- **ACCOUNT-ID** adalah ID akun AWS Anda tanpa tanda hubung; misalnya, 111122223333.
- **DOMAIN-NAME** mengidentifikasi domain pencarian tertentu. Anda dapat menggunakan wildcard (\*) **DOMAIN-NAME** untuk semua domain akun Anda di wilayah yang ditentukan. Jika Anda memiliki beberapa domain yang namanya dimulai dengan awalan yang sama, Anda dapat menggunakan wildcard untuk mencocokkan semua domain tersebut. Misalnya, `dev-*` korek api `dev-testdev-movies`, `dev-sandbox`, dan sebagainya. Perhatikan bahwa jika Anda memberi nama domain baru dengan awalan yang sama, kebijakan ini juga berlaku untuk domain baru tersebut.

Misalnya, ARN berikut mengidentifikasi `movies` domain di `us-east-1` wilayah yang dimiliki oleh akun `111122223333`:

```
arn:aws:cloudsearch:us-east-1:111122223333:domain/movies
```

Contoh berikut menunjukkan bagaimana ARN digunakan untuk menentukan sumber daya dalam kebijakan berbasis pengguna.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:search"],
      "Resource": "arn:aws:cloudsearch:us-east-1:111122223333:domain/movies"
    }
  ]
}
```

ARN domain ditampilkan di dasbor domain di CloudSearch konsol Amazon dan juga tersedia dengan menelepon `DescribeDomains`

#### Important

Saat menentukan ARN untuk domain yang dibuat dengan API 2011-02-01, Anda harus menggunakan nama layanan Amazon sebelumnya, `CloudSearch cs`. Misalnya, `arn:aws:cs:us-east-1:111122223333:domain/movies`. Jika Anda perlu menentukan kebijakan yang mengonfigurasi akses untuk domain 2011 dan 2013, pastikan untuk menentukan format ARN yang benar untuk setiap domain. Untuk informasi selengkapnya, lihat [Konfigurasi Kebijakan Akses Layanan Tidak Berfungsi](#).

## CloudSearch Tindakan Amazon

Tindakan yang Anda tentukan mengontrol CloudSearch API Amazon mana yang berlaku untuk pernyataan tersebut. Semua CloudSearch tindakan Amazon diawali dengan `cloudsearch:`, seperti `cloudsearch:search`. Daftar berikut menunjukkan tindakan yang didukung:

- `cloudsearch:document` memungkinkan akses ke API layanan dokumen. Izin untuk menggunakan `document` tindakan diperlukan untuk mengunggah dokumen ke domain pencarian untuk pengindeksan.
- `cloudsearch:search` memungkinkan akses ke API pencarian. Izin untuk menggunakan `search` tindakan diperlukan untuk mengirimkan permintaan pencarian ke domain.
- `cloudsearch:suggest` memungkinkan akses ke API yang disarankan. Izin untuk menggunakan `suggest` tindakan diperlukan untuk mendapatkan saran dari domain.
- `cloudsearch:CONFIGURATION-ACTION` memungkinkan akses ke tindakan layanan konfigurasi yang ditentukan. Izin untuk menggunakan tindakan `ListDomainNames` konfigurasi `DescribeDomains` dan diperlukan untuk mengakses CloudSearch konsol Amazon. Tindakan konfigurasi hanya dapat ditentukan dalam kebijakan berbasis pengguna. Untuk daftar lengkap tindakan, lihat [Tindakan](#).

## Contoh CloudSearch Kebijakan Amazon

Bagian ini menyajikan beberapa contoh kebijakan CloudSearch akses Amazon.

### Topik

- [Memberikan Akses Hanya Baca ke Layanan Konfigurasi Amazon CloudSearch](#)
- [Memberikan Akses ke Semua Tindakan Layanan CloudSearch Konfigurasi Amazon](#)
- [Memberikan Akses Tanpa Batas ke Semua Layanan Amazon CloudSearch](#)
- [Memberikan Izin untuk Mengunggah Dokumen ke Domain Amazon CloudSearch](#)
- [Memberikan Amazon CloudSearch Akses ke Akun AWS Lain](#)
- [Memberikan Akses ke CloudSearch Domain Amazon dari Alamat IP Terpilih](#)
- [Memberikan Akses Publik ke Layanan Pencarian CloudSearch Domain Amazon](#)

### Memberikan Akses Hanya Baca ke Layanan Konfigurasi Amazon CloudSearch

Anda dapat memberikan akses hanya-baca ke layanan konfigurasi dengan hanya mengizinkan tindakan berikut. Ini mungkin berguna jika Anda ingin mengizinkan pengguna untuk melihat konfigurasi domain produksi tanpa dapat membuat perubahan.

- `cloudsearch:DescribeAnalysisSchemes`
- `cloudsearch:DescribeAvailabilityOptions`

- `cloudsearch:DescribeDomains`
- `cloudsearch:DescribeExpressions`
- `cloudsearch:DescribeIndexFields`
- `cloudsearch:DescribeScalingParameters`
- `cloudsearch:DescribeServiceAccessPolicies`
- `cloudsearch:DescribeSuggesters`
- `cloudsearch:ListDomainNames`

Kebijakan berbasis pengguna berikut memberikan akses hanya-baca ke layanan konfigurasi untuk `movies` domain yang dimiliki oleh akun `555555555555`. Kebijakan menggunakan wildcard untuk tindakan, karena memberikan akses ke semua tindakan yang dimulai dengan `Deskripsikan` atau `Daftar`. Perhatikan bahwa ini juga akan memberikan akses ke tindakan deskripsi atau daftar apa pun yang mungkin ditambahkan ke API di masa mendatang.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:Describe*",
                "cloudsearch:List*"],
      "Resource": "arn:aws:cloudsearch:us-east-1:555555555555:domain/movies"
    }
  ]
}
```

## Memberikan Akses ke Semua Tindakan Layanan CloudSearch Konfigurasi Amazon

Anda dapat memberikan akses ke semua tindakan layanan CloudSearch konfigurasi Amazon dengan menyertakan `Allow` pernyataan yang memberikan akses ke semua tindakan layanan konfigurasi, tetapi bukan tindakan layanan domain. Ini memungkinkan Anda untuk memberikan akses administratif tanpa mengizinkan pengguna untuk mengunggah atau mengambil data dari domain. Salah satu cara untuk melakukannya adalah dengan menggunakan wildcard untuk memberikan akses ke semua CloudSearch tindakan Amazon, dan kemudian menyertakan pernyataan penolakan yang memblokir akses ke tindakan layanan domain. Kebijakan berbasis pengguna berikut memberikan akses ke layanan konfigurasi untuk semua domain yang dimiliki oleh akun `111122223333` di wilayah tersebut. `us-west-2`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:*"],
      "Resource": "arn:aws:cloudsearch:us-west-2:111122223333:domain/*"
    },
    {
      "Effect": "Deny",
      "Action": ["cloudsearch:document",
        "cloudsearch:search",
        "cloudsearch:suggest"],
      "Resource": "arn:aws:cloudsearch:us-west-2:111122223333:domain/*"
    }
  ]
}
```

## Memberikan Akses Tanpa Batas ke Semua Layanan Amazon CloudSearch

Anda dapat memberikan akses tak terbatas ke semua CloudSearch layanan Amazon, termasuk semua tindakan layanan konfigurasi dan semua layanan domain dengan kebijakan berbasis pengguna. Untuk melakukan ini, Anda menentukan wildcard untuk tindakan, wilayah, dan nama domain. Kebijakan berikut memungkinkan pengguna untuk mengakses semua CloudSearch tindakan Amazon untuk domain apa pun di wilayah mana pun yang dimiliki oleh akun 111122223333.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:*"],
      "Resource": "arn:aws:cloudsearch:*:111122223333:domain/*"
    }
  ]
}
```

## Memberikan Izin untuk Mengunggah Dokumen ke Domain Amazon CloudSearch

Anda dapat memberikan izin kepada pengguna untuk mengunggah dokumen ke domain penelusuran dengan menentukan `cloudsearch:document` tindakan. Misalnya, kebijakan berbasis pengguna

berikut memungkinkan pengguna untuk mengunggah dokumen ke movies domain yang us-east-1 dimiliki oleh akun 111122223333.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:document"],
      "Resource": "arn:aws:cloudsearch:us-east-1:111122223333:domain/movies"
    }
  ]
}
```

## Memberikan Amazon CloudSearch Akses ke Akun AWS Lain

Anda memiliki dua opsi untuk mengonfigurasi akses lintas akun untuk CloudSearch domain:

Opsi	Deskripsi
Konfigurasi peran IAM untuk akses lintas akun.	Peningkatan keamanan, tetapi membutuhkan penandatanganan permintaan yang kompleks. Untuk informasi selengkapnya, lihat <a href="#">Akses API Lintas Akun Menggunakan Peran IAM</a> dalam dokumentasi IAM.
Lampirkan kebijakan berbasis sumber daya ke CloudSearch domain dan lampirkan kebijakan terkelola berbasis pengguna ke peran IAM.	Lebih mudah diimplementasikan. Untuk informasi selengkapnya, lihat <a href="#">Membuat Peran untuk Mendelegasikan Izin kepada Pengguna IAM dan Panduan: Mendelegasikan Akses di Seluruh Akun AWS Untuk Akun yang Anda Miliki Menggunakan Peran IAM dalam dokumentasi IAM</a> .

Topik ini memberikan contoh opsi kedua, menambahkan kebijakan berbasis sumber daya ke domain. CloudSearch Asumsikan bahwa akun #1 dimiliki oleh id akun 111111111111 dan akun #2 dimiliki oleh id akun999999999999. Akun #1 ingin memberikan akses ke akun #2 untuk menggunakan layanan pencarian movies domain, yang memerlukan dua langkah:

1. Akun #1 melampirkan kebijakan berbasis sumber daya ke domain menggunakan CloudSearch konsol Amazon yang memberikan akses ke akun #2.



```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "search_only",
      "Effect": "Allow",
      "Action": ["cloudsearch:search"],
      "Principal": {"AWS": ["arn:aws:iam::999999999999:root"]}
    }
  ]
}
```

2. Akun #2 melampirkan kebijakan terkelola berbasis pengguna ke peran IAM yang dimiliki oleh akun tersebut menggunakan konsol IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["cloudsearch:search"],
      "Resource": "arn:aws:cloudsearch:us-east-1:111111111111:domain/movies"
    }
  ]
}
```

### Important

Untuk mengonfigurasi kebijakan berbasis sumber daya untuk Amazon CloudSearch, Anda harus memiliki izin untuk menggunakan tindakan tersebut. `cloudsearch:UpdateServiceAccessPolicies`

## Memberikan Akses ke CloudSearch Domain Amazon dari Alamat IP Terpilih

Kebijakan akses berbasis sumber daya yang ditetapkan melalui layanan CloudSearch konfigurasi Amazon mendukung akses anonim, yang memungkinkan Anda mengirimkan permintaan yang tidak ditandatangani ke layanan domain penelusuran. Untuk mengizinkan akses anonim dari alamat IP

yang dipilih, gunakan wildcard untuk Principal nilai dan tentukan alamat IP yang diizinkan sebagai Condition elemen dalam kebijakan.

### Important

Mengizinkan akses anonim dari alamat IP yang dipilih secara inheren kurang aman daripada memerlukan kredensi pengguna untuk mengakses domain pencarian Anda. Kami menyarankan agar tidak mengizinkan akses anonim meskipun hanya diizinkan dari alamat IP tertentu. Jika saat ini Anda mengizinkan akses anonim, Anda harus memutakhirkan aplikasi Anda untuk mengirimkan permintaan yang ditandatangani dan mengontrol akses dengan mengonfigurasi kebijakan berbasis pengguna atau berbasis sumber daya.

Jika Anda membuat kebijakan berbasis sumber daya yang memberikan akses ke permintaan yang berasal dari instans Amazon EC2, Anda harus menentukan alamat IP publik instans tersebut.

Alamat IP ditentukan dalam format Classless Inter-Domain Routing (CIDR) standar. Misalnya 10.24.34.0/24 menentukan rentang 10.24.34.0 - 10.24.34.255, sedangkan 10.24.34.0/32 menentukan alamat IP tunggal 10.24.34.0. Untuk informasi selengkapnya tentang notasi CIDR, lihat [RFC 4632](#).

Misalnya, kebijakan berikut memberikan akses ke tindakan pencarian untuk movies domain yang dimiliki oleh akun AWS 111122223333 dari alamat IP 192.0.2.0/32.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "search_only",
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["cloudsearch:search"],
      "Condition": {"IpAddress": {"aws:SourceIp": "192.0.2.0/32"}}
    }
  ]
}
```

## Memberikan Akses Publik ke Layanan Pencarian CloudSearch Domain Amazon

Jika Anda perlu mengizinkan akses publik ke titik akhir penelusuran domain, Anda dapat mengonfigurasi kebijakan berbasis sumber daya tanpa syarat. Ini memungkinkan permintaan yang tidak ditandatangani untuk dikirim dari alamat IP apa pun.

### Important

Mengizinkan akses publik ke domain pencarian berarti Anda tidak memiliki kendali atas volume permintaan yang dikirimkan ke domain. Pengguna jahat dapat membanjiri domain dengan permintaan, memengaruhi pengguna yang sah serta biaya pengoperasian Anda.

Misalnya, kebijakan berikut memberikan akses publik ke tindakan penelusuran untuk movies domain yang dimiliki oleh akun AWS 111122223333.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "public_search",
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["cloudsearch:search"]
    }
  ]
}
```

## Mengonfigurasi Akses untuk Amazon CloudSearch Menggunakan AWS Management Console

Untuk mengonfigurasi kebijakan berbasis pengguna

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Konfigurasi CloudSearch izin Amazon dengan melampirkan kebijakan ke pengguna, grup, atau peran. Untuk informasi selengkapnya, lihat [Mengelola Kebijakan \(AWS Management Console\)](#). Untuk informasi selengkapnya tentang kebijakan berbasis pengguna untuk Amazon CloudSearch, lihat [Menulis Kebijakan Akses untuk Amazon CloudSearch](#)

Untuk mengonfigurasi kebijakan berbasis sumber daya

1. Masuk ke AWS Management Console dan buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Pilih nama domain yang ingin Anda konfigurasikan.
3. Pada tab Konfigurasi domain, pilih Edit di samping Kebijakan akses.
4. Setelah selesai membuat perubahan pada kebijakan akses domain, pilih Kirim.

Domain Anda tetap dalam Processing keadaan sementara Amazon CloudSearch memperbarui kebijakan akses.

## Mengonfigurasi Akses untuk Amazon CloudSearch dengan AWS CLI

Anda dapat mengonfigurasi kebijakan berbasis pengguna dan kebijakan berbasis sumber daya untuk Amazon dengan CloudSearch AWS CLI Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk mengonfigurasi kebijakan berbasis pengguna

- Konfigurasikan CloudSearch izin Amazon dengan melampirkan kebijakan ke pengguna, grup, atau peran dengan `aws put-user-policy`, `aws put-group-policy`, atau `aws put-role-policy` perintah. Untuk informasi selengkapnya, lihat [Mengelola Kebijakan \(AWS Management Console\)](#). Untuk informasi selengkapnya tentang kebijakan berbasis pengguna untuk Amazon CloudSearch, lihat [Menulis Kebijakan Akses untuk Amazon CloudSearch](#)

Untuk mengonfigurasi kebijakan berbasis sumber daya

- Jalankan `aws cloudsearch update-service-access-policies` perintah dan tentukan kebijakan akses dengan `--access-policies` opsi. Kebijakan akses harus dilampirkan dalam tanda kutip dan semua kutipan dalam kebijakan akses harus diloloskan dengan garis miring terbalik. Untuk informasi selengkapnya tentang kebijakan berbasis sumber daya untuk Amazon, lihat [CloudSearch Menulis Kebijakan Akses untuk Amazon CloudSearch](#)

Contoh berikut mengonfigurasi `movies` domain untuk menerima permintaan pencarian dari alamat `192.0.2.0` IP.

```
aws cloudsearch update-service-access-policies --domain-name movies
--access-policies "{\"Version\":\"2012-10-17\",\"Statement\":[{
```

```

    \"Sid\": \"search_only\",
    \"Effect\": \"Allow\",
    \"Principal\": \"*\",
    \"Action\": \"cloudsearch:search\",
    \"Condition\": {\"IpAddress\": {\"aws:SourceIp\": \"192.0.2.0/32\"}}
  ]}\"
{
  \"AccessPolicies\": {
    \"Status\": {
      \"PendingDeletion\": false,
      \"State\": \"Processing\",
      \"CreationDate\": \"2014-04-30T22:07:30Z\",
      \"UpdateVersion\": 9,
      \"UpdateDate\": \"2014-04-30T22:07:30Z\"
    },
    \"Options\":
    \"[{\"Version\": \"2012-10-17\", \"Statement\": [{\"Sid\": \"\",
      \"Effect\": \"Allow\", \"Principal\": \"*\",
      \"Action\": \"cloudsearch:search\",
      \"Condition\": {\"IpAddress\": {\"aws:SourceIp\":
      \"192.0.2.0/32\"}}}]}]\"
  }
}

```

Memperbarui kebijakan akses berbasis sumber daya membutuhkan waktu untuk diselesaikan. Anda dapat memeriksa status kebijakan dengan `aws cloudsearch describe-service-access-policies` perintah. Setelah kebijakan diterapkan, keadaan kebijakan berubah menjadi `Active`.

Anda dapat mengambil kebijakan domain Anda menggunakan `aws cloudsearch describe-service-access-policies` perintah.

## Mengonfigurasi Akses ke Endpoint Domain Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk.

[UpdateServiceAccessPolicies](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

# Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch

Sebuah domain pencarian memiliki satu atau lebih contoh pencarian, masing-masing dengan jumlah terbatas RAM dan sumber daya CPU untuk mengindeks data dan memproses permintaan. Anda dapat mengonfigurasi opsi penskalaan untuk mengontrol jenis instans yang digunakan, jumlah instance indeks pencarian Anda didistribusikan (jumlah partisi), dan jumlah replika dari setiap partisi indeks (jumlah replikasi). Semua instance untuk domain selalu dari jenis yang sama.

Anda dapat mengonfigurasi jenis instans, jumlah partisi, atau jumlah replikasi yang diinginkan untuk CloudSearch domain Amazon menjadi:

- Meningkatkan kapasitas upload Secara default, semua domain pencarian dimulai pada sebuah `search.small` instance. Anda dapat meningkatkan kapasitas upload dokumen domain Anda dengan mengubah jenis instans yang diinginkan. Jika Anda memiliki sejumlah besar data untuk diunggah — misalnya, ketika Anda awalnya mengisi domain pencarian Anda — Anda dapat memilih jenis instans yang lebih besar untuk meningkatkan jumlah pembaruan yang dapat dikirimkan secara paralel dan mengurangi berapa lama waktu yang dibutuhkan untuk mengindeks data Anda. Jika Anda sudah menggunakan jenis instans terbesar, Anda dapat meningkatkan jumlah partisi yang diinginkan untuk lebih meningkatkan kapasitas unggah. Untuk informasi selengkapnya, lihat [Unggahan Massal](#). Perhatikan bahwa meningkatkan jumlah replikasi yang diinginkan umumnya tidak meningkatkan kapasitas unggah domain.
- Mempercepat permintaan pencarian. Memilih jenis instans yang diinginkan yang lebih besar juga dapat mempercepat permintaan pencarian. Jika Anda telah menyetel permintaan dan masih belum memenuhi target kinerja Anda, coba pilih jenis instans yang lebih besar. Jika Anda sudah menggunakan jenis instans terbesar, Anda dapat meningkatkan jumlah partisi yang diinginkan untuk lebih meningkatkan kinerja kueri. Untuk informasi selengkapnya, lihat [Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch](#).
- Meningkatkan kapasitas pencarian. Secara default, Amazon CloudSearch menggunakan satu instance per partisi indeks. Saat Amazon CloudSearch menskalakan domain secara otomatis, Amazon menambahkan replika berdasarkan sumber daya yang diperlukan untuk memproses lalu lintas kueri. Untuk meningkatkan kapasitas pencarian domain, Anda mengatur jumlah replikasi yang diinginkan. Namun, menerapkan instance tambahan membutuhkan waktu. Jika Anda tahu sebelumnya bahwa Anda akan memerlukan kapasitas tambahan—misalnya, sebelum peluncuran atau pengumuman besar—tambahkan replika sebelumnya untuk memastikan bahwa domain pencarian Anda siap menangani beban.
- Meningkatkan toleransi kesalahan. Meningkatkan jumlah replikasi yang diinginkan juga meningkatkan toleransi kesalahan domain—jika ada masalah dengan salah satu replika, yang

lain akan terus menangani permintaan saat sedang dipulihkan. Namun, perhatikan bahwa replika berada di Availability Zone yang sama. Jika Anda perlu memastikan ketersediaan domain Anda jika terjadi gangguan layanan Availability Zone, Anda harus mengaktifkan opsi MultiAZ. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Ketersediaan](#).

Saat Anda mengatur jenis instans yang diinginkan, jumlah replika yang diinginkan, atau jumlah partisi yang diinginkan, Amazon CloudSearch menskalakan domain Anda seperlunya, tetapi tidak akan pernah menskalakan domain ke jenis instans yang lebih kecil dari jenis yang diinginkan, menggunakan replika yang lebih sedikit daripada jumlah replika yang diinginkan, atau kurangi jumlah partisi di bawah jumlah partisi yang diinginkan.

#### Note

Progresi penskalaan otomatis didasarkan pada ruang disk tipe instans yang tersedia. Tipe `search.small` dan `search.medium` instance memiliki jumlah ruang disk yang sama, jadi keduanya berskalasearch.large.

Anda dapat mengubah opsi penskalaan Anda kapan saja. Jika kebutuhan untuk kapasitas tambahan bersifat sementara, Anda dapat melakukan pra-skala domain Anda dengan menyetel opsi penskalaan dan kemudian mengembalikan perubahan setelah volume unggahan atau kueri Anda kembali ke status stabil domain Anda. Ketika Anda membuat perubahan, Anda perlu mengindeks ulang domain Anda, yang dapat memakan waktu beberapa saat agar perubahan diterapkan. Berapa lama waktu yang dibutuhkan untuk mengindeks ulang tergantung pada jumlah data dalam indeks Anda. Anda dapat memantau status domain untuk menentukan kapan pengindeksan selesai — status berubah dari PEMROSESAN menjadi AKTIF.

#### Topik

- [Memilih Opsi Penskalaan di Amazon CloudSearch](#)
- [Mengonfigurasi Opsi Penskalaan melalui Konsol Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi Penskalaan melalui AWS CLI](#)
- [Mengkonfigurasi Opsi Penskalaan melalui AWS SDK](#)

## Memilih Opsi Penskalaan di Amazon CloudSearch

Saat Anda menetapkan opsi penskalaan untuk domain, Anda membuat trade-off antara biaya dan kinerja—mengubah jenis instans yang diinginkan, jumlah replikasi, dan jumlah partisi dapat secara signifikan memengaruhi biaya menjalankan domain Anda.

Untuk menentukan jenis instans mana yang akan dipilih untuk menangani lalu lintas unggahan, pantau performa unggahan saat Anda meningkatkan laju unggahan. Jika Anda mulai melihat sejumlah besar kesalahan 504 atau 507 sebelum mencapai kecepatan unggah yang diinginkan, pilih jenis instans yang lebih besar. Jika Anda sudah menggunakan jenis instans terbesar, Anda dapat menambah jumlah partisi untuk lebih meningkatkan kapasitas unggah.

Untuk kumpulan data kurang dari 1 GB data atau kurang dari satu juta dokumen 1 KB, contoh pencarian kecil harus cukup. Untuk mengunggah kumpulan data antara 1 GB dan 8 GB, sebaiknya setel jenis instans yang diinginkan `search.large` sebelum Anda mulai mengunggah. Untuk kumpulan data antara 8 GB dan 16 GB, mulailah dengan `file.search.xlarge` Untuk kumpulan data antara 16 GB dan 32 GB, mulailah dengan `file.search.2xlarge` Jika Anda memiliki lebih dari 32 GB untuk diunggah, pilih jenis `search.2xlarge` instans dan tingkatkan jumlah partisi yang diinginkan untuk mengakomodasi kumpulan data Anda. Setiap partisi dapat berisi hingga 32 GB data. Kirim [Permintaan Batas Peningkatan Layanan](#) jika Anda membutuhkan kapasitas unggah lebih besar atau memiliki lebih dari 500 GB untuk diindeks.

Untuk menentukan berapa banyak replika yang Anda butuhkan untuk menangani volume kueri tertentu, lakukan beberapa pengujian menggunakan sampel kueri yang Anda harapkan pada tingkat yang perlu Anda dukung. Perlu diingat bahwa kinerja kueri sangat bergantung pada jenis kueri yang sedang diproses. Secara umum, penelusuran yang mengembalikan volume klik yang besar dan kueri terstruktur yang kompleks lebih intensif sumber daya daripada kueri teks sederhana yang cocok dengan persentase kecil dokumen di domain penelusuran Anda. Jika Anda mengharapkan volume kueri kompleks yang tinggi, pilih jenis instans yang diinginkan yang lebih besar dan tingkatkan jumlah replikasi yang diinginkan.

## Mengonfigurasi Opsi Penskalaan melalui Konsol Amazon CloudSearch

Untuk mengonfigurasi opsi penskalaan domain pencarian

Perhatikan bahwa mengubah jenis instans dan jumlah replikasi yang diinginkan dapat secara signifikan meningkatkan biaya menjalankan domain Anda.

1. Di CloudSearch konsol Amazon, pilih nama domain yang ingin Anda konfigurasi.



2. Pada tab Konfigurasi domain, pilih Edit di samping opsi Penskalaan.
3. Pilih jenis instance dari menu tipe instans yang diinginkan.
4. Pilih jumlah replika yang ingin Anda gunakan dari menu Hitungan replikasi yang diinginkan.
5. Jika Anda memilih jenis `search.2xlarge` instance, konfigurasi jumlah partisi yang diinginkan. Tingkatkan jumlah partisi jika Anda memiliki lebih banyak data untuk diunggah daripada yang akan muat pada satu `search.2xlarge` partisi. Untuk informasi selengkapnya, lihat [Unggahan Massal](#).
6. Pilih Kirim.
7. Setelah Anda selesai membuat perubahan pada konfigurasi domain Anda, pilih Tindakan, Jalankan pengindeksan untuk memperbarui dan menerapkan indeks Anda ke instance baru.

## Mengkonfigurasi Opsi Penskalaan melalui AWS CLI

Anda menggunakan `aws cloudsearch update-scaling-parameters` perintah untuk mengkonfigurasi opsi penskalaan untuk domain pencarian. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk mengkonfigurasi opsi penskalaan domain pencarian

- Jalankan perintah `aws cloudsearch update-scaling-parameters`. Anda dapat menentukan jenis instance yang diinginkan dan jumlah replikasi yang diinginkan. Jika Anda memilih jenis instance terbesar (`search.2xlarge`), Anda juga dapat mengatur jumlah partisi yang diinginkan. Misalnya, perintah berikut menetapkan jenis instance yang diinginkan `search.xlarge` dan jumlah replikasi yang diinginkan menjadi dua. Anda harus menentukan opsi `--domain-name` dan `--scaling-parameters` opsi.

```
aws cloudsearch update-scaling-parameters --domain-name movies --scaling-parameters
DesiredInstanceType=search.xlarge,DesiredReplicationCount=2
{
  "ScalingParameters": {
    "Status": {
      "PendingDeletion": false,
      "State": "RequiresIndexDocuments",
      "CreationDate": "2014-06-25T21:41:21Z",
      "UpdateVersion": 10,
      "UpdateDate": "2014-06-25T21:41:21Z"
    },
    "Options": {
```

```
        "DesiredInstanceType": "search.xlarge",  
        "DesiredReplicationCount": 2  
    }  
}
```

### Important

Saat Anda menentukan `--scaling-parameters`, Amazon CloudSearch memperlakukan opsi yang tidak ditentukan sebagai “reset ke default” daripada “biarkan apa adanya.”

Misalnya, jika Anda menentukan `--scaling-parameters DesiredInstanceType=search.xlarge` dalam perintah dan kemudian `--scaling-parameters DesiredReplicationCount=2` dalam perintah berikutnya, Amazon CloudSearch me-reset `DesiredInstanceType` ke nilai defaultnya selama perintah kedua.

Jika Anda ingin perubahan dari perintah pertama bertahan, Anda harus menentukannya lagi di semua perintah berikutnya: `--scaling-parameters DesiredInstanceType=search.xlarge,DesiredReplicationCount=2`.

Agar perubahan diterapkan, Anda harus memulai pembuatan indeks. Anda dapat membangun kembali indeks dengan menelepon `aws cloudsearch index-documents`.

## Mengkonfigurasi Opsi Penskalaan melalui AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk [UpdateScalingParameters](#). Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Mengkonfigurasi Opsi Ketersediaan di Amazon CloudSearch

Anda dapat memperluas CloudSearch domain Amazon ke Availability Zone tambahan di wilayah yang sama untuk meningkatkan toleransi kesalahan jika terjadi gangguan layanan. Availability Zones adalah lokasi yang terpisah secara fisik dengan infrastruktur independen yang dirancang untuk diisolasi dari kegagalan di Availability Zone lainnya. Untuk informasi selengkapnya, lihat [Wilayah dan Zona Ketersediaan](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Saat Anda mengaktifkan opsi Multi-AZ, Amazon CloudSearch menyediakan dan memelihara instans tambahan untuk domain penelusuran Anda di Availability Zone kedua untuk memastikan ketersediaan tinggi. Jumlah maksimum Availability Zone domain dapat digunakan adalah dua.

Mengaktifkan Multi-AZ tidak memengaruhi titik akhir layanan domain pencarian atau meningkatkan volume data atau lalu lintas yang dapat ditangani oleh domain pencarian Anda. Pembaruan secara otomatis diterapkan ke instance di kedua Availability Zone. Lalu lintas pencarian didistribusikan di semua instance dan instance di salah satu zona mampu menangani beban penuh jika terjadi kegagalan.

Jika ada gangguan layanan Availability Zone atau instans di satu zona menjadi terdegradasi, Amazon CloudSearch merutekan semua lalu lintas ke Availability Zone lainnya. Instans redundan dipulihkan di Availability Zone terpisah tanpa intervensi administratif atau gangguan dalam layanan.

Anda memperluas domain pencarian yang ada ke Availability Zone kedua dengan mengaktifkan opsi Multi-AZ. Demikian pula, Anda dapat mematikan opsi Multi-AZ untuk menurunkan versi domain ke satu Availability Zone. Menghidupkan atau mematikan opsi Multi-AZ membutuhkan waktu sekitar setengah jam.

Anda dapat mengonfigurasi opsi ketersediaan domain melalui CloudSearch konsol Amazon, menggunakan `aws cloudsearch update-availability-options` perintah, atau AWS SDK.

#### Important

Jika domain Anda berjalan pada satu instance penelusuran, mengaktifkan opsi Multi-AZ menambahkan instance penelusuran kedua di zona ketersediaan yang berbeda, yang menggandakan biaya menjalankan domain Anda. Demikian pula, jika indeks Anda dibagi di beberapa partisi, instance baru diterapkan di Availability Zone kedua untuk setiap partisi. Replika tambahan ditambahkan untuk memastikan bahwa Availability Zone memiliki kapasitas yang cukup untuk menangani semua lalu lintas Anda—ketika Multi-AZ diaktifkan, domain Anda akan memiliki setidaknya satu replika dari setiap partisi indeks. Jika Anda mengatur jumlah replika yang diinginkan dan mengaktifkan opsi Multi-AZ, Amazon CloudSearch memastikan bahwa Anda memiliki setidaknya banyak replika yang tersedia secara total di dua zona ketersediaan. Anda dapat memantau jumlah instance yang digunakan untuk domain Anda dari dasbor domain.

## Topik

- [Mengkonfigurasi Opsi Ketersediaan melalui Konsol Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi CloudSearch Ketersediaan Amazon Menggunakan AWS CLI](#)
- [Mengkonfigurasi Opsi Ketersediaan melalui AWS SDK](#)

## Mengkonfigurasi Opsi Ketersediaan melalui Konsol Amazon CloudSearch

Jika domain Anda saat ini menggunakan instance penelusuran tunggal, mengaktifkan Multi-AZ menambahkan instance penelusuran kedua, yang secara signifikan dapat meningkatkan biaya menjalankan domain Anda.

Untuk mengonfigurasi opsi ketersediaan domain pencarian

1. Di dalam CloudSearch konsol Amazon, pilih nama domain Anda.
2. Dalam konfigurasi Domain, pilih Edit di samping opsi Ketersediaan.
3. Aktifkan opsi Toggle Multi-AZ.
4. Pilih Kirim.

## Mengkonfigurasi Opsi CloudSearch Ketersediaan Amazon Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch update-availability-options` perintah untuk mengonfigurasi opsi ketersediaan untuk domain pencarian. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk mengonfigurasi opsi ketersediaan domain pencarian

- Jalankan `aws cloudsearch update-availability-options` perintah dan tentukan `--multi-az` opsi untuk mengaktifkan MultiAZ untuk domain, atau `--no-multi-az` mematikan MultiAZ. Misalnya, permintaan berikut memungkinkan MultiAZ untuk domain: `movies`

```
aws cloudsearch update-availability-options --domain-name movies --multi-az

{
  "AvailabilityOptions": {
    "Status": {
      "PendingDeletion": false,
      "State": "Processing",
    }
  }
}
```

```
        "CreationDate": "2014-04-30T20:42:57Z",
        "UpdateVersion": 13,
        "UpdateDate": "2014-05-01T00:17:45Z"
    },
    "Options": true
}
}
```

## Mengkonfigurasi Opsi Ketersediaan melalui AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [UpdateAvailabilityOptions](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Mengkonfigurasi Opsi Titik Akhir Domain di Amazon CloudSearch

CloudSearch Domain Amazon memungkinkan Anda mengharuskan semua lalu lintas ke domain tiba melalui HTTPS. Fitur keamanan ini membantu Anda memblokir klien yang mengirim permintaan tidak terenkripsi ke domain.

### Topik

- [Mengonfigurasi Opsi Titik Akhir Domain Menggunakan Konsol Amazon CloudSearch](#)
- [Mengkonfigurasi Opsi Endpoint Domain Menggunakan AWS CLI](#)
- [Mengonfigurasi Opsi Titik Akhir Domain Menggunakan AWS SDK](#)

## Mengonfigurasi Opsi Titik Akhir Domain Menggunakan Konsol Amazon CloudSearch

Untuk mengonfigurasi opsi titik akhir domain pencarian

1. Di dalam CloudSearch konsol Amazon, pilih nama domain Anda untuk membuka pengaturannya.
2. Di bawah Konfigurasi domain, pilih Edit di samping opsi HTTPS.
3. Aktifkan opsi Toggle HTTPS.
4. Pilih Kirim.

## Mengkonfigurasi Opsi Endpoint Domain Menggunakan AWS CLI

Gunakan perintah `aws cloudsearch update-domain-endpoint-options`. Untuk informasi selengkapnya, lihat [Referensi Perintah AWS AWS CLI](#).

## Mengonfigurasi Opsi Titik Akhir Domain Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk [the section called “DescribeDomainEndpointOptions”](#) dan [the section called “UpdateDomainEndpointOptions”](#). Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Memantau CloudSearch Domain Amazon

AWS Management Console memungkinkan Anda memantau status dan konfigurasi domain penelusuran dengan mudah dan melihat CloudSearch penggunaan Amazon Anda. Anda juga bisa mendapatkan informasi konfigurasi tentang domain tertentu dengan AWS SDK AWS CLI dan AWS.

### Topik

- [Mendapatkan Informasi Tentang CloudSearch Domain Amazon](#)
- [Memantau CloudSearch Domain Amazon dengan Amazon CloudWatch](#)
- [Mencatat Panggilan API CloudSearch Konfigurasi Amazon dengan AWS CloudTrail](#)
- [Melacak CloudSearch Penggunaan dan Biaya Amazon Anda](#)

## Mendapatkan Informasi Tentang CloudSearch Domain Amazon

Anda dapat mengambil informasi berikut tentang setiap domain pencarian Anda:

- Nama domain — Nama domain.
- ARN —Nama Sumber Daya Amazon (ARN) domain.
- Titik akhir dokumen —Titik akhir di mana Anda dapat mengirimkan pembaruan dokumen.
- Titik akhir pencarian —Titik akhir yang melaluinya Anda dapat mengirimkan permintaan pencarian.
- Dokumen yang dapat dicari —Jumlah dokumen yang telah diindeks.

- Kebijakan akses —Kebijakan akses yang dikonfigurasi untuk dokumen domain dan titik akhir pencarian.
- Skema analisis —Skema analisis teks yang dapat diterapkan ke bidang indeks domain.
- Bidang indeks —Nama dan jenis setiap bidang indeks yang dikonfigurasi.
- Ekspresi —Ekspresi yang dapat digunakan untuk menyortir hasil pencarian.
- Suggesters —Saran yang dapat digunakan untuk mengambil saran untuk kueri yang tidak lengkap.

Ketika domain pertama kali dibuat, status domain akan menunjukkan bahwa domain saat ini sedang diaktifkan dan tidak ada informasi lain yang tersedia. Setelah dokumen domain dan titik akhir pencarian Anda tersedia, status domain menunjukkan alamat titik akhir yang dapat Anda gunakan untuk menambahkan data dan mengirimkan permintaan pencarian. Jika Anda belum mengirimkan data apa pun untuk pengindeksan, jumlah dokumen yang dapat dicari adalah nol.

Anda dapat melihat semua informasi tentang domain Anda melalui [Amazon CloudSearch console](#). Saat Anda menggunakan `aws cloudsearch describe-domains` perintah atau AWS SDK, ARN domain ditampilkan dalam kebijakan akses domain.

Untuk mendapatkan jumlah dokumen yang dapat dicari, gunakan konsol atau kirimkan `matchall` permintaan ke titik akhir penelusuran domain Anda.

```
q=matchall&q.parser=structured&size=0
```

## Topik

- [Mendapatkan Informasi Domain Menggunakan CloudSearch Konsol Amazon](#)
- [Mendapatkan Informasi CloudSearch Domain Amazon Menggunakan AWS CLI](#)
- [Mendapatkan Informasi Domain Menggunakan AWS SDK](#)

## Mendapatkan Informasi Domain Menggunakan CloudSearch Konsol Amazon

Anda dapat menggunakan CloudSearch konsol Amazon untuk melihat informasi tentang semua domain Anda. Dasbor konsol menampilkan ringkasan setiap domain yang telah Anda buat, termasuk nama domain, status, dan jumlah dokumen yang dapat dicari. Untuk memperbarui tabel dengan informasi terbaru, klik tombol Refresh di bagian atas halaman.

Sebuah domain dapat berada di salah satu dari lima negara bagian:

- **Loading** —Domain baru saja dibuat dan masih diinisialisasi. Anda harus menunggu hingga status domain berubah menjadi PROCESSING, NEEDS INDEXING, atau ACTIVE sebelum Anda dapat mulai mengunggah dokumen.
- **Aktif** —Domain sedang berjalan dan semua bidang yang dikonfigurasi telah diindeks.
- **Perlu Pengindeksan** —Anda telah membuat perubahan pada konfigurasi domain yang memerlukan pembangunan kembali indeks. Jika Anda mencari domain, perubahan ini tidak akan tercermin dalam hasil. Setelah selesai membuat perubahan, pilih Tindakan, Jalankan pengindeksan untuk membangun kembali indeks Anda.
- **Pemrosesan** —Perubahan konfigurasi sedang diterapkan ke domain Anda. Jika Anda mencari domain, perubahan konfigurasi terbaru mungkin tidak tercermin dalam hasil.
- **Dihapus** — Anda memilih untuk menghapus domain dan isinya, dan domain dan semua sumber dayanya sedang dalam proses dihapus. Ketika penghapusan selesai, domain akan dihapus dari daftar domain.

Dari CloudSearch dasbor Amazon, Anda dapat melakukan hal berikut:

- Melihat status domain penelusuran Anda
- Akses dasbor untuk domain tertentu
- Akses CloudSearch dokumentasi Amazon dan sumber daya lainnya

Untuk melihat informasi rinci tentang domain tertentu

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Pilih Domain dari panel navigasi kiri.

Dasbor domain menunjukkan ringkasan status untuk domain yang dipilih. Dari dasbor domain, Anda dapat melakukan hal berikut:

- Lihat status domain
- Unggah dokumen ke domain
- Cari domain
- Akses halaman konfigurasi domain
- Hapus domain



## Mendapatkan Informasi CloudSearch Domain Amazon Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch describe-domains` perintah untuk mendapatkan status domain pencarian Anda. Untuk mendapatkan informasi spesifik seperti kebijakan akses, opsi ketersediaan, dan opsi penskalaan yang dikonfigurasi untuk domain, Anda menggunakan `describe` perintah terpisah untuk setiap opsi. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk mendapatkan informasi status domain

- Jalankan `aws cloudsearch describe-domains` perintah untuk mendapatkan informasi tentang semua domain Anda. Untuk mendapatkan informasi tentang domain tertentu, gunakan `--domain-names` opsi untuk menentukan domain yang Anda minati. Misalnya, permintaan berikut mendapatkan status `movies` domain:

```
aws cloudsearch describe-domains --domain-names movies

{
  "DomainStatusList": [
    {
      "SearchInstanceType": "search.small",
      "DomainId": "965407640801/movies",
      "Created": true,
      "Deleted": false,
      "SearchInstanceCount": 1,
      "DomainName": "movies",
      "SearchService": {
        "Endpoint": "search-movies-m4fcjhuxgj6i76smhyiz7pfxsu.us-
east-1.cloudsearch.amazonaws.com"
      },
      "RequiresIndexDocuments": false,
      "Processing": true,
      "DocService": {
        "Endpoint": "doc-movies-m4fcjhuxgj6i76smhyiz7pfxsu.us-
east-1.cloudsearch.amazonaws.com"
      },
      "ARN": "arn:aws:cloudsearch:us-east-1:965407640801:domain/movies",
      "SearchPartitionCount": 1
    }
  ]
}
```

`describe-domains` Perintah tidak mengembalikan jumlah dokumen yang dapat dicari di domain. Untuk mendapatkan jumlah dokumen yang dapat dicari, gunakan konsol atau kirimkan `matchall` permintaan ke titik akhir penelusuran domain Anda:

```
q=matchall&q.parser=structured&size=0
```

Untuk mendapatkan skema analisis yang dikonfigurasi untuk domain

- Jalankan perintah `aws cloudsearch describe-analysis-schemes`. Misalnya, permintaan berikut mendapatkan skema analisis yang dikonfigurasi untuk `movies` domain:

```
aws cloudsearch describe-analysis-schemes --domain-name movies

{
  "AnalysisSchemes": [
    {
      "Status": {
        "PendingDeletion": false,
        "State": "Active",
        "CreationDate": "2014-03-28T19:27:30Z",
        "UpdateVersion": 31,
        "UpdateDate": "2014-03-28T19:27:30Z"
      },
      "Options": {
        "AnalysisSchemeLanguage": "en",
        "AnalysisSchemeName": "samplescheme",
        "AnalysisOptions": {
          "AlgorithmicStemming": "none",
          "Synonyms": "{\n\"aliases\":{\n\"youth\":[\n\"young adult\"],\n\"groups\":[[\n\"tool box\",\n\"toolbox\"],[\n\"band saw\",\n\"bandsaw\"],[\n\"drill press\",\n\"drillpress\"]]]}",
          "StemmingDictionary": "{}",
          "Stopwords": "[]"
        }
      }
    }
  ]
}
```

Untuk mendapatkan opsi ketersediaan yang dikonfigurasi untuk domain

- Jalankan perintah `aws cloudsearch describe-availability-options`. Misalnya, permintaan berikut mendapatkan opsi ketersediaan yang dikonfigurasi untuk `movies` domain. Jika Multi-AZ diaktifkan untuk domain, `Options` nilainya diatur ke `true`:

```
aws cloudsearch describe-availability-options --domain-name movies

{
  "AvailabilityOptions": {
    "Status": {
      "PendingDeletion": false,
      "State": "Processing",
      "CreationDate": "2014-04-30T20:42:57Z",
      "UpdateVersion": 13,
      "UpdateDate": "2014-05-01T00:17:45Z"
    },
    "Options": true
  }
}
```

Untuk mendapatkan ekspresi yang dikonfigurasi untuk domain

- Jalankan perintah `aws cloudsearch describe-expressions`. Misalnya, permintaan berikut mendapatkan ekspresi yang dikonfigurasi untuk `movies` domain:

```
aws cloudsearch describe-expressions --domain-name movies

{
  "Expression": {
    "Status": {
      "PendingDeletion": false,
      "State": "Processing",
      "CreationDate": "2014-05-01T01:15:18Z",
      "UpdateVersion": 52,
      "UpdateDate": "2014-05-01T01:15:18Z"
    },
    "Options": {
      "ExpressionName": "popularhits",
      "ExpressionValue": "((0.3*popularity)/10.0)+(0.7* _score)"
    }
  }
}
```

```
}  
}
```

## Mendapatkan Informasi Domain Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [DescribeDomains](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

DescribeDomainsTindakan tidak mengembalikan jumlah dokumen yang dapat dicari di domain. Untuk mendapatkan jumlah dokumen yang dapat dicari, gunakan konsol atau kirimkan `matchall` permintaan ke titik akhir penelusuran domain Anda:

```
q=matchall&q.parser=structured&size=0
```

## Memantau CloudSearch Domain Amazon dengan Amazon CloudWatch

Amazon CloudSearch secara otomatis mengirimkan metrik ke Amazon CloudWatch sehingga Anda dapat mengumpulkan dan menganalisis statistik kinerja. Anda dapat memantau metrik ini dengan menggunakan CloudSearch konsol Amazon, atau dengan menggunakan konsol, AWS CLI, atau AWS SDK. CloudWatch Setiap instance penelusuran domain Anda mengirimkan metrik ke CloudWatch interval satu menit. Metrik diarsipkan selama dua minggu; setelah periode itu, data dibuang.

Tidak ada biaya untuk CloudSearch metrik Amazon yang dilaporkan melalui CloudWatch. [Jika Anda menyetel alarm pada metrik, Anda akan ditagih dengan tarif standar. CloudWatch](#) Anda dapat menggunakan metrik di semua wilayah yang didukung oleh Amazon CloudSearch.

### Topik

- [CloudSearch Metrik Amazon](#)
- [Dimensi untuk CloudSearch Metrik Amazon](#)
- [Menghasilkan SDK for Java Metrics untuk Amazon CloudSearch](#)
- [Melihat CloudWatch Metrik untuk Domain Amazon CloudSearch](#)

Tidak semua statistik, seperti Average atau Sum, berlaku untuk setiap metrik. Namun, semua nilai ini tersedia melalui CloudSearch konsol Amazon, atau dengan menggunakan konsol, AWS CLI,

atau AWS SDK untuk semua metrik. CloudWatch Dalam tabel berikut, setiap metrik memiliki daftar Statistik Valid yang berlaku untuk metrik tersebut.

## CloudSearch Metrik Amazon

Namespace AWS/CloudSearch mencakup metrik berikut.

Metrik	Deskripsi
SuccessfulRequests	<p>Jumlah permintaan pencarian yang berhasil diproses oleh instance pencarian.</p> <p>Unit: Hitungan</p> <p>Statistik yang valid: Maksimum, Jumlah</p>
SearchableDocuments	<p>Jumlah dokumen yang dapat dicari dalam indeks pencarian domain.</p> <p>Unit: Hitungan</p> <p>Statistik yang valid: Maksimum</p>
IndexUtilization	<p>Persentase kapasitas indeks instance pencarian yang telah digunakan. Nilai maksimum menunjukkan persentase kapasitas indeks domain yang telah digunakan.</p> <p>Unit: Persen</p> <p>Statistik yang valid: Rata-rata, Maksimum</p>
Partitions	<p>Jumlah partisi indeks didistribusikan.</p> <p>Unit: Hitungan</p> <p>Statistik yang valid: Minimum, Maksimum</p>

## Dimensi untuk CloudSearch Metrik Amazon

Amazon CloudSearch mengirimkan ClientId dan DomainName dimensi ke CloudWatch.

Dimensi	Deskripsi
ClientId	ID akun AWS.
DomainName	Nama domain pencarian.

## Menghasilkan SDK for Java Metrics untuk Amazon CloudSearch

Ini AWS SDK for Java dapat menghasilkan metrik kinerja untuk CloudSearch klien Amazon Anda dan mengirimkannya CloudWatch untuk visualisasi. Untuk argumen Java VM yang mengaktifkan fitur ini, lihat [Mengaktifkan Metrik untuk AWS SDK for Java di Panduan Pengembang](#). AWS SDK for Java

Anda dapat menggunakan kode berikut untuk menguji pembuatan metrik. Kode membuat CloudWatch klien baru dan melakukan 2.500 pencarian. Karena SDK hanya mengirimkan metrik sekali per menit, klien yang berjalan lama bekerja paling baik. Kode menggunakan [rantai penyedia kredensi default](#).

```
import com.amazonaws.client.builder.AwsClientBuilder;
import com.amazonaws.services.cloudsearchdomain.AmazonCloudSearchDomain;
import com.amazonaws.services.cloudsearchdomain.AmazonCloudSearchDomainClientBuilder;
import com.amazonaws.services.cloudsearchdomain.model.SearchRequest;

public class Metrics {

    public static void main(String[] args) {

        String search_endpoint = "https://search-domain-id.us-west-1.cloudsearch.amazonaws.com";
        String region = "us-west-1";

        AwsClientBuilder.EndpointConfiguration endpointConfig = new AwsClientBuilder
            .EndpointConfiguration(search_endpoint, region);

        AmazonCloudSearchDomainClientBuilder builder = AmazonCloudSearchDomainClientBuilder
            .standard()
            .withEndpointConfiguration(endpointConfig);

        AmazonCloudSearchDomain client = builder.build();

        String query;
```

```
SearchRequest request = new SearchRequest();
com.amazonaws.services.cloudsearchdomain.model.SearchResult test =
client.search(request);

for (int i = 0; i < 2500; i++) {
    query = "test";
    request.setQuery(query);
    test = client.search(request);
    System.out.println(test.toString());
}
}
```

Untuk memverifikasi bahwa SDK mengirim metrik ke CloudWatch, periksa halaman Metrik CloudWatch konsol dan cari AWSSDK/Java di bawah bagian Custom Namespaces. Metrik mungkin membutuhkan waktu beberapa menit untuk ditampilkan.

## Melihat CloudWatch Metrik untuk Domain Amazon CloudSearch

CloudSearch Konsol Amazon membuat grafik metrik yang dilaporkan. CloudWatch Anda juga dapat mengakses metrik melalui [CloudWatchkonsol](#), AWS CLI, dan AWS SDK. Untuk informasi selengkapnya, lihat [Melihat, Membuat Grafik, dan Menerbitkan Metrik](#) di Panduan CloudWatch Pengembang Amazon.

Untuk melihat metrik untuk domain penelusuran menggunakan konsol Amazon CloudSearch

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch>.
2. Pilih Domain dari panel navigasi kiri.
3. Klik nama domain, lalu buka tab Monitoring.

## Mencatat Panggilan API CloudSearch Konfigurasi Amazon dengan AWS CloudTrail

Amazon CloudSearch terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Amazon CloudSearch. CloudTrail menangkap semua panggilan API konfigurasi untuk Amazon CloudSearch sebagai peristiwa.

**Note**

CloudTrail hanya menangkap panggilan ke [API konfigurasi](#), seperti `CreateDomain` dan `UpdateServiceAccessPolicies`, bukan [API layanan dokumen atau API pencarian](#).

Panggilan yang diambil termasuk panggilan dari CloudSearch konsol Amazon, CLI, atau SDK. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk Amazon CloudSearch. 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 ke Amazon CloudSearch, alamat IP dari mana permintaan itu dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

## CloudSearch Informasi Amazon di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Ketika aktivitas terjadi di Amazon CloudSearch, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di akun AWS. Untuk informasi selengkapnya, lihat [Melihat Acara dengan Riwayat CloudTrail Acara](#).

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk Amazon CloudSearch, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol tersebut, jejak diterapkan ke semua Wilayah AWS. Jejak mencatat kejadian dari semua Wilayah di partisi AWS dan mengirimkan berkas log ke bucket Amazon 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](#)



Semua tindakan API CloudSearch konfigurasi Amazon dicatat oleh CloudTrail dan didokumentasikan dalam file [the section called "Referensi API Konfigurasi"](#).

Setiap peristiwa atau entri log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut ini:

- Bahwa permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna (IAM) AWS Identity and Access Management.
- Baik permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan dibuat oleh layanan AWS lain.

Untuk informasi lain, lihat [Elemen userIdentity CloudTrail](#).

## Memahami Entri File CloudSearch Log Amazon

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, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan CreateDomain tindakan.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/test-user",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "test-user",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T23:31:33Z"
      }
    }
  },
}
```

```
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2018-08-21T23:32:15Z",
  "eventSource": "cloudsearch.amazonaws.com",
  "eventName": "CreateDomain",
  "awsRegion": "us-west-1",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "domainName": "test-domain"
  },
  "responseElements": {
    "domainStatus": {
      "aRN": "arn:aws:cloudsearch:us-west-1:123456789012:domain/test-domain",
      "searchInstanceCount": 0,
      "docService": {},
      "requiresIndexDocuments": false,
      "deleted": false,
      "searchService": {},
      "domainId": "123456789012/test-domain",
      "processing": false,
      "created": true,
      "searchPartitionCount": 0,
      "domainName": "test-domain"
    }
  },
  "requestID": "12345678-1234-1234-1234-987654321098",
  "eventID": "87654321-4321-4321-4321-987654321098",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

## Melacak CloudSearch Penggunaan dan Biaya Amazon Anda

Halaman aktivitas akun AWS memungkinkan Anda melacak CloudSearch penggunaan dan tagihan Amazon Anda.

Untuk mendapatkan informasi CloudSearch penggunaan Amazon Anda

1. Buka [aws.amazon.com](https://aws.amazon.com) dan pilih Akun Saya, Penagihan & Manajemen Biaya.
2. Pilih Laporan Biaya & Penggunaan, lalu pilih Laporan AWS Penggunaan.
3. Pilih Amazon CloudSearch dari dropdown layanan.

4. Tentukan informasi yang ingin Anda sertakan dalam laporan, lalu pilih tombol unduh untuk format data yang ingin Anda unduh. Laporan dapat diunduh dalam format XML/CSV.

## Menghapus Domain Amazon CloudSearch

Jika Anda tidak lagi menggunakan domain pencarian, Anda harus menghapusnya untuk menghindari biaya penggunaan tambahan. Anda masih akan dikenakan biaya untuk domain meskipun tidak berisi dokumen apa pun—menghapus semua dokumen tidak menghapus domain. Menghapus domain menghapus indeks yang terkait dengan domain dan mengambil dokumen domain dan titik akhir pencarian offline secara permanen. Diperlukan beberapa waktu untuk sepenuhnya menghapus domain dan menonaktifkan semua sumber dayanya. Domain kecil biasanya dihapus dalam waktu singkat, sementara terutama domain besar mungkin memerlukan waktu yang lama untuk dihapus. Selama proses ini, status domain adalah `Being Deleted` dan akun Anda tidak dikenakan biaya.

Anda dapat menghapus domain dari CloudSearch konsol Amazon, menggunakan `aws cloudsearch delete-domain` perintah, atau menggunakan AWS SDK.

### Topik

- [Menghapus Domain Menggunakan Konsol Amazon CloudSearch](#)
- [Menghapus Domain Menggunakan AWS CLI](#)
- [Menghapus CloudSearch Domain Amazon Menggunakan AWS SDK](#)

## Menghapus Domain Menggunakan Konsol Amazon CloudSearch

Anda dapat dengan mudah menghapus domain dari dasbor domain di CloudSearch konsol Amazon.

### Cara menghapus domain

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel Navigasi kiri, pilih Domain.
3. Pilih kotak centang di sebelah domain yang ingin Anda hapus, lalu pilih Hapus dan konfirmasi penghapusan.

## Menghapus Domain Menggunakan AWS CLI

Jalankan `aws cloudsearch delete-domain` perintah dan tentukan nama domain yang ingin Anda hapus. Misalnya, untuk menghapus domain film, Anda tentukan `--domain-name movies`.

```
aws cloudsearch delete-domain --domain-name movies
```

Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

## Menghapus CloudSearch Domain Amazon Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [DeleteDomain](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Menandai Domain Amazon CloudSearch

Gunakan CloudSearch tag Amazon untuk melampirkan metadata ke domain penelusuran Anda. AWS tidak menerapkan makna semantik pada tag Anda; tag ditafsirkan secara ketat sebagai string karakter. Semua tag berisi elemen-elemen berikut.

Elemen tanda	Deskripsi
Tombol tanda	Kunci tag adalah nama wajib tag. Kunci tag harus unik untuk domain tempat mereka dilampirkan. Untuk daftar batasan dasar pada kunci dan nilai tag, lihat <a href="#">Pembatasan Tag</a> .
Nilai tanda	Nilai tag adalah nilai string opsional dari tag. Nilai tag bisa null dan tidak harus unik dalam satu set tag. Misalnya, Anda dapat menggunakan pasangan kunci-nilai dalam satu rangkaian tag <code>project/Trinity</code> dan <code>cost-center/Trinity</code> . Untuk daftar batasan dasar pada kunci dan nilai tag, lihat <a href="#">Pembatasan Tag</a> .

Setiap CloudSearch domain Amazon memiliki kumpulan tag, yang berisi semua tag yang ditetapkan ke domain tersebut. AWS tidak secara otomatis menyetel tag apa pun di CloudSearch domain

Amazon. Rangkaian tag dapat berisi sepuluh tag atau kosong. Jika Anda menambahkan tag ke CloudSearch domain Amazon yang memiliki kunci yang sama dengan tag yang ada untuk sumber daya, nilai baru akan menimpa nilai lama.

Anda dapat menggunakan kunci tag untuk menentukan kategori, dan nilai tag dapat berupa item dalam kategori tersebut. Misalnya, Anda dapat menentukan kunci tag `project` dan nilai tag yang `Salix` menunjukkan bahwa domain ditetapkan ke proyek Salix. Anda juga dapat menggunakan tag untuk menunjuk domain untuk lingkungan pengujian atau produksi dengan menggunakan kunci seperti `environment=test` dan `environment=production`. Sebaiknya gunakan sekumpulan kunci tag yang konsisten agar lebih mudah melacak metadata yang terkait dengan domain penelusuran Anda.

Anda juga dapat menggunakan tag untuk mengatur tagihan AWS Anda untuk mencerminkan struktur biaya Anda sendiri dan untuk melacak biaya dengan mengelompokkan pengeluaran untuk sumber daya yang ditandai serupa. Untuk melakukannya, daftar untuk mendapatkan tagihan akun AWS Anda dengan menyertakan nilai kunci tag. Lalu, kelola informasi penagihan Anda sesuai dengan sumber daya dengan nilai kunci tanda yang sama untuk melihat biaya sumber daya gabungan. Misalnya, Anda dapat menandai beberapa CloudSearch domain Amazon dengan pasangan nilai kunci, lalu mengatur informasi penagihan untuk melihat total biaya untuk setiap domain di beberapa layanan. Untuk informasi selengkapnya, lihat [Alokasi dan Penandaan Biaya dalam dokumentasi](#) AWS Billing and Cost Management.

#### Note

Tag disimpan di cache untuk diotorisasi. Karena itu, penambahan dan pembaruan tag di CloudSearch domain Amazon mungkin memakan waktu beberapa menit sebelum tersedia.

## Bekerja dengan Tag (Konsol)

Gunakan prosedur berikut untuk membuat tag sumber daya dengan CloudSearch konsol Amazon.

Untuk membuat tag

1. Buka CloudSearch konsol Amazon dan pilih nama domain Anda untuk membuka panel konfigurasinya.
2. Buka tab Tag dan pilih Kelola.
3. Masukkan kunci tag dan nilai opsional, lalu pilih Kirim.

Untuk informasi selengkapnya tentang menggunakan konsol untuk bekerja dengan tag, lihat [Bekerja dengan Editor Tag](#) di Panduan Memulai Konsol Manajemen AWS.

# Mengontrol Bagaimana Data Diindeks di Amazon CloudSearch

Anda mengontrol bagaimana data Anda diindeks dengan mengonfigurasi opsi pengindeksan dan skema analisis untuk domain Anda. Opsi pengindeksan mengontrol bagaimana data Anda dipetakan ke bidang indeks dan informasi apa yang dapat Anda cari dan ambil dari indeks. Data yang Anda unggah harus berisi bidang yang sama yang dikonfigurasi dalam opsi pengindeksan domain Anda, dan nilai bidang harus kompatibel dengan jenis bidang yang dikonfigurasi. Skema analisis mengontrol bagaimana `text` dan `text-array` bidang diproses selama pengindeksan dengan mendefinisikan opsi stemming, stopwords, dan sinonim khusus bahasa.

## Topik

- [Mempersiapkan Data Anda untuk Amazon CloudSearch](#)
- [Mengonfigurasi Bidang Indeks untuk Domain Amazon CloudSearch](#)
- [Menggunakan Bidang Dinamis di Amazon CloudSearch](#)
- [Mengkonfigurasi Skema Analisis Teks untuk Amazon CloudSearch](#)
- [Pemrosesan Teks di Amazon CloudSearch](#)

## Mempersiapkan Data Anda untuk Amazon CloudSearch

Anda perlu memformat data Anda dalam JSON atau XML sebelum Anda dapat mengunggahnya ke domain pencarian Anda untuk pengindeksan. Setiap item yang ingin Anda terima sebagai hasil pencarian direpresentasikan sebagai dokumen. Setiap dokumen memiliki ID dokumen unik dan satu atau beberapa bidang yang berisi data yang ingin Anda cari dan kembalikan hasilnya. Bidang dokumen ini digunakan untuk mengisi bidang indeks yang Anda konfigurasi untuk domain Anda. Untuk informasi selengkapnya, lihat [configure indexing options](#).

[Creating Document Batches](#) menjelaskan cara memformat data Anda. Untuk penjelasan rinci tentang skema Amazon CloudSearch JSON dan XHTML, lihat [Document Service API](#)

## Topik

- [Memetakan Data Dokumen ke Bidang Indeks di Amazon CloudSearch](#)
- [Membuat Batch Dokumen di Amazon CloudSearch](#)

## Memetakan Data Dokumen ke Bidang Indeks di Amazon CloudSearch

Untuk mengisi bidang dalam indeks Anda, Amazon CloudSearch membaca data dari bidang dokumen yang sesuai. Setiap bidang yang ditentukan dalam data dokumen Anda harus dikonfigurasi dalam opsi pengindeksan Anda. Dokumen dapat berisi subset bidang yang dikonfigurasi untuk domain—setiap dokumen tidak harus berisi semua bidang. Selain itu, Anda dapat mengisi bidang tambahan dalam indeks Anda dengan menyalin data dari satu bidang ke bidang lainnya. Ini memungkinkan Anda untuk menggunakan data sumber yang sama dengan cara yang berbeda dengan mengonfigurasi opsi yang berbeda untuk bidang.

Bidang array seperti `text-array` dapat berisi hingga 1000 nilai. Pada waktu pencarian, dokumen dikembalikan sebagai hit jika salah satu dari nilai-nilai tersebut cocok dengan permintaan pencarian.

## Membuat Batch Dokumen di Amazon CloudSearch

### Important

Sebelum mengunggah data ke CloudSearch domain Amazon, ikuti panduan berikut:

- Kelompokkan dokumen ke dalam batch sebelum Anda mengunggahnya. Terus mengunggah batch yang hanya terdiri dari satu dokumen memiliki dampak negatif yang besar pada kecepatan Amazon CloudSearch dapat memproses pembaruan Anda. Sebagai gantinya, buat batch yang sedekat mungkin dengan batas dan unggah lebih jarang. Untuk informasi selengkapnya tentang ukuran batch maksimum dan frekuensi unggah, lihat [Batas](#).
- Dokumen domain dan titik akhir pencarian tetap sama untuk masa pakai domain. Anda harus men-cache titik akhir daripada mengambilnya sebelum setiap permintaan unggahan atau pencarian. Menanyakan layanan CloudSearch konfigurasi Amazon dengan menelepon `aws cloudsearch describe-domains` atau `DescribeDomains` sebelum setiap permintaan kemungkinan akan mengakibatkan permintaan Anda dibatasi.

Anda membuat kumpulan dokumen untuk mendeskripsikan data yang ingin Anda cari. Saat Anda mengirim batch dokumen ke domain, data diindeks secara otomatis sesuai dengan opsi pengindeksan domain. CloudSearch Konsol Amazon dapat secara otomatis menghasilkan kumpulan dokumen dari berbagai dokumen sumber.



Kumpulan dokumen adalah kumpulan operasi tambah dan hapus yang mewakili dokumen yang ingin Anda tambahkan, perbarui, atau hapus dari domain Anda. Batch dapat dijelaskan dalam JSON atau XHTML. Lihat [Batas](#) untuk ukuran batch maksimum dan ukuran dokumen.

Untuk mendapatkan performa unggahan sebaik mungkin, kelompokkan operasi tambah dan hapus dalam batch yang mendekati ukuran batch maksimum. Mengirimkan sejumlah besar kumpulan dokumen tunggal ke layanan dokumen dapat meningkatkan waktu yang diperlukan agar perubahan Anda terlihat di hasil penelusuran. Jika Anda memiliki sejumlah besar data untuk diunggah, Anda dapat mengirim batch secara paralel. Jumlah pengunggah simultan yang dapat Anda gunakan bergantung pada jenis instance pencarian. Anda dapat melakukan prescale untuk upload massal dengan menyetel opsi jenis instans yang diinginkan untuk domain Anda. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

Untuk setiap dokumen dalam batch, Anda harus menentukan:

- Operasi yang ingin Anda lakukan: menambah atau menghapus.
- ID unik untuk dokumen. ID dokumen dapat berisi huruf atau angka dan karakter berikut: `_ - # ; / ? @ &`. ID dokumen harus minimal 1 dan panjangnya tidak lebih dari 128 karakter.
- Pasangan nama-nilai untuk setiap bidang dokumen. Untuk menentukan nilai `latlon` bidang, Anda menentukan garis lintang dan bujur sebagai daftar yang dipisahkan koma; misalnya, `"location_field": "35.628611, -120.694152"` Saat menentukan dokumen di JSON, nilai untuk bidang tidak bisa `null` (Namun, Anda dapat menghilangkan bidang sepenuhnya.)

Misalnya, batch JSON berikut menambahkan satu dokumen dan menghapus satu dokumen:

```
[
  {"type": "add",
   "id": "tt0484562",
   "fields": {
     "title": "The Seeker: The Dark Is Rising",
     "directors": ["Cunningham, David L."],
     "genres": ["Adventure", "Drama", "Fantasy", "Thriller"],
     "actors": ["McShane, Ian", "Eccleston, Christopher", "Conroy, Frances",
               "Crewson, Wendy", "Ludwig, Alexander", "Cosmo, James",
               "Warner, Amelia", "Hickey, John Benjamin", "Piddock, Jim",
               "Lockhart, Emma"]
   }
 },
  {"type": "delete",
   "id": "tt0484575"
 }
```

```
}
]
```

Batch yang sama yang diformat dalam XML terlihat seperti ini:

```
<batch>
  <add id="tt0484562">
    <field name="title">The Seeker: The Dark Is Rising</field>
    <field name="directors">Cunningham, David L.</field>
    <field name="genres">Adventure</field>
    <field name="genres">Drama</field>
    <field name="genres">Fantasy</field>
    <field name="genres">Thriller</field>
    <field name="actors">McShane, Ian</field>
    <field name="actors">Eccleston, Christopher</field>
    <field name="actors">Conroy, Frances</field>
    <field name="actors">Ludwig, Alexander</field>
    <field name="actors">Crewson, Wendy</field>
    <field name="actors">Warner, Amelia</field>
    <field name="actors">Cosmo, James</field>
    <field name="actors">Hickey, John Benjamin</field>
    <field name="actors">Pidcock, Jim</field>
    <field name="actors">Lockhart, Emma</field>
  </add>
  <delete id="tt0484575" />
</batch>
```

Amazon CloudSearch menerima batch hanya jika semua dokumen di dalamnya valid. Anda dapat memverifikasi validitas data JSON atau XHTML Anda menggunakan alat seperti `xmlint` dan `jsonlint`.

Batch JSON dan XHTML hanya dapat berisi karakter UTF-8 yang valid dalam XML.2. Karakter yang valid adalah tab karakter kontrol (0009), carriage return (000D), dan line feed (000A), dan karakter hukum Unicode dan ISO/IEC 10646. FFFE, FFFF, dan blok pengganti D800—DBFF dan DC00—DFFF tidak valid dan akan menyebabkan kesalahan. (Untuk informasi selengkapnya, lihat [Extensible Markup Language \(XHTML\) 1.0 \(Edisi Kelima\)](#).) Anda dapat menggunakan ekspresi reguler berikut untuk mencocokkan karakter yang tidak valid sehingga Anda dapat menghapusnya: `[^\u0009\u000a\u000d\u0020-\u0D7FF\uE000-\uFFFF]/`

Saat memformat data Anda di JSON, tanda kutip (“) dan garis miring terbalik (\) dalam nilai bidang harus diloloskan dengan garis miring terbalik. Misalnya:

```
"title":"Where the Wild Things Are"  
"isbn":"0-06-025492-0"  
"image":"images\\covers\\Where_The_Wild_Things_Are_(book)_cover.jpg"  
"comment":"Sendak's \"Where the Wild Things Are\" is a children's classic."
```

Saat memformat data Anda dalam XHTML, ampersands (&) dan simbol kurang dari (<) dalam nilai bidang perlu direpresentasikan dengan referensi entitas yang sesuai (dan). & amp; & lt; ;

Misalnya:

```
<field name="title">Little Cow & amp; ; the Turtle</field>  
<field name="isbn">0-84466-4774</field>  
<field name="image">images\\covers\\Little_Cow_& amp; ;_the_Turtle.jpg</field>  
<field name="comment">& lt; ;insert comment<></field>
```

Jika Anda memiliki blok besar konten buatan pengguna, Anda mungkin ingin membungkus seluruh bidang di bagian CDATA, daripada mengganti setiap kejadian dengan referensi entitas. Misalnya:

```
<field name="comment"><![CDATA[Monsters & mayhem--what's not to like! ]]>
```

## Menambahkan dan Memperbarui Dokumen di Amazon CloudSearch

Operasi add menentukan dokumen baru yang ingin Anda tambahkan ke indeks atau dokumen yang sudah ada yang ingin Anda perbarui.

Saat Anda menambahkan atau memperbarui dokumen, Anda menentukan ID dokumen dan semua bidang yang terkandung dalam dokumen. Anda tidak perlu menentukan setiap bidang yang dikonfigurasi untuk setiap dokumen—dokumen dapat berisi subset dari bidang yang dikonfigurasi. Namun, setiap bidang dalam dokumen harus sesuai dengan bidang yang dikonfigurasi untuk domain.

Untuk menambahkan dokumen ke domain pencarian

1. Tentukan operasi tambah yang berisi ID dokumen yang ingin Anda tambahkan dan setiap bidang yang ingin Anda cari atau kembalikan dalam hasil. Jika dokumen sudah ada, operasi add akan menggantikannya. (Anda tidak dapat memperbarui bidang yang dipilih, dokumen ditimpa dengan versi baru.) Misalnya, operasi berikut menambahkan dokumen tt0484562:

```
{ "type": "add",
```

```
"id": "tt0484562",
"fields": {
  "title": "The Seeker: The Dark Is Rising",
  "directors": ["Cunningham, David L."],
  "genres": ["Adventure", "Drama", "Fantasy", "Thriller"],
  "actors": ["McShane, Ian", "Eccleston, Christopher", "Conroy, Frances",
    "Crewson, Wendy", "Ludwig, Alexander", "Cosmo, James",
    "Warner, Amelia", "Hickey, John Benjamin", "Piddock, Jim",
    "Lockhart, Emma"]
}
```

2. Sertakan operasi tambah dalam kumpulan dokumen dan unggah batch ke domain Anda. Anda dapat mengunggah data melalui CloudSearch konsol Amazon atau dengan memposting permintaan langsung ke titik akhir layanan dokumen domain. Untuk informasi selengkapnya, lihat [upload documents](#).

## Menghapus Dokumen di Amazon CloudSearch

Operasi hapus menentukan dokumen yang ingin Anda hapus dari indeks domain. Setelah dokumen dihapus, itu tidak akan lagi dapat dicari atau dikembalikan dalam hasil.

Saat memposting pembaruan untuk menghapus dokumen, Anda harus menentukan setiap dokumen yang ingin Anda hapus.

Jika domain Anda telah ditingkatkan untuk mengakomodasi ukuran indeks Anda dan Anda menghapus sejumlah besar dokumen, skala domain akan turun saat berikutnya indeks penuh dibangun kembali. Meskipun indeks secara otomatis dibangun kembali secara berkala, untuk menurunkan skala secepat mungkin Anda dapat secara eksplisit [menjalankan pengindeksan](#) ketika Anda selesai menghapus dokumen.

### Note

Untuk menghapus dokumen, Anda mengunggah kumpulan dokumen yang berisi operasi penghapusan. Anda ditagih untuk jumlah total kumpulan dokumen yang diunggah ke domain pencarian Anda, termasuk batch yang berisi operasi penghapusan. Untuk informasi selengkapnya tentang CloudSearch harga Amazon, lihat [aws.amazon.com/cloudsearch/pricing/](https://aws.amazon.com/cloudsearch/pricing/).

Untuk menghapus dokumen dari domain pencarian

1. Tentukan operasi hapus yang berisi ID dokumen yang ingin Anda hapus. Misalnya, operasi berikut akan menghapus dokumen `tt0484575`:

```
{
  "type": "delete",
  "id": "tt0484575"
}
```

2. Sertakan operasi hapus dalam kumpulan dokumen dan unggah batch ke domain Anda. Anda dapat mengunggah batch melalui CloudSearch konsol Amazon atau dengan memposting permintaan langsung ke titik akhir layanan dokumen domain. Untuk informasi selengkapnya, lihat [upload documents](#).
3. [Operasi hapus menghapus dokumen dari indeks Anda—dokumen tersebut tidak akan muncul di hasil penelusuran—tetapi untuk menghapusnya seluruhnya dari Amazon CloudSearch, Anda juga harus membangun kembali indeks Anda.](#)

## Memproses Data Sumber Anda untuk Amazon CloudSearch

Untuk mengunggah data untuk pengindeksan, Anda perlu memformat data Anda dalam JSON atau XHTML. CloudSearch Konsol Amazon menyediakan cara untuk secara otomatis menghasilkan JSON atau XHTML yang diformat dengan benar dari beberapa jenis file umum: CSV, teks, dan HTML. Anda juga dapat memproses batch yang diformat untuk Amazon CloudSearch 2011-02-01 API untuk mengonversinya ke format 2013-01-01.

Untuk sebagian besar jenis file, setiap file sumber direpresentasikan sebagai dokumen terpisah dalam JSON atau XHTML yang dihasilkan. Jika metadata tersedia untuk file, metadata dipetakan ke bidang dokumen yang sesuai—bidang yang dihasilkan dari metadata dokumen bervariasi tergantung pada jenis file. Isi file sumber diuraikan menjadi satu bidang teks. Jika file berisi lebih dari 1 MB data, data yang dipetakan ke bidang teks terpotong sehingga dokumen tidak melebihi 1 MB.

File CSV ditangani secara berbeda. Saat memproses file CSV, Amazon CloudSearch menggunakan konten baris pertama untuk menentukan bidang dokumen, dan membuat dokumen terpisah untuk setiap baris berikut. Jika ada header kolom yang disebut `docid`, nilai di kolom tersebut digunakan sebagai ID dokumen. Jika perlu, nilai `docid` dinormalisasi agar sesuai dengan set karakter yang diizinkan. ID dokumen dapat berisi huruf atau angka dan karakter berikut: `_ - = #; : / ? @ &`. Jika tidak ada kolom `docid`, ID unik dihasilkan untuk setiap dokumen berdasarkan nama file dan nomor baris.

Jika Anda mengunggah beberapa jenis file, file CSV diuraikan row-by-row, dan file non-CSV diperlakukan sebagai dokumen individual.

 Note

Saat ini, hanya file CSV yang diurai untuk secara otomatis mengekstrak data bidang khusus dan menghasilkan banyak dokumen.

Anda juga dapat memproses data yang disimpan di DynamoDB. Amazon CloudSearch mewakili setiap item yang dibaca dari tabel sebagai dokumen terpisah.

### Memproses Data Sumber Menggunakan CloudSearch Konsol Amazon

Saat Anda mengunggah dokumen sumber atau item DynamoDB melalui konsol CloudSearch Amazon, dokumen tersebut secara otomatis dikonversi ke format Amazon CloudSearch JSON. Anda dapat menggunakan konsol untuk mengunggah hingga 5 MB data sekaligus. Jika mau, Anda dapat mengunduh file JSON yang dihasilkan. Untuk informasi selengkapnya tentang mengunggah data melalui konsol, lihat [upload documents](#) dan [Uploading DynamoDB Data](#).

## Mengonfigurasi Bidang Indeks untuk Domain Amazon CloudSearch

Setiap dokumen yang Anda tambahkan ke domain pencarian Anda memiliki kumpulan bidang yang berisi data yang dapat dicari atau dikembalikan. Setiap dokumen harus memiliki ID dokumen unik dan setidaknya satu bidang.

Dalam konfigurasi domain Anda, Anda menentukan bidang indeks untuk setiap bidang yang terjadi dalam dokumen Anda. Anda tidak dapat mengunggah dokumen yang berisi bidang yang tidak dikenal. Namun, setiap dokumen tidak harus berisi semua bidang—dokumen dapat berisi subset bidang yang dikonfigurasi untuk domain.

### Topik

- [Mengkonfigurasi Bidang Indeks Individu dengan AWS CLI](#)
- [Mengonfigurasi Bidang Indeks Menggunakan Konsol Amazon CloudSearch](#)
- [Mengonfigurasi Bidang CloudSearch Indeks Amazon Menggunakan AWS SDK](#)

Amazon CloudSearch mendukung jenis bidang indeks berikut:

- `date`—berisi stempel waktu. Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut [IETF RFC3339](#): `yyyy-mm-ddTHH:mm:ss.SSSZ` Di UTC, misalnya, 17:00 23 Agustus 1970 adalah: `1970-08-23T17:00:00Z` Perhatikan bahwa Anda juga dapat menentukan detik pecahan saat menentukan waktu di UTC. Misalnya, `1967-01-31T23:20:50.650Z`.
- `date-array`—bidang tanggal yang dapat berisi beberapa nilai.
- `double`—berisi nilai floating point 64-bit presisi ganda.
- `double-array`—bidang ganda yang dapat berisi beberapa nilai.
- `int`—berisi nilai integer bertanda 64-bit.
- `int-array`—sebuah bidang integer yang dapat berisi beberapa nilai.
- `latlon`—berisi lokasi yang disimpan sebagai pasangan nilai lintang dan bujur (`lat`, `lon`)
- `literal`—berisi pengenalan atau data lain yang ingin Anda cocokkan dengan tepat. Bidang literal peka huruf besar/kecil.
- `literal-array`—bidang literal yang dapat berisi beberapa nilai.
- `text`—berisi data alfanumerik arbitrer.
- `text-array`—bidang teks yang dapat berisi beberapa nilai.

Nama kolom indeks reguler harus dimulai dengan huruf dan setidaknya 3 dan panjangnya tidak lebih dari 64 karakter. Karakter yang diizinkan adalah: a-z (huruf kecil), 0-9, dan `_` (garis bawah). Skor nama dicadangkan dan tidak dapat ditentukan sebagai nama bidang. Semua nama bidang dan ekspresi harus unik.

Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (`*`). String sebelum atau sesudah wildcard dapat berisi kumpulan karakter yang sama dengan bidang indeks biasa. Untuk informasi selengkapnya tentang bidang dinamis, lihat [the section called "Menggunakan Bidang Dinamis"](#).

Opsi yang dapat Anda konfigurasi untuk bidang bervariasi sesuai dengan jenis bidang:

- `HighlightEnabled`—Anda bisa mendapatkan informasi penyorotan untuk klik pencarian di bidang `HighlightEnabled` teks apa pun. Berlaku untuk: `text`, `text-array`.
- `FacetEnabled`—Anda bisa mendapatkan informasi aspek untuk bidang apa pun `FacetEnabled`. Bidang teks tidak dapat digunakan untuk faset. Berlaku untuk: `int`, `int-array`, `date`, `date-array`, `double`, `double-array`, `latlon`, `literal`, `literal-array`.
- `ReturnEnabled`—Anda dapat mengambil nilai `ReturnEnabled` bidang apa pun dengan hasil pencarian Anda. Perhatikan bahwa ini meningkatkan ukuran indeks Anda, yang dapat meningkatkan biaya menjalankan domain Anda. Jika memungkinkan, yang terbaik adalah

mengambil sejumlah besar data dari sumber eksternal, daripada menyimpannya dalam indeks Anda. Karena perlu beberapa waktu untuk menerapkan pembaruan dokumen di seluruh domain, data penting seperti informasi harga harus diambil dari sumber eksternal menggunakan ID dokumen yang dikembalikan. Berlaku untuk: `int,int-array,date,date-array,double,double-array,latlon,,literal,literal-array,text,text-array`.

- **SearchEnabled**—Anda dapat mencari isi SearchEnabled bidang apa pun. Bidang teks selalu dapat dicari. Berlaku untuk: `int,int-array,date,date-array,double,double-array,latlon,,literal,literal-array,text,text-array`.
- **SortEnabled**—Anda dapat mengurutkan hasil pencarian menurut abjad atau numerik menggunakan bidang apa pun. SortEnabled Bidang tipe array tidak bisa. SortEnabled Hanya mengurutkan bidang numerik yang diaktifkan yang dapat digunakan dalam ekspresi. Berlaku untuk: `int,date,latlon,double,literal,text`.

Anda juga dapat menentukan nilai default dan sumber untuk bidang apa pun. Menentukan nilai default dapat menjadi penting jika Anda menggunakan bidang numerik dalam ekspresi dan bidang itu tidak ada di setiap dokumen. Menentukan sumber menyalin data dari satu bidang ke bidang lainnya, memungkinkan Anda untuk menggunakan data sumber yang sama dengan cara yang berbeda dengan mengonfigurasi opsi yang berbeda untuk bidang. Anda dapat menggunakan wildcard (\*) saat menentukan nama sumber untuk menyalin data dari semua bidang yang cocok dengan pola yang ditentukan.

Ketika Anda menambahkan bidang atau memodifikasi bidang yang ada, Anda harus secara eksplisit mengeluarkan permintaan untuk mengindeks ulang data Anda ketika Anda selesai membuat perubahan konfigurasi. Untuk informasi selengkapnya, lihat [rebuild the index](#).

#### Important

Jika Anda mengubah jenis bidang dan memiliki dokumen dalam indeks Anda yang berisi data yang tidak kompatibel dengan jenis bidang baru, semua bidang yang sedang diproses dimasukkan ke dalam `FailedToValidate` status saat Anda menjalankan pengindeksan dan operasi pengindeksan gagal. Mengembalikan perubahan konfigurasi yang tidak kompatibel akan memungkinkan Anda untuk berhasil membangun kembali indeks Anda. Jika perubahan diperlukan, Anda harus memperbarui atau menghapus dokumen yang tidak kompatibel dari indeks Anda untuk menggunakan konfigurasi baru.



## Mengkonfigurasi Bidang Indeks Individu dengan AWS CLI

Anda menggunakan `aws cloudsearch define-index-field` perintah untuk mengonfigurasi kolom indeks individual untuk domain pencarian. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk menambahkan kolom indeks ke domain Anda

- Jalankan `aws cloudsearch define-index-field` perintah dan tentukan nama bidang baru dengan `--name` opsi, dan jenis bidang dengan `--type` opsi. Contoh berikut menambahkan int bidang yang dipanggil `year` ke domain `film`.

### Example

```
aws cloudsearch define-index-field --domain-name movies --name year --type int
{
  "IndexField": {
    "Status": {
      "PendingDeletion": false,
      "State": "RequiresIndexDocuments",
      "CreationDate": "2014-06-25T23:03:06Z",
      "UpdateVersion": 15,
      "UpdateDate": "2014-06-25T23:03:06Z"
    },
    "Options": {
      "IndexFieldType": "int",
      "IndexFieldName": "year"
    }
  }
}
```

### Note

Ketika Anda menambahkan bidang atau memodifikasi bidang yang ada, Anda harus secara eksplisit mengeluarkan permintaan untuk mengindeks ulang data Anda ketika Anda selesai membuat perubahan konfigurasi. Untuk informasi selengkapnya, lihat [rebuild the index](#).

## Mengonfigurasi Bidang Indeks Menggunakan Konsol Amazon CloudSearch

Anda dapat dengan mudah [configure individual index fields](#) untuk domain Anda melalui panel Opsi Pengindeksan di CloudSearch konsol Amazon. Mengonfigurasi bidang indeks di konsol memerlukan `DefineIndexFields` tindakan, yang AWS CLI tidak didukung.

### Mengonfigurasi Bidang Individu Menggunakan Konsol Amazon CloudSearch

Untuk mengkonfigurasi bidang indeks baru

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih Domain.
3. Klik nama domain yang ingin Anda konfigurasi, lalu buka tab Opsi pengindeksan.
4. Pilih Kolom indeks baru untuk menambahkan spesifikasi bidang ke daftar.
5. Tentukan nama unik untuk bidang dan pilih jenis bidang. Nama bidang harus dimulai dengan huruf dan setidaknya 3 dan panjangnya tidak lebih dari 64 karakter. Karakter yang diizinkan adalah: a-z (huruf kecil), 0-9, dan `_` (garis bawah). Skor nama dicadangkan dan tidak dapat digunakan sebagai nama bidang.
6. Pilih detail kueri yang ingin Anda aktifkan untuk bidang tersebut. Untuk informasi selengkapnya, lihat [configure indexing options](#).
7. Pilih skema analisis yang akan digunakan untuk setiap bidang teks. Skema analisis menentukan opsi pemrosesan teks khusus bahasa yang digunakan selama pengindeksan. Secara default, bidang teks menggunakan skema `_en_default_` analisis. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Skema Analisis](#).
8. Tentukan nilai default untuk bidang (opsional). Nilai ini digunakan ketika tidak ada nilai yang ditentukan untuk bidang dalam data dokumen.
9. Secara opsional, tambahkan bidang tambahan di bidang Sumber.
10. Pilih Kirim.

#### Note

Ketika Anda menambahkan bidang atau memodifikasi bidang yang ada, Anda harus secara eksplisit mengeluarkan permintaan untuk mengindeks ulang data Anda ketika Anda selesai membuat perubahan konfigurasi. Untuk informasi selengkapnya, lihat [rebuild the index](#).

# Mengonfigurasi Bidang CloudSearch Indeks Amazon Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [DefineIndexField](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Menggunakan Bidang Dinamis di Amazon CloudSearch

Bidang dinamis menyediakan cara untuk mengindeks dokumen tanpa mengetahui terlebih dahulu bidang apa yang dikandungnya. Misalnya, pertimbangkan kasus di mana Anda ingin mencari satu set produk. Anda mungkin tidak mengetahui nama semua atribut produk yang mungkin di semua kategori produk, tetapi Anda dapat menyusun data Anda sehingga semua atribut berbasis teks disimpan di bidang yang diakhiri `_t`, dan semua nilai integer disimpan di bidang yang diakhiri `_i`. Dengan bidang dinamis, Anda dapat memetakan bidang atribut ke jenis bidang yang sesuai tanpa harus mengonfigurasi bidang untuk setiap atribut yang mungkin. Ini mengurangi jumlah konfigurasi yang perlu Anda lakukan di depan, dan menghilangkan kebutuhan untuk memodifikasi konfigurasi domain Anda setiap kali produk dengan atribut baru ditambahkan. Anda juga dapat menggunakan bidang dinamis untuk mengabaikan bidang baru dengan memetakannya ke bidang yang tidak dapat dicari atau dikembalikan.

### Topik

- [Mengkonfigurasi Bidang Dinamis di Amazon CloudSearch](#)
- [Menggunakan Bidang Dinamis untuk Mengabaikan Bidang yang Tidak Dikenali di Amazon CloudSearch](#)
- [Mencari Bidang Dinamis di Amazon CloudSearch](#)

## Mengkonfigurasi Bidang Dinamis di Amazon CloudSearch

Anda menetapkan bidang sebagai bidang dinamis dengan menentukan wildcard (\*) sebagai karakter pertama, terakhir, atau satu-satunya dalam nama bidang. Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (\*). Beberapa wildcard dan wildcard (\*) yang disematkan dalam string tidak didukung.

Nama bidang dinamis mendefinisikan pola. Wildcard cocok dengan nol atau lebih karakter arbitrer. Setiap bidang yang tidak dikenal yang cocok dengan pola tersebut dikonfigurasi dengan opsi pengindeksan bidang dinamis. Bidang indeks reguler lebih diutamakan daripada bidang dinamis. Jika nama bidang dokumen cocok dengan bidang indeks reguler dan pola bidang dinamis, itu dipetakan ke bidang indeks reguler.

#### Note

Opsi yang dapat Anda konfigurasi untuk bidang dinamis sama dengan [bidang statis](#). Demikian pula, nama bidang dokumen yang cocok dengan bidang dinamis harus memenuhi semua kriteria yang sama dengan nama bidang statis.

Misalnya, jika Anda menetapkan konvensi penamaan yang `_i` ditambahkan ke nama `int` bidang baru, Anda dapat menentukan bidang dinamis dengan pola `*_i` yang menyetel jenis bidang ke `int` dan mengonfigurasi sekumpulan opsi pengindeksan yang telah ditentukan untuk bidang baru. `int` Saat Anda menambahkan bidang seperti `review_rating_i`, itu dikonfigurasi sesuai dengan `*_i` opsi dan diindeks secara otomatis.

Jika bidang dokumen cocok dengan lebih dari satu pola bidang dinamis, pola pencocokan terpanjang digunakan. Jika pola memiliki panjang yang sama, bidang dinamis yang terjadi pertama kali ketika nama bidang diurutkan menurut abjad digunakan.

Anda dapat mendefinisikan\* sebagai bidang dinamis untuk mencocokkan bidang apa pun yang tidak dipetakan ke bidang yang ditentukan secara eksplisit atau pola bidang dinamis yang lebih panjang. Ini berguna jika Anda ingin mengabaikan bidang yang tidak dikenal. Untuk informasi selengkapnya, lihat [Menggunakan Bidang Dinamis untuk Mengabaikan Bidang yang Tidak Dikenali di Amazon CloudSearch](#).

Bidang dinamis dihitung terhadap jumlah total bidang yang ditentukan untuk domain. Sebuah domain dapat memiliki maksimal 200 definisi bidang, yang mencakup bidang dinamis. Namun, pola yang ditentukan oleh satu bidang dinamis biasanya cocok dengan beberapa bidang dokumen, sehingga jumlah total bidang dalam indeks Anda dapat melebihi 200. Untuk informasi selengkapnya, lihat [Batas](#). Saat menggunakan bidang dinamis, perlu diingat bahwa meningkatkan jumlah bidang dalam indeks Anda secara signifikan dapat memengaruhi kinerja kueri.

Menambahkan bidang baru ke konfigurasi domain Anda dapat memengaruhi cara bidang yang dihasilkan secara dinamis divalidasi selama pengindeksan. Jika validasi gagal, pengindeksan akan

gagal. Misalnya, jika Anda menentukan bidang dinamis yang disebut `*_new` dan mengunggah dokumen yang berisi bidang yang disebut `rating_new`, `rating_new` bidang tersebut akan ditambahkan ke indeks Anda. Jika Anda kemudian secara eksplisit mengonfigurasi bidang yang disebut `rating_new`, konfigurasi bidang baru itu akan digunakan untuk memvalidasi isi `rating_new` bidang dokumen Anda saat Anda menjalankan pengindeksan. Jika `*_new` dikonfigurasi sebagai `text` bidang dan Anda mengkonfigurasi `rating_new` sebagai `int` bidang, validasi akan gagal jika `rating_new` bidang yang ada berisi data non-integer.

Untuk informasi selengkapnya tentang mengonfigurasi bidang indeks, lihat [configure indexing options](#).

## Menggunakan Bidang Dinamis untuk Mengabaikan Bidang yang Tidak Dikenali di Amazon CloudSearch

Amazon CloudSearch mengharuskan Anda mengonfigurasi bidang indeks untuk setiap bidang yang terjadi dalam dokumen yang Anda indeks. Namun, dalam beberapa kasus, Anda ingin mengindeks sekumpulan bidang tertentu dan mengabaikan yang lainnya. Anda dapat menggunakan bidang dinamis untuk mengabaikan semua bidang yang tidak dikenal dengan mendefinisikan bidang literal yang disebut `*` dan menonaktifkan semua opsi pengindeksan untuk bidang tersebut. Setiap bidang yang tidak dikenal akan mewarisi opsi tersebut dan akan ditambahkan ke domain Anda; namun, konten bidang tidak akan dapat dicari atau dikembalikan, sehingga akan berdampak minimal pada ukuran indeks Anda. (Namun, mereka menghitung jumlah total bidang yang dikonfigurasi untuk domain.) Demikian pula, Anda dapat secara selektif mengabaikan bidang yang cocok dengan pola tertentu, seperti `*_n`.

Untuk mengabaikan bidang yang tidak dikenal

1. Konfigurasi bidang yang ingin Anda indeks, cari, atau kembalikan dalam hasil.
2. Tambahkan bidang dinamis yang cocok dengan bidang lain yang ditemukan dalam dokumen dan menonaktifkan semua opsi pengindeksan untuknya:
  - Tentukan `*` sebagai nama bidang, tanpa awalan atau string akhiran. (Anda juga dapat menentukan pola yang lebih spesifik untuk menonaktifkan bidang secara selektif.)
  - Atur jenis bidang ke `literal` dan nonaktifkan `search`, `facet`, dan `return` opsi. Perhatikan bahwa ukuran maksimum bidang literal adalah 4096 poin kode Unicode.

Karena pola bidang dinamis yang lebih panjang dicocokkan terlebih dahulu, Anda masih dapat menggunakan bidang dinamis untuk mengonfigurasi opsi untuk bidang yang ingin Anda gunakan.

Bidang apa pun yang tidak dipetakan ke bidang indeks reguler atau bidang dinamis yang lebih panjang akan cocok dengan pola\*.

### Note

Saat Anda membuat bidang dinamis dengan nama\*, itu berarti indeks Anda berpotensi berisi nama bidang yang valid. Ini juga berarti bahwa Anda dapat mereferensikan nama bidang yang valid dalam permintaan pencarian Anda, apakah itu benar-benar ada dalam indeks Anda atau tidak.

## Mencari Bidang Dinamis di Amazon CloudSearch

Anda dapat mereferensikan bidang yang dihasilkan secara dinamis berdasarkan nama dalam permintaan dan ekspresi pencarian Anda, sama seperti bidang lainnya. Misalnya, untuk mencari bidang yang dihasilkan secara dinamis `color_t` untuk `warnared`, Anda menggunakan parser kueri terstruktur:

```
q=color_t:'red'&q.parser=structured
```

Jika Anda telah menetapkan bidang dinamis catch-all (\*) untuk memetakan bidang apa pun yang tidak cocok dengan bidang reguler atau pola bidang dinamis yang lebih spesifik, Anda dapat menentukan nama bidang yang valid dalam permintaan pencarian, apakah bidang tersebut benar-benar ada dalam indeks Anda atau tidak.

Wildcard tidak didukung dalam nama bidang, jadi Anda tidak dapat mereferensikan bidang dinamis itu sendiri. Misalnya, menentukan `q=*_t:'red'` akan mengembalikan kesalahan.

Opsi bidang yang dihasilkan secara dinamis mewarisi dari konfigurasi bidang dinamis mengontrol bagaimana Anda dapat menggunakan bidang dalam permintaan penelusuran, misalnya, apakah Anda dapat mencarinya, mendapatkan aspek atau sorotan, menggunakannya untuk menyortir, atau mengembalikan hasilnya. Perhatikan bahwa bidang yang dihasilkan secara dinamis harus dicari secara eksplisit — bidang dinamis TIDAK termasuk dalam bidang yang dicari secara default saat Anda menggunakan parser kueri sederhana atau tidak menentukan bidang saat mencari dengan parser kueri terstruktur.

Anda dapat menentukan bidang dinamis sebagai sumber untuk bidang lain jika bidang target adalah array. Atribut sumber bidang mendukung wildcard, yang memungkinkan Anda menentukan pola yang cocok dengan sekelompok bidang dinamis. Misalnya, untuk mencari semua bidang yang

dihasilkan dari bidang `*_t` dinamis, Anda dapat membuat bidang yang dipanggil `all_t_fields` dan menyetel atribut sumbernya `*_t`. Ini menyalin isi dari semua bidang yang namanya berakhir dengan `_tall_t_fields`. Perhatikan, bagaimanapun, bahwa mencari bidang ini akan mencari semua bidang yang cocok dengan pola, tidak hanya bidang yang dihasilkan secara dinamis.

Untuk informasi selengkapnya tentang membuat dan mengirimkan permintaan pencarian, lihat

[Mencari Data Anda dengan Amazon CloudSearch](#)

## Mengkonfigurasi Skema Analisis Teks untuk Amazon CloudSearch

Amazon CloudSearch memungkinkan Anda mengkonfigurasi skema analisis khusus bahasa untuk masing-masing bidang `text`. `text-array` Skema analisis mengontrol bagaimana isi bidang diproses selama pengindeksan. Meskipun default untuk setiap bahasa bekerja dengan baik dalam banyak kasus, menyempurnakan opsi analisis memungkinkan Anda untuk mengoptimalkan hasil pencarian berdasarkan pengetahuan Anda tentang data yang Anda cari. Untuk daftar bahasa yang didukung, lihat [Bahasa yang Didukung](#).

Skema analisis menentukan bahasa teks yang akan diproses dan opsi analisis berikut:

- **Algorithmic stemming** — menentukan tingkat pembatalan algoritmik untuk dilakukan. Tingkat stemming yang tersedia bervariasi tergantung pada bahasa.
- **Kamus Tokenisasi Jepang** — menentukan penggantian tokenisasi algoritmik saat memproses bahasa Jepang. Kamus menentukan bagaimana set karakter tertentu harus dikelompokkan ke dalam kata-kata.
- **Kamus stemming** — menentukan penggantian untuk hasil stemming algoritmik. Kamus memetakan kata-kata terkait tertentu ke akar kata atau batang umum.
- **Stopwords** — menentukan kata-kata yang harus diabaikan selama pengindeksan dan pencarian.
- **Sinonim** — menentukan kata-kata yang memiliki arti yang sama dengan kata-kata yang muncul dalam data Anda dan harus menghasilkan hasil pencarian yang sama.

Selama pemrosesan teks, nilai bidang dan istilah pencarian diubah menjadi huruf kecil (dilipat huruf kecil), sehingga kata kunci, batang, dan sinonim tidak peka huruf besar/kecil. Untuk informasi selengkapnya tentang cara Amazon CloudSearch memproses teks selama pengindeksan dan saat menangani permintaan penelusuran, lihat [Pemrosesan Teks di Amazon CloudSearch](#).

Anda harus menentukan bahasa untuk setiap skema analisis dan mengkonfigurasi skema analisis untuk masing-masing `text text-array` bidang. Saat Anda mengkonfigurasi bidang melalui

CloudSearch konsol Amazon, skema analisis default ke skema analisis. `_en_default_` Jika Anda tidak menentukan opsi analisis untuk skema analisis, Amazon CloudSearch menggunakan opsi default untuk bahasa yang ditentukan. Untuk informasi tentang default untuk setiap bahasa, lihat.

### [Pengaturan Khusus Bahasa](#)

Cara termudah untuk menentukan skema analisis adalah melalui halaman Skema Analisis di CloudSearch konsol Amazon. Anda harus menerapkan skema analisis ke bidang agar dapat diterapkan. Anda dapat menerapkan skema analisis ke bidang dari halaman Opsi Pengindeksan. Anda juga dapat menentukan skema analisis dan mengonfigurasi skema analisis untuk setiap bidang melalui alat baris perintah dan AWS SDK.

Saat menerapkan skema analisis baru ke bidang indeks atau memodifikasi skema analisis yang sedang digunakan, Anda harus secara eksplisit [rebuild the index](#) agar perubahan tersebut tercermin dalam hasil penelusuran.

### Topik

- [Berasal di Amazon CloudSearch](#)
- [Stopwords di Amazon CloudSearch](#)
- [Sinonim di Amazon CloudSearch](#)
- [Mengkonfigurasi Skema Analisis Menggunakan Konsol Amazon CloudSearch](#)
- [Mengkonfigurasi Skema Analisis Menggunakan AWS CLI](#)
- [Mengkonfigurasi Skema Analisis Menggunakan AWS SDK](#)
- [Pengindeksan Bigram untuk China, Jepang, dan Korea di Amazon CloudSearch](#)
- [Menyesuaikan Tokenisasi Jepang di Amazon CloudSearch](#)

## Berasal di Amazon CloudSearch

Stemming adalah proses pemetaan kata-kata terkait ke batang umum. Batang biasanya merupakan akar atau kata dasar dari mana varian diturunkan. Misalnya, lari adalah batang berlari dan berlari. Stemming dilakukan selama pengindeksan serta pada waktu kueri. Stemming mengurangi jumlah istilah yang disertakan dalam indeks, dan memfasilitasi kecocokan ketika istilah pencarian adalah varian dari istilah yang muncul dalam konten yang dicari. Misalnya, jika Anda memetakan istilah yang berjalan ke stem run dan kemudian mencari running, permintaan tersebut cocok dengan dokumen yang berisi run dan juga running.



Amazon CloudSearch mendukung kamus stemming algoritmik dan stemming eksplisit. Anda mengonfigurasi stemming algoritmik dengan menentukan tingkat stemming yang ingin Anda gunakan. Tingkat pembatalan algoritmik yang tersedia bervariasi tergantung pada bahasa:

- tidak ada—nonaktifkan pembendungan algoritmik
- minimal—lakukan pembendungan dasar dengan menghapus sufiks jamak
- ringan—targetkan infleksi kata benda/kata sifat yang paling umum dan sufiks turunan
- infleksi batang penuh—agresif dan sufiks

Selain mengontrol tingkat pembatalan algoritmik yang dilakukan, Anda dapat menentukan kamus stemming yang memetakan kata-kata terkait tertentu ke batang umum. Anda menentukan kamus sebagai objek JSON yang berisi kumpulan pasangan string:value yang memetakan istilah ke batangnya, misalnya, `{"term1": "stem1", "term2": "stem2", "term3": "stem3"}`. Kamus stemming diterapkan selain stemming algoritmik apa pun. Hal ini memungkinkan Anda untuk mengesampingkan hasil pembatalan algoritmik untuk memperbaiki kasus-kasus tertentu dari overstemming atau understemming. Ukuran maksimum kamus stemming adalah 500 KB. Entri kamus stemming harus huruf kecil.

Anda menggunakan `StemmingDictionary` kunci untuk menentukan kamus stemming kustom dalam skema analisis. Karena Anda meneruskan kamus ke Amazon CloudSearch sebagai string, Anda harus menghindari semua tanda kutip ganda dalam string. Misalnya, skema analisis berikut mendefinisikan batang untuk berlari dan melompat:

```
{
  "AnalysisSchemeName": "myscheme",
  "AnalysisSchemeLanguage": "en",
  "AnalysisOptions": {
    "AlgorithmicStemming": "light",
    "StemmingDictionary": "{\"running\": \"run\\\", \"jumping\": \"jump\\\"}"
  }
}
```

Jika Anda tidak menentukan tingkat stemming algoritmik atau kamus stemming dalam skema analisis Anda, Amazon CloudSearch menggunakan tingkat pembatalan algoritmik default untuk bahasa yang ditentukan. Meskipun stemming dapat membantu pengguna menemukan dokumen yang relevan yang mungkin dikecualikan dari hasil penelusuran, overstemming dapat mengakibatkan terlalu banyak kecocokan dengan relevansi yang dipertanyakan. Tingkat default stemming algoritmik yang dikonfigurasi untuk setiap bahasa berfungsi dengan baik untuk sebagian besar kasus penggunaan.

Secara umum, yang terbaik adalah memulai dengan default dan kemudian membuat penyesuaian untuk mengoptimalkan relevansi hasil pencarian untuk kasus penggunaan Anda. Untuk informasi tentang default untuk setiap bahasa, lihat. [Pengaturan Khusus Bahasa](#)

## Stopwords di Amazon CloudSearch

Stopwords adalah kata-kata yang biasanya harus diabaikan baik selama pengindeksan dan pada waktu pencarian karena mereka tidak signifikan atau sangat umum sehingga memasukkannya akan menghasilkan sejumlah besar kecocokan.

Selama pengindeksan, Amazon CloudSearch menggunakan kamus stopwords saat memproses `text` dan `text-array` bidang. Dalam kebanyakan kasus, stopwords tidak termasuk dalam indeks. Kamus stopwords juga digunakan untuk memfilter permintaan pencarian.

Kamus stopwords adalah array istilah JSON, misalnya, `["a", "an", "the", "of"]` Kamus stopwords harus secara eksplisit mencantumkan setiap kata yang ingin Anda abaikan. Wildcard dan ekspresi reguler tidak didukung.

Anda menggunakan Stopwords kunci untuk menentukan kamus stopwords kustom dalam skema analisis. Karena Anda meneruskan kamus ke Amazon CloudSearch sebagai string, Anda harus menghindari semua tanda kutip ganda dalam string. Misalnya, skema analisis berikut mengkonfigurasi stopwords `a`, `an`, dan:

```
{
  "AnalysisSchemeName": "myscheme",
  "AnalysisSchemeLanguage": "en",
  "AnalysisOptions": {
    "Stopwords": "[\"a\", \"an\", \"the\"]"
  }
}
```

Jika Anda tidak menentukan kamus stopwords dalam skema analisis Anda, Amazon CloudSearch menggunakan kamus stopwords default untuk bahasa yang ditentukan. Stopword default yang dikonfigurasi untuk setiap bahasa berfungsi dengan baik untuk sebagian besar kasus penggunaan. Secara umum, yang terbaik adalah memulai dengan default dan kemudian membuat penyesuaian untuk mengoptimalkan relevansi hasil pencarian untuk kasus penggunaan Anda. Untuk informasi tentang default untuk setiap bahasa, lihat. [Pengaturan Khusus Bahasa](#)

## Sinonim di Amazon CloudSearch

Anda dapat mengonfigurasi sinonim untuk istilah yang muncul di data yang Anda cari. Dengan begitu, jika pengguna mencari sinonim daripada istilah yang diindeks, hasilnya akan menyertakan dokumen yang berisi istilah yang diindeks. Misalnya, Anda dapat menentukan sinonim kustom untuk melakukan hal berikut:

- Petakan kesalahan ejaan umum ke ejaan yang benar
- Mendefinisikan istilah yang setara, seperti `film` dan `movie`
- Memetakan istilah umum ke istilah yang lebih spesifik, seperti `fish` dan `barracuda`
- Memetakan beberapa kata ke satu kata atau sebaliknya, seperti `tool box` dan `toolbox`

Ketika Anda mendefinisikan sinonim, sinonim ditambahkan ke indeks di mana pun token dasar terjadi. Misalnya, jika Anda mendefinisikan `fish` sebagai sinonim dari `barracuda`, istilah `fish` ditambahkan ke setiap dokumen yang berisi istilah `barracuda`. Menambahkan sejumlah besar sinonim dapat meningkatkan ukuran indeks serta latensi kueri — sinonim meningkatkan jumlah kecocokan dan semakin banyak kecocokan, semakin lama waktu yang dibutuhkan untuk memproses hasilnya.

Kamus sinonim digunakan selama pengindeksan untuk mengonfigurasi pemetaan untuk istilah yang terjadi di bidang teks. Tidak ada pemrosesan sinonim yang dilakukan pada permintaan pencarian. Secara default, Amazon CloudSearch tidak mendefinisikan sinonim apa pun.

Anda dapat menentukan sinonim dengan dua cara:

- Sebagai kelompok konflasi di mana setiap istilah dalam kelompok dianggap sebagai sinonim dari setiap istilah lain dalam kelompok.
- Sebagai alias untuk istilah tertentu. Alias dianggap sebagai sinonim dari istilah yang ditentukan, tetapi istilah tersebut tidak dianggap sebagai sinonim dari alias.

Kamus sinonim ditentukan sebagai objek JSON yang mendefinisikan kelompok sinonim dan alias. `groups` Nilainya adalah array array, di mana setiap sub-array adalah grup konflasi. `aliases` Nilai adalah objek yang berisi kumpulan pasangan `string:value` di mana `string` menentukan istilah dan `array` nilai menentukan masing-masing sinonim untuk istilah itu. Contoh berikut mencakup grup konflasi dan alias:

```
{
```

```
"groups": [ ["1st", "first", "one"], ["2nd", "second", "two"] ],
"aliases": { "youth": ["child", "kid", "boy", "girl"],
             "adult": ["men", "women"] }
}
```

Kedua grup dan alias mendukung sinonim multiword. Dalam contoh berikut, sinonim multiword digunakan dalam grup konflasi serta alias:

```
{
  "groups": [ ["tool box", "toolbox"], ["band saw", "bandsaw"] ],
  "aliases": { "workbench": ["work bench"]}
}
```

Anda menggunakan Synonyms kunci untuk menentukan kamus sinonim kustom dalam skema analisis. Karena Anda meneruskan kamus ke Amazon CloudSearch sebagai string, Anda harus menghindari semua tanda kutip ganda dalam string. Misalnya, skema analisis berikut mengonfigurasi alias untuk istilah pemuda:

```
{
  "AnalysisSchemeName": "myscheme",
  "AnalysisSchemeLanguage": "en",
  "AnalysisOptions": {
    "Synonyms": "{ \"aliases\": { \"youth\": [ \"child\", \"kid\" ] } }"
  }
}
```

## Mengkonfigurasi Skema Analisis Menggunakan Konsol Amazon CloudSearch

Anda dapat menentukan skema analisis dari panel Skema Analisis di CloudSearch konsol Amazon.

Untuk mendefinisikan skema analisis

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Dari panel navigasi kiri, pilih Domain.
3. Pilih nama domain Anda untuk membuka konfigurasinya.
4. Buka tab Opsi pencarian lanjutan.
5. Di panel Skema analisis, pilih Tambahkan skema analisis.
6. Tentukan nama untuk skema analisis dan pilih bahasa.

7. Pilih Berikutnya.
8. Dalam tiga langkah berikutnya, konfigurasi opsi stopword, stemming, dan sinonim teks skema. Anda dapat mengonfigurasi kata kunci, batang, dan sinonim individual, atau mengedit kamus yang ditampilkan secara langsung. Kamus diformat dalam JSON. Stopwords ditentukan sebagai array string. Batang ditentukan sebagai objek yang berisi satu atau lebih pasangan key:value. Alias sinonim juga ditentukan sebagai objek JSON dengan satu atau pasangan pindahkan kunci:nilai, di mana nilai alias ditentukan sebagai array string. Sebuah grup sinonim ditentukan sebagai array JSON. (Kamus sinonim adalah array array.)

Jika Anda memilih bahasa Jepang sebagai bahasa, Anda juga memiliki opsi untuk menentukan kamus tokenisasi kustom yang mengesampingkan tokenisasi default frasa tertentu. Untuk informasi selengkapnya, lihat [Menyesuaikan Tokenisasi Jepang](#).

9. Pada halaman ringkasan, tinjau konfigurasi skema analisis dan pilih Simpan.

#### Important

Untuk menggunakan skema analisis, Anda harus menerapkannya ke satu atau lebih text atau text-array bidang dan membangun kembali indeks. Anda dapat mengonfigurasi skema analisis bidang dari tab Opsi pengindeksan. Untuk membangun kembali indeks Anda, pilih Tindakan, Jalankan pengindeksan.

## Mengkonfigurasi Skema Analisis Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch define-analysis-scheme` perintah untuk menentukan opsi pemrosesan teks khusus bahasa, termasuk opsi stemming, stopword, dan sinonim. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Anda menentukan skema analisis sebagai bagian dari konfigurasi masing-masing text atau text-array bidang. Untuk informasi selengkapnya, lihat [configure indexing options](#).

Untuk mendefinisikan skema analisis

- Jalankan `aws cloudsearch define-analysis-scheme` perintah dan tentukan `--analysis-scheme` opsi dan objek JSON yang berisi opsi analisis Anda. Skema analisis harus JSON yang valid. Kunci opsi analisis dan pasangan nilai harus diapit tanda kutip, dan semua

tanda kutip dalam nilai opsi harus diloloskan dengan garis miring terbalik. Untuk format opsi analisis, lihat [define-analysis-scheme](#) di Referensi AWS CLI Perintah. Lihat [Mengkonfigurasi Skema Analisis](#) untuk informasi selengkapnya tentang menentukan opsi stemming, stopwords, dan sinonim.

Jika Anda menentukan Japanese (ja) sebagai bahasa, Anda juga memiliki opsi untuk menentukan kamus tokenisasi kustom yang mengesampingkan tokenisasi default frasa tertentu. Untuk informasi selengkapnya, lihat [Menyesuaikan Tokenisasi Jepang](#).

### Tip

Cara termudah untuk mengkonfigurasi skema analisis dengan AWS CLI adalah dengan menyimpan skema analisis dalam file teks dan menentukan file itu sebagai `--analysis-scheme` nilai. Ini memungkinkan Anda untuk memformat skema sehingga lebih mudah dibaca. Misalnya, skema berikut mendefinisikan skema analisis bahasa Inggris yang disebut `myscheme` yang menggunakan stemming algoritmik ringan dan mengkonfigurasi dua stopwords:

```
{
  "AnalysisSchemeName": "myscheme",
  "AnalysisSchemeLanguage": "en",
  "AnalysisOptions": {
    "AlgorithmicStemming": "light",
    "Stopwords": "[\"a\", \"the\"]"
  }
}
```

Jika Anda menyimpan skema ini dalam file teks yang disebut `myscheme.txt`, Anda dapat meneruskan file sebagai nilai `--analysis-scheme` parameter:

```
aws cloudsearch define-analysis-scheme --region us-east-1 --domain-name
movies --analysis-scheme file://myscheme.txt
```

### Important

Untuk menggunakan skema analisis, Anda harus menerapkannya ke satu atau lebih `text` atau `text-array` bidang dan membangun kembali indeks. Anda dapat mengonfigurasi

skema analisis bidang dengan `aws cloudsearch define-index-field` perintah. Untuk membangun kembali indeks, hubungi `aws cloudsearch index-documents`.

## Mengkonfigurasi Skema Analisis Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk [DefineAnalysisScheme](#). Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

### Important

Untuk menggunakan skema analisis, Anda harus menerapkannya ke satu atau lebih `text` atau `text-array` bidang dan membangun kembali indeks. Anda dapat mengonfigurasi skema analisis bidang dengan metode bidang indeks definisi. Untuk membangun kembali indeks Anda, Anda menggunakan metode dokumen indeks.

## Pengindeksan Bigram untuk China, Jepang, dan Korea di Amazon CloudSearch

Tionghoa, Jepang, dan Korea tidak memiliki batasan kata yang eksplisit. Cukup mengindeks karakter individu (unigram) dapat menghasilkan kecocokan yang tidak terlalu relevan dengan kueri penelusuran. Salah satu solusinya adalah mengindeks bigram. Bigram adalah setiap urutan dari dua karakter yang berdekatan dalam sebuah string. Misalnya, contoh berikut menunjukkan bigram untuk string

我的氣墊船裝滿了鱈魚

我的 的氣 氣墊 墊船 船裝 裝滿 滿了 了鱈 鱈魚

Meskipun pengindeksan bigram dapat meningkatkan kualitas hasil pencarian, perlu diingat bahwa itu dapat secara signifikan meningkatkan ukuran indeks Anda.

Untuk mengindeks bigram untuk bahasa China, Jepang, dan Korea

1. Buat skema analisis teks dan atur bahasa ke beberapa bahasa (mu1).
2. Konfigurasi kolom indeks yang berisi data CJK untuk menggunakan skema analisis multi-bahasa Anda.

Saat Anda menetapkan skema analisis yang menetapkan bahasa bidangmu1, Amazon CloudSearch secara otomatis menghasilkan bigram untuk semua teks Mandarin, Jepang, dan Korea di dalam bidang tersebut.

Untuk informasi selengkapnya tentang membuat dan menggunakan skema analisis, lihat [Mengkonfigurasi Skema Analisis](#).

Jika Anda mengindeks konten Jepang, Anda mungkin juga tertarik untuk menggunakan kamus tokenisasi kustom dengan prosesor bahasa Jepang standar. Untuk informasi selengkapnya, lihat [Menyesuaikan Tokenisasi Jepang](#).

## Menyesuaikan Tokenisasi Jepang di Amazon CloudSearch

Jika Anda memerlukan kontrol lebih besar atas cara Amazon CloudSearch memberi token bahasa Jepang, Anda dapat menambahkan kamus tokenisasi Jepang khusus ke skema analisis Anda. Mengkonfigurasi kamus tokenisasi kustom memungkinkan Anda untuk mengganti bagaimana entri tertentu diberi token oleh prosesor bahasa Jepang standar. Ini dapat meningkatkan akurasi hasil pencarian dalam beberapa kasus, terutama ketika Anda perlu mengindeks dan mengambil frase khusus domain.

Kamus tokenisasi adalah kumpulan entri di mana setiap entri menentukan satu set karakter, bagaimana karakter harus diberi token, bagaimana setiap token harus diucapkan (bacaan), dan tag part-of-speech Anda menentukan kamus sebagai array, dan setiap entri kamus adalah array string. Entri adalah dari bentuk berikut:

```
["<text>","<token 1> ... <token n>","<reading 1> ... <reading n>","<part-of-speech tag>"]
```

Anda harus menentukan bacaan untuk setiap token dan part-of-speech tag untuk entri. Lihat bagian [Tag Part-of-Speech Jepang](#) tag ucapan yang diperlakukan sebagai stopword.

Anda menggunakan `JapaneseTokenizationDictionary` kunci untuk menentukan kamus tokenisasi kustom dalam skema analisis. Karena Anda meneruskan kamus tokenisasi ke Amazon



CloudSearch sebagai string, Anda harus menghindari semua tanda kutip ganda dalam string. Misalnya, kamus dalam skema analisis berikut menentukan penggantian segmentasi untuk senyawa Kanji dan Katakana, dan pembacaan khusus untuk nama yang tepat:

```
"AnalysisSchemeName": "jascheme",
"AnalysisSchemeLanguage": "ja",
"AnalysisOptions": {
  "Stopwords": "[\"a\", \"the\"]",
  "AlgorithmicStemming": "full",
  "JapaneseTokenizationDictionary": "[
    [\"日本経済新聞\", \"日本 経済 新聞\", \"ニホン ケイザイ シンブン\", \"カスタム名詞\"],
    [\"トートバッグ\", \"トート バッグ\", \"トート バッグ\", \"かずカナ名詞\"],
    [\"朝青龍\", \"朝青龍\", \"アサショウリュウ\", \"カスタム人名\"]
  ]"
}
```

Saat mengonfigurasi skema analisis dengan AWS CLI, Anda dapat menyimpan skema analisis dalam file teks dan menentukan file itu sebagai `--analysis-scheme` nilainya. Ini memungkinkan Anda untuk memformat skema sehingga lebih mudah dibaca. Misalnya, jika Anda menyimpan skema `jascheme` analisis dalam file bernama `jascheme.txt`, Anda dapat meneruskan file itu saat Anda menelepon `aws cloudsearch define-analysis-scheme`:

```
aws cloudsearch define-analysis-scheme --region us-east-1 --domain-name
mydomain --analysis-scheme file://jascheme.txt
```

Untuk informasi selengkapnya tentang membuat dan menggunakan skema analisis, lihat [Mengkonfigurasi Skema Analisis](#).

## Tag Part-of-Speech Jepang di Amazon CloudSearch

Saat Anda menggunakan kamus tokenisasi khusus untuk bahasa Jepang, Anda menentukan part-of-speech tag untuk setiap entri. Jika part-of-speech tag cocok dengan salah satu tag yang dikonfigurasi sebagai tag berhenti, entri diperlakukan sebagai stopword.

Tabel berikut menunjukkan bagian dari tag ucapan yang dikonfigurasi sebagai tag berhenti di Amazon CloudSearch.

## Hentikan Tag

Tag	Bagian pidato	Deskripsi
助動詞	Kata kerja bantu	Kata kerja yang menambahkan makna fungsional atau tata bahasa pada klausa di mana ia muncul.
接続詞	Konjungsi	Konjungsi yang dapat terjadi secara independen.
フィラー	Pengisi	Aizuchi yang terjadi selama percakapan atau suara dimasukkan sebagai pengisi.
非言語音	Non-verbal	Suara non-verbal.
その他-間投	Kata seru lainnya	Kata-kata yang sulit diklasifikasikan sebagai kata benda-sufiks atau partikel akhir kalimat.
助詞-副詞化	Partikel-adnominalizer	Kata “ni” dan “to” yang muncul mengikuti kata benda dan kata keterangan.
助詞-連体化	Partikel-adnominalizer	Kata “tidak” yang melekat pada kata benda dan memodifikasi kata-kata non-infleksional.

Tag	Bagian pidato	Deskripsi
<b>助詞-副助詞</b>	Partikel-adverbial	Kata keterangan yang digunakan untuk menunjukkan posisi, arah gerakan, dan sebagainya.
助詞-副助詞/並立助詞/終助詞	Partikel-adverbial/ konjungtif/final	Partikel “ka” ketika tidak diketahui apakah itu adverbial, konjungtif, atau kalimat final.
<b>助詞-格助詞-連語</b>	P article-case-compound	Senyawa partikel dan kata kerja yang terutama berperilaku seperti partikel kasus.
<b>助詞-格助詞-一般</b>	P article-case-misc	Partikel kasus.
<b>助詞-格助詞-引用</b>	P article-case-quote	Kata “ke” yang muncul setelah kata benda, ucapan seseorang, tanda kutip, ekspresi keputusan dari rapat, alasan, penilaian, dugaan, dan sebagainya.
<b>助詞-格助詞</b>	Kasus partikel	Partikel kasus di mana subklasifikasi tidak terdefinisi.
<b>助詞-接続助詞</b>	Partikel-konjungtif	Partikel konjungtif.

Tag	Bagian pidato	Deskripsi
助詞-並立助詞	Partikel-koordinat	Partikel koordinat.
助詞-係助詞	Ketergantungan partikel	Partikel ketergantungan.
助詞-終助詞	Partikel-final	Partikel akhir.
助詞-間投助詞	Partikel-interjektif	Partikel dengan peran tata bahasa interjektif.
助詞-特殊	Partikel-khusus	Sebuah partikel yang tidak cocok dengan salah satu klasifikasi lainnya. Ini termasuk partikel yang digunakan dalam Tanka, Haiku, dan puisi lainnya.
助詞	Partikel	Partikel yang tidak diklasifikasikan.
記号-括弧閉	Simbol-Close_bracket	Tutup braket:].
記号-読点	Simbol-koma	Koma:.,
記号-一般	Simbol-lain-lain	Simbol umum tidak dalam salah satu kategori lainnya.
記号-括弧開	Simbol-Open_Bracket	Braket terbuka: [.

Tag	Bagian pidato	Deskripsi
記号-句点	Periode simbol	Periode dan pemberhentian penuh.
記号-空白	Simbol-ruang	Spasi putih lebar penuh.
記号	Simbol	Simbol yang tidak diklasifikasikan.

## Pemrosesan Teks di Amazon CloudSearch

Selama pengindeksan, Amazon CloudSearch memproses text dan text-array bidang sesuai dengan skema analisis yang dikonfigurasi untuk bidang untuk menentukan istilah apa yang akan ditambahkan ke indeks. Sebelum opsi analisis diterapkan, teks diberi token dan dinormalisasi.

Selama tokenisasi, aliran teks dalam bidang dibagi menjadi token terpisah pada batas yang dapat dideteksi menggunakan aturan pemutusan kata yang didefinisikan dalam algoritma Segmentasi Teks Unicode. Untuk informasi selengkapnya, lihat [Segmentasi Teks Unicode](#).

Menurut aturan break word, string yang dipisahkan oleh spasi seperti spasi dan tab diperlakukan sebagai token terpisah. Dalam banyak kasus, tanda baca dijatuhkan dan diperlakukan sebagai spasi putih. Misalnya, string dibagi pada tanda hubung (-) dan simbol at (@). Namun, periode yang tidak diikuti oleh spasi putih dianggap sebagai bagian dari token.

Perhatikan bahwa string tidak dibagi pada batas kasus— CamelCasestring tidak diberi token.

Selama normalisasi, karakter huruf besar dikonversi ke huruf kecil. Aksen biasanya ditangani sesuai dengan opsi stemming yang dikonfigurasi dalam skema analisis lapangan. (Skema analisis default untuk bahasa Inggris menghilangkan aksen.)

Setelah tokenisasi dan normalisasi selesai, opsi stemming, stopword, dan sinonim yang ditentukan dalam skema analisis diterapkan.

Saat Anda mengirimkan permintaan pencarian, teks yang Anda cari mengalami pemrosesan teks yang sama sehingga dapat dicocokkan dengan istilah yang muncul di indeks. Namun, tidak ada

analisis teks yang dilakukan pada istilah pencarian saat Anda melakukan pencarian awalan. Ini berarti bahwa pencarian awalan yang diakhiri s biasanya tidak akan cocok dengan versi tunggal istilah saat stemming diaktifkan. Ini bisa terjadi untuk istilah apa pun yang berakhirs, bukan hanya jamak. Misalnya, jika Anda mencari actor bidang dalam data film sampelAnders, ada tiga film yang cocok. Jika Anda mencariAndeɾ\*, Anda mendapatkan film-film itu serta beberapa lainnya. Namun, jika Anda mencari tidak Anders\* ada kecocokan. Ini karena istilah disimpan dalam indeks sebagaiander, anders tidak muncul dalam indeks.

Jika stemming mencegah pencarian wildcard mengembalikan semua kecocokan yang relevan, Anda dapat menekan stemming untuk bidang teks dengan menyetel AlgorithmicStemming opsi ke none, atau Anda dapat memetakan data ke bidang alih-alih bidang. literal text

## Topik

- [Bahasa yang Didukung di Amazon CloudSearch](#)
- [Pengaturan Pemrosesan Teks Khusus Bahasa di Amazon CloudSearch](#)

## Bahasa yang Didukung di Amazon CloudSearch

Arabic (ar)	Armenian (hy)	Basque (eu)
Bulgarian (bg)	Catalan (ca)	Chinese - Simplified (zh-Hans)
Chinese - Traditional (zh-Hant)	Czech (cs)	Danish (da)
Dutch (nl)	English (en)	Finnish (fi)
French (fr)	Galician (gl)	German (de)
Greek (el)	Hindi (hi)	Hebrew (he)
Hungarian (hu)	Indonesian (id)	Irish (ga)
Italian (it)	Japanese (ja)	Korean (ko)
Latvian (lv)	Multiple (mul)	Norwegian (no)
Persian (fa)	Portuguese (pt)	Romanian (ro)
Russian (ru)	Spanish (es)	Swedish (sv)



## Skema analisis default: `_eu_default_`

- Opsi stemming algoritmik: `full`
- Kamus stopword default:

al anitz arabera asko baina bat batean batek bati batzuei batzuei batzuek batzuetan batzuk bera beraiek berau berauek berori beroriek beste bezala da dago dira ditu du dute edo egin ere eta eurak ez gainera gu gutxi guzti haiei haiek haietan hainbeste hala han handik hango hara hari hark hartan hau hauei hauek hauetan hemen hemendik hemengo hi hona honek honek honetan honi hor hori horiei horiek horko horra horra horrek horrela horri horri hortik hura izan ni noiz nola non nondik nongo nor nora ze zein zen zenbait zenbat zer zergatik ziren zituen zu zuek zuen zuten

## Bulgaria (bg)

Opsi stemming algoritmik: `light`

## Skema analisis default: `_bg_default_`

- Pembendungan algoritmik: `light`
- Kamus stopword default:

а аца ако ала бе бебее би бибила били били било буимо бъдае бъде биха в ас вас вае верорono вема вие винаи вси е всеки всимки всимко всека във въмреки върху Г и главно го да да дали до докато доко дори дори доса дин ето на саца саедно иради масега иатова мамова мамо ие или или има има има иска иска кана какака како като като като като като като ога когато което които кобойто колко кото кото където където къМи ме меду мен мени мнoнина мога моля та му н на нае нанаса направи направи натреример нас не него нея ни ние никосниoниo но но нокои некои нема обaме около освен особено от отгоре отново ое ие тацо иo товеме овеме овето тоне оради косле коти трaви треби креби треди мрецури тЪК търво с са само се сега си скоро слее сме сторен срен срен сте със със със сѐаo така такива така в таа твoсте тени ти то то това тогава томи толкова торно торно трябва тук тѐа тох ух харесва

## Catalan (ca)

Opsi stemming algoritmik: `full`

Filter elision diaktifkan



## Skema analisis default: `_ca_default_`

- Pembendungan algoritmik: `full`
- Kamus stopword default:

a abans ací ah així això al als aleshores algun alguna algunes alguns alhora allà allí allò altra altre  
altre altres amb ambdós ambdues apa aquell aquella aquelles aquest aquestes aquests  
aquí baix cada cadascú cadascuna cadascunes cadascuns com contra d'unes d'uns dalt de del  
dels des després dins dintre donat doncs durant e eh el els em en encara ens entre érem eren  
éreu es és esta estàvem estàvem estàvem estàveu esteu et etc ets fins fora gairebé ha han has  
havia he hem heu ho hi i iigual ja l'hi la les li li'n llavors m'he ma mal malgrat mateix mateixa  
mateixos me mentre més meu meus meva meves molt molta moltes molts mon mons n'he n'hi ne  
ni no nogensmenys només nosaltres nostra nostre nostres o oh oi on pas pel pels per però perquè  
poca pocs poques potser propi qual quals quan quant que què quelcom qui quin quina quines  
quins s'ha s'han sa semblant semblant ses seu seus seva seva seves si sobre sobre sobretot sóc  
solament solament son son son son sota sou t'ha t'han t'he ta tal tampoc tan tan tanta tantes teu  
teva teves ton ton tot tota totes tots un una unes uns us va vaig vam van vas veu vosaltres vostra  
vostre vostres

## Mandarin - Sederhana (Zh-hans)

Pembatalan algoritmik tidak didukung

Kamus stemming tidak didukung

Skema analisis default: `_zh-Hans_default_`

## Tionghoa - Tradisional (Zh-hant)

Pembatalan algoritmik tidak didukung

Kamus stemming tidak didukung

Skema analisis default: `_zh-Hant_default_`

## Ceko (cz)

Opsi stemming algoritmik: `light`

Skema analisis default: `_cs_default_`

- Pembendungan algoritmik: `light`
- Kamus stopwords default:

a s k o i u v z dnes cz tímto budeš budem byli jseš můj svým ta tomta tohle tuto tyto jej zda proč máte tato kam tohoto kdo kteří mi nám tom tomuto mít nic proto kterou byla toho protože asi ho naši napiši což tím takže svých její svými jste aj tu tedy teto bylo kde ke pravé ji nad nejsou či pod téma mezi přes ty pak vám ani když však neg jsem tento článku články aby jsme před pta jejich byl ještě až bez také pouze první vaše která nás nový tipy pokud může strana jeho své jiné zprávy nové není vás jen podle zde už být více bude již než který by které co nebo ten tak má při od po jsou jak další ale si se ve to jako za zpět ze do pro je na atd atp jakmile přičemž já on ona ono oni ony my vy jí ji mě mne jemu tomu těm těmu němuž jehož již jelikož jakož jakož ž načěž

## Denmark (da)

Opsstemming algoritmik: `full`

Skema analisis default: `_da_default_`

- Pembendungan algoritmik: `full`
- Kamus stopwords default:

og i jeg det at en den til er som på de med han af for ikke der var mig sig men et har om vi min havde ham hun nu over da fra du ud sin dem os op man hans hvor or hvad skal selv her alle vil blev kunne ind når være dog noget ville jo deres efter ned skulle le denne end dette mit også under have dig anden hende mine alt meget sit sine vor mod disse hvis din nogle hos blive mange ad bliver hendes været thi jer sådan

## Belanda (nl)

Opsstemming algoritmik: `full`

Skema analisis default: `_nl_default_`

- Pembendungan algoritmik: `full`
- Kamus stopwords default:

en van ik te dat die in a hij het niet zijn is was op aan met als voor had er maar om hem dan zou of wat saya men dit zo door over ze zich bij ook tot je mij uit der daar haar haar haar heeft hebben

deze u want nog zal me zij nu ge geen omden iets worden toch al waren veel meer doen toen moet ben zonder kan hun dus alles onder ja eens hier wie werd doch wordt wezen kunnen ons zelf tegen na reeds wil kon niets uw iemand geweest andere

- Kamus stemming default:

bunga bromfiet bromfiets ei eier kind kinder

## Bahasa Inggris (id)

Opsi stemming algoritmik: `|| minimal light full`

Skema analisis default: `_en_default_`

- Pembendungan algoritmik: `full`
- Kamus stopword default:

a dan seperti yang ada tetapi karena jika masuk ke dalam tidak ada atau sedemikian rupa sehingga mereka kemudian di sana ini mereka ini adalah kehendak dengan

## Finlandia (fi)

Opsi stemming algoritmik: `| light full`

Skema analisis default: `_fi_default_`

- Pembendungan algoritmik: `light`
- Kamus stopword default:

olla olen olet on olemme olette ovat ole oli olisi olisit olisin olisimme olisitte olisivat olit olin olimme olitte olivat ollut olleet en et ei emme ette eivät minä minun minua minussa minussa minusta minulla minulta minulle sinä sinun sinut sinussa sinusta sinuun sinulla sinulta sinulle hänen häntä hänessä hänestä häneen hänellä hänellä hänen häntä meidät meitä meistä meillä meiltä meille te teidän teidän teitä teissä teihin teillä teiltä teille he heidän heitä heistä heihin heillä heiltä heiltä heille tämä tämän tätä tätä tälle tälle tänä tuo tuon tuotä tuossa tuosta tuosta tuohon tuolla tuolta tuolle tuona tuoksi se sen siitä siihen sillä siltä siltä siltä sille sinä siä siä nämä näiden näitä näissä näillä näiltä näitä näiksi nuo noiden noita noissa noista noihin noilla noilta noilta noille noina noiksi ne niiden niissä niistä niihin niillä niillä niille niille niiksi kuka kenen kenet ketä kenessä keneen kenellä keneltä kenelle kenenä keneksi ketkä keiden ketkä keitä keissä keistä keihin keillä keiltä

keille keinä keiksi mungkin minkä minkä mitä mihin millä millä mille minä miksi mitkä joka jota jossa josta johon jolla jolta jolle jona joksi jotka joiden joita joissa joista joihin joilla joille joina joina joiksi että ja jos koska kuin mutin niin sekä tai vaan vai vaikka kanssa noin poikki yli kun niin nyt itse

## Perancis (fr)

Opsi stemming algoritmik: `|| minimal light full`

Filter elision diaktifkan

Skema analisis default: `_fr_default_`

- Pembendungan algoritmik: `minimal`
- Kamus stopword default:

au aux avec ce ces dans de des du elle en et eux il je la le leur lui ma mais me même mes moi mon ne nos notre nous on ou par pas pour qu que qui sa se ses son sur ta te tes toi ton tu un une vos votre vous c d j l à m n s t y été étées étées étées étant suis es est sommes êtes sont serai seras sera serons serez seront serais serait serions seraient étai était étions étiez étaiet fus fut fûmes fûtes furent sois soit soyons soyez soient fusse fusses fussiez fussiez fussiez fussent ayant eu eue eues eus ai sebai avons avez ont aurai aura aura auron aurez auront aurai aurait aurions auriez auraient avais avait avions aviez avaient eut eûmes eûtes eurent aie aies ait ayons ayez aient eusse eusses eussions eussiez eussent ceci celà cet cette ici ils les leurs quel quel quels quelle quelles sans soi

## Galicia (gl)

Opsi stemming algoritmik: `| minimal full`

Skema analisis default: `_gl_default_`

- Pembendungan algoritmik: `minimal`
- Kamus stopword default:

# kata kunci galican a aínda alí aquel aquela aquelas aqueles aquilo aquí ao aos as así á ben cando che co coa comigo con connosco contigo convosco coas cos cun cuns cunha cunhas da dalgunha dalgunhas dalgún dalgúns das de del dela delas deles desde deste do dos dun duns dunha dunhas e el ela elas eles en era eran esa esas ese eses esta estar estaba está este este

estes estiven estou eu é facer foi foron fun había hai iso isto la las lle lles lo los mais me meu meu min miña miñas moi nas neste nin no no no no nos nosa nosas noso nosos nós biarawati nunha biarawati biarawati o os ou ó ós para pero pode pois pola polas polo polos por que se senón ser seu seus sexa sido sobre súa súas tamén tan te ten teñen teño ter teu teus ti tido tido tiña tiven túa unha unhas uns vos vosa voso vosos vós

## Jerman (de)

Opsi stemming algoritmik: `||minimal light full`

Skema analisis default: `_de_default_`

- Pembendungan algoritmik: `light`
- Kamus stopword default:

aber alle allem allen aller alles als also am an ander andere anderem anderen anderer anderes andern anderr anderr anders auch auf aus bei bin bis bist da damit dann der den des dem die das daß derselbe derselben denselben demselben dieselbe dasselbe dazu dein deine deinem deinem deinem deinem deiner deines denn derer dessen dich dir du dies diese diesen dieser dieses doch dort durch ein eine einem einen eines einige einige einige einigen einiges einiges einiges einiges ihm es etwas euer eure euren eurer eures gegen gewesen hab habe haben hat hatte hatten hier hin hinter ich mich mir ihr ihre ihren ihrer ihres euch im in indem ins ist jede jedem jeden jeder jedes jene jenem jenem jenem jenen jener jenes jetzt kann keine keinem keinen keines können könnte machen man manche manche chem manchen mancher manches mein meine meinem meinen meiner meines mit muss musste nach nichts noch nun nur ob oder ohne sehr sein seine seinem seinen seiner seines selbst sich sie ihnen sind so solche solche solchen solcher soll sollte sondern sondern st über um und uns unse unsem unser unses unter viel vom von vor während perang waren warst was weg weil weil weiter welche welchen welcher welches wenn werde werden wie wieder will wir wird wird wird wo wollen wollte würde würden zum zur zwar zwischen

## Yunani (el)

Opsi stemming algoritmik: `full`

Skema analisis default: `_el_default_`

- Pembendungan algoritmik: `full`



bár be belül benne cikk cikkek cikkekét csak de e eddig egész egy egyes egyéb egyik egyre ekkor el elég ellen elő először előtt előtt első éppen ebben ehhez emilyen ennek erre ez ezek ezen ezzel ezért és fel felé hanem hiszen hogy hogyan igen így illetve ill. ill ilyen ilyenkor ison ismét itt jó jól jobban kell kellett keresztül keressünk ki kívül között közül legalább lehet legyen lenni lesz lett maga magát majd már más másik meg még mellett mert mely melyek mi mit miért miért milyen mikor minden mindent mindenki mindig mindig mint tha mivel most nagy nagyobb nagyon ne néha nekem neki nem néhány nélkül nincs olyan ott össze ő ōk őket pedig persze rá s saját sem semmi sok sokat sokkal szemben szerint szinte talán tehát teljes vább továbbá több úgy ugyanis új újra után utána utána vagyis valaki valamint való vagyok van vannak volt voltam volta voltunk vissza vele viszont volna

## Indonesia (id)

Opsi stemming algoritmik: | light full

Skema analisis default: id\_default\_

- Pembendungan algoritmik: full
- Kamus stopwords default:

ada adanya adalah adapun agak agaknya agar akan akankah akhirnya aku akulah amat amatlah anda andalah antar diantaranya antara diantara diantara apa apaan mengapa apabila apalagi apatah atau atukah atau bagaikan bagaikan sebagai sebagainya sebagaimana baglah bagi bahkan bahwa bahwasanya sebaliknya banyak sebanyak seberapa begini beginian beginikah beginilah begitukah begitupun begitupun belum sebelih umlah sebelum sebelumnya sebenarnya berapa berapalah berapapun betulkah sebetulnya biasa biasanya bila bilakah bisa bisakah sebisanya boleh bolehkah kah buat bukan bukankah bukannya cuma bebas dahulu dalam dan dapat dari dekat kata demi demikianlah dengan sedemikian depan di dia dialah dini diri sendiri terdiri dong dulu enggak enggak entah entahlah entahlah terhadap hal hanya harus hanyalah haruslah seharusnya seharusnya hendaknya sehingga ibarat ingin inginkah inginkan inikah inilah itukah itu jangan jangankan janganlah jika jikalau juga justru kala kalau kalaulah kalaupun kalian kamilah kamu kamulah kan kapan kapankah kapanpun dikarenakan dikarenakan karena karenanya ke kecil kemudian kenapa kepada kepadanya ketika khususnya kini kinilah kiranya sekiranya kita kitalah kok lagi lagian selagi lah selaku lagi selaku lagi selaku lalu melalui terlalu lama selama lebih lama selama lebih bermacam macam manakala makanya makin malah mampu mampukah mana manalagi masih masihkah semasih memang mereka merekalah meski semula mungkin mungkinglah nah namun nanti nantinya nyaris oleh olehnya seorang pada padanya

padahal paling cepat sepantasnya sepantasnyalah para pastilah pastilah per pernah pula pun merupakan rupanya serupa saat sesaat saja sajalah saling bersama sesama sesama sambil sangat sana sangatlah sayalah se sebab sebabnya tersebutlah sedang sedangkan sedikit sedikitnya sedikitnya sesegera segalanya sesegera sesegera sejenak sekali sekalian sepun sesaat sepun sesaat sepun sesaat sepun sesaat sepun sesaat sesama sepun sesaat sepun sesaat sekarang juga a selain selalu seluruh seluruhnya semakin sementara sempat semua semuanya sendiri seolah seolah seperti sering seringnya serta siapa siapakah siapapun disini disinilah sinilah sesuatu sesuatunya suatu sesudah sesudahnya sudah sudahkah sudahlah agar tadi tadinya tak tak tanpa tentang tentulah tentu tentu saja seterusnya tetapi setiap minimal tiap-tiap tidak tidakkah tidaklah toh waduh waduh walaupun wahai sewaktu walau wong yaitu yang

## Irlandia (ga)

Opsi stemming algoritmik: `full`

Filter elision diaktifkan

Skema analisis default: `_ga_default_`

- Opsi stemming algoritmik: `full`
- Kamus stopword default:

a ach ag agus an aon ar arna as b' ba beirt bhúr caoga ceathair ceathrar chomh chtó chuig chun cois céad cúig cúigear d' daichead dar de deich deichniúr den dhá do don dtí dá dár dár dó faoi faoin faoina faoinár ffiche gach gan go gur haon hocht i iad idir in ina ins inár is le leis lena lenár m' mar mo mé na nach naoi naonúr ná ní níor no ócha ocht ochtar os roimh sa seacht seachtar seachtó seasca seisear siad sibh sinn sn na sé sí tar thar thú triúr trí trína trínár tríocha tú um ár é éis í ó óna óna ónár

## Italia (itu)

Opsi stemming algoritmik: `| light full`

Filter elision diaktifkan

Skema analisis default: `_it_default_`

- Pembendungan algoritmik: `light`



- Kamus stopwords default:

ad al allo ai agli semua agl alla alle con col coi da dal dallo dai dagli dall dagl dalla dalle di del dello dei degli dell degl della delle di nel nello nei negli nell negl nella nelle su sul sullo sugli sull sugl sulla sulle per tra contro io tu lui lei noi voi loro mio mia miei mie tuo tua tuoi tue suo sua suoi sue nostro kami kami vostro vostro vostra vostri vostre mi ti ci vi lo la li le gli ne il un uno ma ed se perché anche dove che chi cui non più quale quanto quanti quanta quanta quello quelli quella quelle questo questi questi queste tutto tutti a c e i l o ho hai ha abbiamo avete hanno abbia abbiate abbiano avrò avrai avremo avrete avrete avrei avrebbe avrebbe avremmo avrebbero avevo avevamo avevamo avevate avevano ebbero avesti avemmo aveste ebbero avessi avesse avessimo avessero avendo avuto avuta avuti avute sono sei dan siamo siete sia siate siano sarò sarai sarà saremo saranno sarei saresti sarebbe saremmo sareste sarebbero ero eri era eravamo eravate erano fui fosti fu fummo foste furono fosi fosse fossimo fossero essendo faccio fai facciamo fanno faccia facciate facciano farai farai farà faremo farete faranno farei faresti farebbe faremmo fareste farebbero facevi facevi facevamo facevate facevano feci facesti fece facemmo faceste fecero facessi facessimo facessimo facessero facendo sto stai sta stiamo stanno stia stiano starò starai starà starà staremo starete staranno starei staresti starebbe staremmo stareste starebbero stavo stavi stava stavamo stavate stavano stavano stetti stesti stemmo stettero stessi stessimo stessimo stessero stando

## Jepang (ja)

Opsi stemming algoritmik: full

Penguraian algoritmik diaktifkan

Kamus tokenisasi opsional

Skema analisis default: `_ja_default_`

- Pembendungan algoritmik: full
- Kamus stopwords default:

表

## Korea (ko)

Pembatalan algoritmik tidak didukung

## Penguraian algoritmik diaktifkan

Skema analisis default: `_ko_default_`

- Kamus stopword default

## Latvia (lv)

Pembendungan algoritmik: `light`

Skema analisis default: `_lv_default_`

- Pembendungan algoritmik: `light`
- Kamus stopword default:

aiz ap ar apakš ārpus augšpus bez caur dēgar iekš iz kopš labad lejpus līdz no otrpus pa par pār  
pēc pie pirms pret priekš starp šaipus uz viņpus virs virspus zem apakšpus un bet jo ja ka lai tomēr  
tikko turpretī arī kaut gan tādētā ne tikvien vien kā ir te vai kamēr ar diezin droši diemžēl nebūt ik  
it taču nu pat tiklab iekšpus nedz tik nevis turpretim jeb iekam iekām iekāms kolīdz līdzko tiklīdz  
jebīdz šu tālab tāpēc nekā itin jā jau jel nē nezina tad tikai vis tak iekams vien būt biju biji bija bijām  
bijāt esmu esi esam esat būšu būsi būsīm būsiet tikt tiku tiki tika tikām tikāt tieku tiec tiek tiekam  
tiekat tikšu tiks tiksīm tiksiet tapt tapi tapāt topat tapsi tapsi tapsiet tapsīm tapsiet kuvu kuvām  
kuvām kuvāt kūstu kūsti kuvām kļūstām kļūšu kļūsi kļūsim kļūsiet varēju varējām varēšu varēsīm  
var varēji varējāt varēsiet varēsiet varēja varēs

## Beberapa (mul)

Pembendungan algoritmik: tidak didukung

Skema analisis default: `_mul_default_`

- Kamus stopword default

## Norwegia (no)

Opsi stemming algoritmik: `|| minimal light full`

Skema analisis default: `_no_default_`

- Pembendungan algoritmik: `light`





- Kamus stopwords default:

acea aceasta această aceea acei aceia acel acele acele acelea acest acesta aceste acestea acești aceștia acolo acum ai aia aia aibă aici al ăla ale alea ălea altceva altcineva am ar aș așadar asemenea asta ăsta astăzi astea ăstea ăștia asupra ați au avea avem aveți azi bine bucur bună ca căci când care cărei căror cărui cât câte câți către câtva ce cel ceva chiar când cine cineva cît cîte cîți cîtva contra cu cum curva cumva ând curînd da dă dă dar datorită de deci deja deoarece deși din dinaintea dintr dintre drept după ea ei el ele eram este ești eu wajah fi fie fiecare fii fim fiți iar ieri îi îl îmi împotriva înainte înainte încât încât încotro întrucât întrucât îți la lângă le li lîngă lui mă mâine mea mei mele mereu meu mi mine mult multă mulți ne nici nimeni niște noastre noi noștri noștri nostru nu ori oricând oricare oricât oricât oricînd oricine oricît oricum oriunde până pe pentru peste pînă poate pot prea prima primul prin printr sa să săi sale sau său se și sînt sîntem sînteți spre sub suntem suntem sunteți ta tăi tale tău te ți tine toată tot toți totuși tu un una unde undeva unei uneori unor vă vi voastre voi voi voștri vostru vouă vreo

## Rusia (ru)

Opsi stemming algoritmik: | light full

Skema analisis default: `_ru_default_`

- Pembendungan algoritmik: light
- Kamus stopwords default:

и в во не мо он на я с со каца то все она таК его но да уе вы иа бы бы оо тоо только ее мне было воо оо отеня ее нее ему тему ь когда даме ну вдруг ли если уе или ни быть бые бынего до вас нибудь оть уваме скамамуь танибья ниниего ебо еба оее они тые где есть надо немы тебя тебя их сема была самобебудто будто селовецо рего раго тобен огда ко этоо говорие того тому этого какососовсенининимесь этомомин один томи монобы нее кае кайкас быис ли куда сасканамамать всех никогда сегодння моно три наконеца ов другохоть мосле нае большие тототи нас нас тро всего них какая много рамного Раскамала три эту моя вронооо своирроиррроен иногда луме суть тоннелья такобобое более всегда конено всмеду

## Spanyol (es)

Opsi stemming algoritmik: | light full

Skema analisis default: `_es_default_`

- Pembendungan algoritmik: `light`
- Kamus stopwords default:

de la que el en y a los del se las por un para con no una su al lo como más pero sus le ya o este sí porque esta entre cuando muy sin sobre también me hasta hay donde quien desde todo nos durante todos uno les ni contra otros ese eso ante ellos e esto mí antes algunos qué unos yo otras otra él tanto esa estos mucho quienes nada muchos cual poco ella estar estas algunas algo nosotros mi mis tú te ti tu tus ellas nosotras vosotras vosotros vosotras os mío mía míos mías tuyo tuyo tuyos tuyas suyo suya suyos nuestro nuestra nuestra nuestra vuestro vuestra vuestro vuestras esas estoy estás estás estáis estáis estáis estaréis estarás estarás estaréis estaréis estaréis estaría estarías estarías estaríais estarían estabas estábamos estabas estábamos estaréis estaréis estarías estarías estás estábamos estás estuviste estuvimos estuvisteis estuvieron estuviera estuvieras estuvierais estuvieran estuviese estuviesen estudiésemos estudiéis estudiéis estado estada estadas estad día has ha hemos habéis han haya hayas hayamos hayáis hayan habré habrás habrá habremos habréis habrán habría habrías habríamos habrías habrían había habías habíamos habíais habían hubiste hubo hubimos hubisteis hubieron hubiera hubieras hubiéramos hubierais hubierais hubierais hubiesemos hubieseis hubiesen habiendo habido habida habida habidos habidos habidos habidos soy eres es somos sois son sea seas seamos seáis sean seré serás será seremos seréis sería serías serías seríamos seríais seríais serían era eras erais erais eran fui fuiste fuimos fuisteis fueron fueras fueras fueras fuerais fueran fueses fuésemos fueseis fuesen siendo sido tengo tienes tiene tenemos tenéis tienen tenga tengas tengamos tengáis tengan tendré tendrás tendrá tendremos tendréis tendrán tendría tendrías tendríamos tendríais tendría tenías tenías teníamos teníais tendría tuve tuviste tuvimos tuvisteis tuvieron tuviera tuviera tuviera tuviéramos tuviéramos tuvierais tuvieran tuviese tuviésemos tuviéis tuviéis teniendo tenido tenida tenidos tenidas teneo

## Swedia (sv)

Opsi stemming algoritmik: `|light full`

Skema analisis default: `_sv_default_`

- Pembendungan algoritmik: `light`
- Kamus stopwords default:

och det att i en jag hon som han di den med var sig untuk så to är men ett om hade de av icke mig du henne då sin nu har inte hans honom skulle hennes där min man ej vid kunde något dari ut när

efter upp vi dem vara vad dig kan sina här ha mot alla under någon atau allt mycket sedan ju denna själv detta åt utan varit hur ingen mitt ni bli blev oss din dessa några deras blir mina samma vilken er sådan vår blivit dess inom mellan sådant varför varje vilka ditt vem vilket sitta sådana vart dina vars vårt våra ert era vilkas

## Thailand (th)

Pembatalan algoritmik tidak didukung

Kamus stemming tidak didukung

Skema analisis default: `_th_default_`

- Kamus stopwords default:

ไว้ ไม่ ไป ได้ ให้ ใน โดย แห่ง แล้ว และ แรก แบบ แต่ เอง เห็น เลย เริ่ม เรา เมื่อ เพื่อ เพราะ เป็นการ เป็น เปิดเผย เปิด เนื่องจาก เดียวกัน เดียว เช่น เฉพาะ เคย เข้า เขา อีก อาจ อะไร ออก อย่าง อยู่ อยาก หาก หลาย หลังจาก หลัง หรือ หนึ่ง ส่วน ส่ง สุด สำหรับ ว่า วัน ลง รวม ราย รับ ระหว่าง รวม ยังมี มาก มา พร้อม พบ ผ่าน ผล บาง นำ นี้ นำ นั้น นัก นอกจาก ทุก ที่สุด ที่ ทำให้ ทำ ทาง ทั้งนี้ ทั้ง ถ้า ถูก ถึง ต้อง ต่างๆ ต่าง ต่อ ตาม ตั้งแต่ ตั้ง ด้าน ด้วย ตั้ง ซึ่ง ช่วง จึง จาก จัด จะ คือ ความ ครั้ง คง ขึ้น ของ ขอ ขณะ ก่อน ก็ การ กับ กัน กว่า กล่าว

## Turki (tr)

Pembendungan algoritmik: `full`

Skema analisis default: `_tr_default_`

- Pembendungan algoritmik: `full`
- Kamus stopwords default

# Mengunggah dan Mengindeks Data di Amazon CloudSearch

Untuk membuat data Anda dapat dicari, Anda perlu memformatnya dalam JSON atau XML seperti yang dijelaskan [Mempersiapkan Data Anda](#) dan mengunggahnya ke domain pencarian Anda untuk pengindeksan. Dalam kebanyakan kasus, Amazon CloudSearch secara otomatis mengindeks data Anda dan perubahannya terlihat di hasil penelusuran hanya dalam beberapa menit. Namun, perubahan tertentu pada konfigurasi domain Anda menempatkan domain dalam NEEDS INDEXING status. Agar perubahan tersebut diterapkan, Anda harus secara eksplisit menjalankan pengindeksan untuk membangun kembali indeks Anda. Saat ini, Anda juga perlu menjalankan pengindeksan secara berkala sehingga sugerter Anda mencerminkan data terbaru dalam indeks Anda. Bagian berikut menjelaskan cara mengunggah data ke domain Anda dan menjalankan pengindeksan saat diperlukan.

## Important

Membangun kembali indeks Anda setelah pengunggahan data tidak diperlukan dan dapat menyebabkan domain Anda dikenakan biaya tambahan. Anda hanya perlu membangun kembali indeks Anda setelah perubahan konfigurasi tertentu atau setelah Anda menghapus dokumen dan ingin mereka dihapus secara permanen dari layanan.

## Topik

- [Mengunggah Data ke Domain Amazon CloudSearch](#)
- [Mengindeks Data Dokumen dengan Amazon CloudSearch](#)

## Mengunggah Data ke Domain Amazon CloudSearch

### Important

Sebelum mengunggah data ke CloudSearch domain Amazon, ikuti panduan berikut:

- Kelompokkan dokumen ke dalam batch sebelum Anda mengunggahnya. Terus mengunggah batch yang hanya terdiri dari satu dokumen memiliki dampak negatif yang besar pada kecepatan Amazon CloudSearch dapat memproses pembaruan Anda. Sebagai gantinya, buat batch yang sedekat mungkin dengan batas dan unggah lebih jarang. Untuk informasi selengkapnya tentang ukuran batch maksimum dan frekuensi unggah, lihat [Batas](#).



- Dokumen domain dan titik akhir pencarian tetap sama untuk masa pakai domain. Anda harus men-cache titik akhir daripada mengambilnya sebelum setiap permintaan unggahan atau pencarian. Menanyakan layanan CloudSearch konfigurasi Amazon dengan menelepon `aws cloudsearch describe-domains` atau `DescribeDomains` sebelum setiap permintaan kemungkinan akan mengakibatkan permintaan Anda dibatasi.

Anda membuat kumpulan dokumen untuk menggambarkan data yang ingin Anda unggah ke CloudSearch domain Amazon. Kumpulan dokumen adalah kumpulan operasi tambah dan hapus yang mewakili dokumen yang ingin Anda tambahkan, perbarui, atau hapus dari domain Anda. Batch dapat dijelaskan dalam JSON atau XHTML. Saat Anda mengunggah batch dokumen ke domain, data diindeks secara otomatis sesuai dengan opsi pengindeksan domain.

Saat data Anda berubah, Anda mengunggah batch untuk menambah, mengubah, atau menghapus dokumen dari indeks Anda. Amazon CloudSearch menerapkan pembaruan terus menerus. Anda hanya perlu secara eksplisit mengindeks ulang data Anda ketika Anda membuat perubahan konfigurasi yang menempatkan domain Anda dalam `NEEDS INDEXING` status atau perlu memperbarui sugester.

Untuk mengunggah data ke domain Anda, itu harus diformat sebagai batch JSON atau XHTML yang valid. Bidang yang ditentukan dalam setiap dokumen harus sesuai dengan bidang indeks yang dikonfigurasi untuk domain. Namun, dokumen tidak harus berisi setiap bidang indeks yang dikonfigurasi. Untuk informasi tentang membuat kumpulan dokumen, lihat [Mempersiapkan Data Anda](#). Untuk informasi tentang mengonfigurasi bidang indeks untuk domain, lihat [configure indexing options](#).

Anda ditagih untuk jumlah total kumpulan dokumen yang diunggah ke domain pencarian Anda, termasuk batch yang berisi operasi penghapusan. Untuk informasi selengkapnya tentang CloudSearch harga Amazon, lihat [aws.amazon.com/cloudsearch/pricing/](https://aws.amazon.com/cloudsearch/pricing/).

Anda dapat mengirimkan kumpulan dokumen ke domain menggunakan [Amazon CloudSearch console](#), AWS CLI, atau dengan [posting it directly](#) ke endpoint layanan dokumen domain.

Untuk informasi selengkapnya tentang API layanan dokumen, lihat [Document Service API](#).

## Topik

- [Mengirimkan Permintaan Unggahan Dokumen ke Domain Amazon CloudSearch](#)
- [Unggahan Massal di Amazon CloudSearch](#)
- [Mengunggah Data Menggunakan Konsol Amazon CloudSearch](#)

- [Mengunggah Data Menggunakan AWS CLI](#)
- [Memposting Dokumen ke Endpoint Layanan Dokumen CloudSearch Domain Amazon melalui HTTP](#)

## Mengirimkan Permintaan Unggahan Dokumen ke Domain Amazon CloudSearch

### Important

Sebelum mengunggah data ke CloudSearch domain Amazon, ikuti panduan berikut:

- Kelompokkan dokumen ke dalam batch sebelum Anda mengunggahnya. Terus mengunggah batch yang hanya terdiri dari satu dokumen memiliki dampak negatif yang besar pada kecepatan Amazon CloudSearch dapat memproses pembaruan Anda. Sebagai gantinya, buat batch yang sedekat mungkin dengan batas dan unggah lebih jarang. Untuk informasi selengkapnya tentang ukuran batch maksimum dan frekuensi unggah, lihat [Batas](#).
- Dokumen domain dan titik akhir pencarian tetap sama untuk masa pakai domain. Anda harus men-cache titik akhir daripada mengambilnya sebelum setiap permintaan unggahan atau pencarian. Menanyakan layanan CloudSearch konfigurasi Amazon dengan menelepon `aws cloudsearch describe-domains` atau `DescribeDomains` sebelum setiap permintaan kemungkinan akan mengakibatkan permintaan Anda dibatasi.

Sebaiknya gunakan salah satu AWS SDK atau AWS CLI untuk mengirimkan permintaan unggahan dokumen. SDK dan AWS CLI menangani penandatanganan permintaan untuk Anda dan menyediakan cara mudah untuk melakukan semua CloudSearch tindakan Amazon. Anda juga dapat menggunakan CloudSearch konsol Amazon untuk mengunggah batch individual dan mengimpor data dari DynamoDB atau S3.

Misalnya, permintaan berikut mengunggah batch menggunakan file. AWS CLI

```
aws cloudsearchdomain --endpoint-url http://doc-movies-y6gelr4lv3jeu4rvoelunxsl2e.us-east-1.cloudsearch.amazonaws.com upload-documents --content-type application/json --documents movie-data-2013.json
```

Untuk tujuan pengembangan dan pengujian, Anda dapat mengizinkan akses anonim ke layanan dokumen domain Anda dan mengirimkan permintaan HTTP POST yang tidak ditandatangani

langsung ke layanan dokumen domain Anda. Di lingkungan produksi, batasi akses ke domain Anda ke peran, grup, atau pengguna IAM tertentu dan kirimkan permintaan yang ditandatangani. Untuk informasi tentang mengontrol akses untuk Amazon CloudSearch, lihat [configure access policies](#). Untuk informasi selengkapnya tentang penandatanganan permintaan, lihat [Menandatangani Permintaan AWS API](#).

Misalnya, permintaan POST berikut mengunggah sekumpulan dokumen yang diformat dalam JSON ke titik akhir domain `doc-movies-123456789012.us-east-1.cloudsearch.amazonaws.com`.

```
curl -X POST --upload-file data1.json doc-movies-123456789012.us-east-1.cloudsearch.amazonaws.com/2013-01-01/documents/batch --header "Content-Type: application/json"
```

## Unggahan Massal di Amazon CloudSearch

Batch dokumen dibatasi hingga satu batch setiap 10 detik dan 5 MB per batch. Untuk mempelajari lebih lanjut, lihat [Batas](#). Namun, Anda dapat mengunggah batch secara paralel untuk mengurangi jumlah waktu yang diperlukan untuk mengunggah semua data Anda.

Untuk melakukan upload massal:

- Setel jenis instans yang Anda inginkan ke jenis instans yang lebih besar daripada `defaultsearch.small`. Jumlah utas unggahan yang dapat Anda gunakan tergantung pada jenis contoh pencarian yang digunakan domain Anda dan sifat data dan opsi pengindeksan Anda. Jenis instans yang lebih besar memiliki kapasitas unggah yang lebih tinggi. Mencoba mengunggah batch secara paralel dengan `search.small` instance biasanya menghasilkan tingkat kesalahan 504 atau 507 yang tinggi. Untuk informasi selengkapnya tentang menyetel jenis instans yang diinginkan, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).
- Mulai unggah data setelah perubahan konfigurasi Anda aktif. Jika Anda menemukan tingkat kesalahan 5xx yang tinggi, Anda perlu mengurangi tingkat unggah atau beralih ke jenis instans yang lebih besar. Jika Anda sudah menggunakan jenis instans terbesar, Anda dapat meningkatkan jumlah partisi yang diinginkan untuk lebih meningkatkan kapasitas unggah.

### Important

Jika Anda mengirimkan sejumlah besar pembaruan saat domain Anda berada dalam status PEMROSESAN, itu dapat meningkatkan jumlah waktu yang diperlukan untuk pembaruan untuk diterapkan ke indeks pencarian Anda. Untuk menghindari kelambatan pembaruan

ini, tunggu hingga domain Anda dalam status AKTIF sebelum memulai unggahan massal Anda.

- Setelah selesai dengan upload massal, Anda dapat mengubah jenis instans yang diinginkan kembali ke jenis instans yang lebih kecil. Jika indeks Anda cocok dengan tipe yang lebih kecil, Amazon CloudSearch akan secara otomatis menskalakan domain Anda kembali. Amazon tidak CloudSearch akan menskalakan ke jenis instans yang lebih kecil dari jenis instans yang diinginkan yang dikonfigurasi untuk domain Anda.

Untuk kumpulan data kurang dari 1 GB data atau kurang dari satu juta dokumen 1 KB, contoh pencarian kecil harus cukup. Untuk mengunggah kumpulan data antara 1 GB dan 8 GB, sebaiknya setel jenis instans yang diinginkan `search.large` sebelum Anda mulai mengunggah. Untuk kumpulan data antara 8 GB dan 16 GB, mulailah dengan `file.search.xlarge` Untuk kumpulan data antara 16 GB dan 32 GB, mulailah dengan `file.search.2xlarge` Jika Anda memiliki lebih dari 32 GB untuk diunggah, pilih jenis `search.2xlarge` instans dan tingkatkan jumlah partisi yang diinginkan untuk mengakomodasi kumpulan data Anda. Setiap partisi dapat berisi hingga 32 GB data. Kirim [Permintaan Batas Peningkatan Layanan](#) jika Anda membutuhkan kapasitas unggah lebih besar atau memiliki lebih dari 500 GB untuk diindeks.

## Mengunggah Data Menggunakan Konsol Amazon CloudSearch

Di CloudSearch konsol Amazon, Anda dapat mengunggah data dari sistem file lokal atau Amazon S3 ke domain Anda dari dasbor domain. Konsol dapat secara otomatis mengonversi jenis file berikut ke kumpulan dokumen selama proses pengunggahan:

- Batch dokumen diformat dalam JSON atau XML.`.json`, `.xml`.
- Nilai Terpisah Koma (`.csv`)
- Dokumen Teks (`.txt`)

Anda juga dapat mengonversi dan mengunggah item dari tabel DynamoDB. Untuk informasi selengkapnya, lihat [Uploading DynamoDB Data](#).

### Note

Untuk mengunggah data dari Amazon S3 atau DynamoDB, Anda harus memiliki izin untuk mengakses layanan dan sumber daya yang ingin Anda unggah. Untuk informasi

selengkapnya, lihat [Menggunakan Kebijakan Bucket dan Kebijakan Pengguna dan Menggunakan IAM untuk Mengontrol Akses ke Sumber Daya DynamoDB](#).

File CSV diurai row-by-row dan dokumen terpisah dibuat untuk setiap baris. Semua jenis file lainnya diperlakukan sebagai satu dokumen. Untuk informasi selengkapnya tentang membuat batch dokumen secara otomatis, lihat [Mempersiapkan Data Anda](#).

Untuk mengirim data ke domain untuk pengindeksan

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih Domain.
3. Pilih nama domain Anda untuk membuka konfigurasi domain.
4. Pilih Tindakan, Unggah dokumen.
5. Pilih lokasi data yang ingin Anda unggah ke domain Anda:
  - Mesin lokal
  - Amazon S3
  - Amazon DynamoDB
  - Data sampel

Jika Anda mengunggah data yang tidak diformat sebagai kumpulan dokumen, maka secara otomatis akan dikonversi selama proses pengunggahan.

#### Note

Jika batch tidak valid, Amazon CloudSearch mengonversi konten menjadi kumpulan valid yang berisi satu bidang konten dan bidang metadata generik. Karena ini biasanya bukan bidang yang dikonfigurasi untuk domain, Anda akan mendapatkan kesalahan yang menyatakan bahwa bidang tidak ada.

6. Unggah data Anda.
  - a. Jika Anda mengunggah file lokal, pilih Pilih file untuk menemukan file yang akan diunggah.
  - b. Jika Anda mengunggah objek dari Amazon S3, berikan URI bucket untuk diunggah.

- c. Jika Anda mengunggah item dari DynamoDB, pilih tabel untuk diunggah. Untuk membatasi unit kapasitas baca yang dapat dikonsumsi saat membaca dari tabel, masukkan persentase maksimum unit kapasitas baca. Untuk mulai membaca dari item tertentu, tentukan kunci hash mulai.
  - d. Jika Anda mengunggah data sampel yang telah ditentukan sebelumnya, pilih kumpulan data yang akan digunakan.
7. Pilih Lanjutkan.
  8. Tinjau dokumen yang akan diunggah dan pilih Unggah dokumen.
  9. Dalam Ringkasan Unggah, jika kumpulan dokumen telah dibuat secara otomatis dari data Anda, Anda dapat memilih Unduh kumpulan dokumen yang dihasilkan untuk mendapatkannya. Pilih Tutup untuk kembali ke dasbor domain.

## Mengunggah Data Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch upload-documents` perintah untuk mengirim batch dokumen ke domain pencarian Anda. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line Interface Panduan Pengguna](#).

Untuk mengirim batch dokumen ke domain untuk pengindeksan

- Jalankan `aws cloudsearchdomain upload-documents` perintah untuk mengunggah batch Anda ke domain Anda:

```
aws cloudsearchdomain upload-documents --endpoint-url http://doc-movies-
y6gelr4lv3jeu4rvoelunxsl2e.us-east-1.cloudsearch.amazonaws.com --content-type
application/json --documents document-batch.json
{
  "status": "success",
  "adds": 5000,
  "deletes": 0
}
```

## Memposting Dokumen ke Endpoint Layanan Dokumen CloudSearch Domain Amazon melalui HTTP

Anda menggunakan [dokumen/batch](#) sumber daya untuk memposting batch dokumen ke domain Anda untuk menambah, memperbarui, atau menghapus dokumen. Sebagai contoh:

```
curl -X POST --upload-file movie-data-2013.json doc-movies-123456789012.us-east-1.cloudsearch.amazonaws.com/2013-01-01/documents/batch --header "Content-Type:application/json"
```

## Mengindeks Data Dokumen dengan Amazon CloudSearch

Saat Anda mengirim pembaruan dokumen ke domain Anda, Amazon CloudSearch secara otomatis memperbarui indeks pencarian domain dengan data baru. Anda tidak perlu melakukan apa pun agar pembaruan dapat diindeks. Namun, jika Anda mengubah konfigurasi bidang indeks domain atau opsi teks, Anda harus secara eksplisit membangun kembali indeks penelusuran agar perubahan tersebut terlihat di hasil penelusuran. Karena membangun kembali indeks dapat memakan banyak waktu jika Anda memiliki banyak data, Anda harus menyelesaikan semua perubahan konfigurasi Anda sebelum mengindeks ulang dokumen Anda.

### Important

Jika Anda mengubah jenis bidang dan memiliki dokumen dalam indeks Anda yang berisi data yang tidak kompatibel dengan jenis bidang baru, semua bidang yang sedang diproses dimasukkan ke dalam `FailedToValidate` status saat Anda menjalankan pengindeksan dan operasi pengindeksan gagal. Mengembalikan perubahan konfigurasi yang tidak kompatibel akan memungkinkan Anda untuk berhasil membangun kembali indeks Anda. Jika perubahan diperlukan, Anda harus memperbarui atau menghapus dokumen yang tidak kompatibel dari indeks Anda untuk menggunakan konfigurasi baru.

Saat Anda membuat perubahan yang memerlukan pengindeksan ulang, status domain akan berubah menjadi `Needs Indexing Sementara` indeks sedang dibangun kembali, status domain adalah `Processing`. Anda dapat terus mengirimkan permintaan penelusuran saat pengindeksan sedang dalam proses, tetapi perubahan konfigurasi tidak akan terlihat di hasil penelusuran hingga pengindeksan selesai dan status domain berubah menjadi `Active`. Anda juga dapat terus mengunggah kumpulan dokumen ke domain Anda. Namun, jika Anda mengirimkan sejumlah besar

pembaruan saat domain Anda dalam Processing keadaan, itu dapat meningkatkan jumlah waktu yang diperlukan untuk pembaruan untuk diterapkan ke indeks pencarian Anda. Jika ini menjadi masalah, perlambat tingkat pembaruan Anda hingga domain kembali ke Active status.

#### Note

Tergantung pada volume data, membangun indeks penuh dapat mengambil sejumlah besar daya komputasi. Amazon CloudSearch secara otomatis mengelola sumber daya yang dibutuhkan untuk membangun indeks secara tepat waktu. Sebagian besar pembaruan data dan perubahan konfigurasi domain sederhana dibuat dan digunakan dalam hitungan menit. Mengindeks volume data yang besar dan menerapkan perubahan konfigurasi yang memerlukan pembangunan kembali indeks penuh akan memakan waktu lebih lama untuk diselesaikan.

Anda dapat memulai pengindeksan dari [Amazon CloudSearch console](#), menggunakan `aws cloudsearch index-documents` perintah, atau melalui AWS SDK.

#### Topik

- [Mengindeks Dokumen Menggunakan Konsol Amazon CloudSearch](#)
- [Pengindeksan Dokumen Menggunakan Amazon CloudSearch AWS CLI](#)
- [Mengindeks Dokumen dengan AWS SDK](#)

## Mengindeks Dokumen Menggunakan Konsol Amazon CloudSearch

Saat Anda membuat perubahan yang mengharuskan indeks domain Anda dibangun kembali, status yang ditampilkan di dasbor domain berubah menjadi NEEDS INDEXING. Konsol juga menampilkan pesan di bagian atas halaman konfigurasi yang meminta Anda untuk menjalankan pengindeksan ketika Anda selesai membuat perubahan.

Untuk menjalankan pengindeksan

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Dari panel navigasi kiri, pilih Domain.
3. Pilih nama domain yang membutuhkan pengindeksan.
4. Di dasbor domain, pilih Tindakan, Jalankan pengindeksan.



## Pengindeksan Dokumen Menggunakan Amazon CloudSearch AWS CLI

Anda menggunakan `aws cloudsearch index-documents` perintah untuk membangun kembali indeks pencarian domain Anda. Untuk informasi tentang menginstal dan menyiapkan AWS CLI, lihat [AWS Command Line InterfacePanduan Pengguna](#).

Untuk secara eksplisit mengindeks domain Anda

- Jalankan perintah `aws cloudsearch index-documents`. Contoh berikut membangun kembali indeks untuk domain yang disebut film.

### Example

```
aws cloudsearch index-documents --domain-name movies
```

## Mengindeks Dokumen dengan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [IndexDocuments](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

# Mencari Data Anda dengan Amazon CloudSearch

Anda menentukan istilah atau nilai yang ingin Anda cari dengan `q` parameter. Cara Anda menentukan kriteria pencarian tergantung pada parser kueri mana yang Anda gunakan. Amazon CloudSearch mendukung empat parser kueri:

- `simple`—cari semua `text` dan `text-array` bidang untuk string yang ditentukan. Parser kueri sederhana memungkinkan Anda mencari frasa, istilah individual, dan awalan. Anda dapat menetapkan persyaratan sebagai wajib atau opsional, atau mengecualikan kecocokan yang berisi istilah tertentu. Untuk mencari bidang tertentu, Anda dapat menentukan bidang yang ingin Anda cari dengan `q.options` parameter. Parser `simple query` digunakan secara default jika `q.parser` parameter tidak ditentukan.
- `structured`—cari bidang tertentu, buat kueri majemuk menggunakan operator Boolean, dan gunakan fitur-fitur canggih seperti penguat istilah dan pencarian kedekatan.
- `lucene`—tentukan kriteria pencarian menggunakan sintaks parser kueri Apache Lucene. Jika saat ini Anda menggunakan sintaks Lucene, menggunakan parser `lucene kueri` memungkinkan Anda memigrasikan layanan penelusuran ke CloudSearch domain Amazon tanpa harus menulis ulang kueri penelusuran Anda sepenuhnya dalam sintaks penelusuran terstruktur Amazon. CloudSearch
- `dismax`—tentukan kriteria pencarian menggunakan subset sederhana dari sintaks parser kueri Apache Lucene yang ditentukan oleh parser kueri. `DisMax` Jika saat ini Anda menggunakan `DisMax` sintaks, menggunakan parser `dismax kueri` memungkinkan Anda memigrasikan layanan penelusuran ke CloudSearch domain Amazon tanpa harus menulis ulang kueri penelusuran Anda sepenuhnya dalam sintaks penelusuran terstruktur Amazon CloudSearch .

Anda dapat menggunakan parameter penelusuran tambahan untuk [mengontrol bagaimana hasil penelusuran dikembalikan](#) dan [menyertakan informasi tambahan](#) seperti aspek, sorotan, dan saran dengan hasil penelusuran Anda.

Untuk informasi tentang semua parameter CloudSearch penelusuran Amazon, lihat [Search API](#).

## Topik

- [Mengirimkan Permintaan Pencarian ke Domain Amazon CloudSearch](#)
- [Membangun Kueri Senyawa di Amazon CloudSearch](#)
- [Mencari Teks di Amazon CloudSearch](#)
- [Mencari Angka di Amazon CloudSearch](#)

- [Mencari Tanggal dan Waktu di Amazon CloudSearch](#)
- [Mencari Rentang Nilai di Amazon CloudSearch](#)
- [Pencarian dan Peringkat Hasil berdasarkan Lokasi Geografis di Amazon CloudSearch](#)
- [Mencari Data DynamoDB dengan Amazon CloudSearch](#)
- [Memfilter Dokumen Pencocokan di Amazon CloudSearch](#)
- [Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch](#)

## Mengirimkan Permintaan Pencarian ke Domain Amazon CloudSearch

Sebaiknya gunakan salah satu AWS SDK atau AWS CLI untuk mengirimkan permintaan pencarian. SDK dan AWS CLI menangani penandatanganan permintaan untuk Anda dan menyediakan cara mudah untuk melakukan semua CloudSearch tindakan Amazon. Anda juga dapat menggunakan Penguji Penelusuran di CloudSearch konsol Amazon untuk mencari data, menelusuri hasilnya, dan melihat URL permintaan yang dihasilkan serta respons JSON dan XHTML. Untuk informasi selengkapnya, lihat [Mencari dengan Search Tester](#).

### Important

- Titik akhir penelusuran tidak berubah: Dokumen domain dan titik akhir penelusuran tetap sama selama masa pakai domain. Anda harus men-cache titik akhir daripada mengambilnya sebelum setiap permintaan unggahan atau pencarian. Menanyakan layanan CloudSearch konfigurasi Amazon dengan menelepon `aws cloudsearch describe-domains` atau `DescribeDomains` sebelum setiap permintaan kemungkinan akan mengakibatkan permintaan Anda dibatasi.
- Alamat IP memang berubah: Alamat IP domain Anda dapat berubah seiring waktu, jadi penting untuk men-cache titik akhir seperti yang ditunjukkan di konsol dan dikembalikan oleh `aws cloudsearch describe-domains` perintah daripada alamat IP. Anda juga harus menyelesaikan kembali DNS endpoint ke alamat IP secara teratur. Untuk informasi selengkapnya, lihat [Mengatur TTL JVM untuk Pencarian Nama DNS](#).

Misalnya, permintaan berikut mengirimkan pencarian teks sederhana untuk `wolverine` menggunakan AWS CLI dan mengembalikan hanya ID dari dokumen yang cocok.

```
aws cloudsearchdomain --endpoint-url http://search-movies-
y6gelr4lv3jeu4rvoelunxsl2e.us-east-1.cloudsearch.amazonaws.com search --search-query
wolverine --return _no_fields
{
  "status": {
    "rid": "/rnE+e4oCAqfEEs=",
    "time-ms": 6
  },
  "hits": {
    "found": 3,
    "hit": [
      {
        "id": "tt1430132"
      },
      {
        "id": "tt0458525"
      },
      {
        "id": "tt1877832"
      }
    ],
    "start": 0
  }
}
```

Secara default, Amazon CloudSearch mengembalikan respons di JSON. Anda bisa mendapatkan hasil yang diformat dalam XML dengan menentukan parameter. `format` Menyetel format respons hanya memengaruhi respons terhadap permintaan yang berhasil. Format respons kesalahan tergantung pada asal kesalahan. Kesalahan yang dikembalikan oleh layanan pencarian selalu dikembalikan di JSON. Kesalahan 5xx karena batas waktu server dan masalah perutean permintaan lainnya dikembalikan dalam XHTML.

#### Note

AWS SDK mengembalikan bidang sebagai array. Bidang nilai tunggal dikembalikan sebagai array dengan satu elemen, seperti:

```
"fields": {
  "plot": ["Katniss Everdeen reluctantly becomes the symbol of a mass rebellion
against the autocratic Capitol."]
```

```
}
```

Untuk tujuan pengembangan dan pengujian, Anda dapat mengizinkan akses anonim ke layanan pencarian domain Anda dan mengirimkan permintaan HTTP GET atau POST yang tidak ditandatangani langsung ke titik akhir pencarian domain Anda. Di lingkungan produksi, batasi akses ke domain Anda ke peran, grup, atau pengguna IAM tertentu dan kirimkan permintaan yang ditandatangani menggunakan AWS SDK atau AWS CLI. Untuk informasi tentang mengontrol akses untuk Amazon CloudSearch, lihat [configure access policies](#). Untuk informasi selengkapnya tentang penandatanganan permintaan, lihat [Menandatangani Permintaan AWS API](#).

Anda dapat menggunakan metode apa pun yang ingin Anda kirim permintaan HTTP langsung ke titik akhir pencarian domain Anda—Anda dapat memasukkan URL permintaan secara langsung di browser Web, menggunakan cURL untuk mengirimkan permintaan, atau membuat panggilan HTTP menggunakan pustaka HTTP favorit Anda. Untuk menentukan kriteria penelusuran, Anda menentukan string kueri yang menentukan batasan untuk pencarian Anda dan apa yang ingin Anda dapatkan kembali dalam respons. String kueri harus dikodekan URL. Ukuran maksimum permintaan pencarian yang dikirimkan melalui GET adalah 8190 byte, termasuk metode HTTP, URI, dan versi protokol. Anda dapat mengirimkan permintaan yang lebih besar menggunakan HTTP POST; Namun, perlu diingat bahwa permintaan yang besar dan kompleks membutuhkan waktu lebih lama untuk diproses dan lebih mungkin untuk habis waktu. Untuk informasi selengkapnya, lihat [Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch](#).

Misalnya, permintaan berikut mengirimkan kueri terstruktur ke `search-movies-rr2f34ofg56xneuemujamut52i.us-east-1.cloudsearch.amazonaws.com` domain dan mendapatkan isi `title` bidang.

```
http://search-movies-rr2f34ofg56xneuemujamut52i.us-east-1.cloudsearch.
amazonaws.com/2013-01-01/search?q=(and+(term+field%3Dtitle+'star')
(term+field%3Dyear+1977))&q.parser=structured&return=title
```

### Important

Karakter khusus dalam string kueri harus dikodekan URL. Misalnya, Anda harus menyandikan `=` operator dalam kueri terstruktur sebagai `%3D:(term+field%3Dtitle+'star')`. Jika Anda tidak menyandikan karakter khusus saat mengirimkan permintaan pencarian, Anda akan mendapatkan `InvalidQueryString` kesalahan.

## Mencari dengan Search Tester

Penguji penelusuran di CloudSearch konsol Amazon memungkinkan Anda mengirimkan contoh permintaan penelusuran menggunakan salah satu parser kueri yang didukung: sederhana, terstruktur, lucene, atau dismax. Secara default, permintaan diproses dengan parser kueri sederhana. Anda dapat menentukan opsi untuk parser yang dipilih, memfilter dan mengurutkan hasil, dan menelusuri aspek yang dikonfigurasi. Hit pencarian secara otomatis disorot dalam hasil pencarian. Untuk informasi tentang bagaimana hal ini dilakukan, lihat [Menyoroti Pencarian Hits di Amazon CloudSearch](#). Anda juga dapat memilih pemberi saran untuk mendapatkan saran saat memasukkan istilah di bidang Pencarian. (Anda harus mengonfigurasi sugester sebelum Anda bisa mendapatkan saran. Untuk informasi lebih lanjut lihat [Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch](#).)

Secara default, hasil diurutkan menurut skor relevansi yang dihasilkan secara otomatis, `_score`. Untuk informasi tentang menyesuaikan bagaimana hasil diberi peringkat, lihat [Menyortir Hasil di Amazon CloudSearch](#).

Untuk mencari domain Anda

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih nama domain Anda untuk membuka konfigurasinya.
3. Pilih Jalankan pencarian pengujian.
4. Untuk melakukan pencarian teks sederhana, masukkan kueri penelusuran dan pilih Jalankan. Secara default, semua `text` dan `text-array` bidang dicari.

Untuk mencari bidang tertentu, perluas Opsi dan masukkan daftar bidang yang dipisahkan koma yang ingin Anda cari di bidang bidang Penelusuran. Anda dapat menambahkan bobot ke setiap bidang dengan tanda sisipan (^) untuk mengontrol kepentingan relatif setiap bidang dalam hasil pencarian. Misalnya, menentukan hit `title^5, description` bobot di `title` lapangan lima kali lebih banyak daripada hit di `description` lapangan saat menghitung skor relevansi untuk setiap dokumen yang cocok.

Untuk menggunakan sintaks kueri terstruktur, pilih Structured dari menu Query parser. Setelah Anda memilih parser kueri terstruktur, masukkan kueri terstruktur Anda di bidang Penelusuran dan pilih Jalankan. Misalnya, untuk menemukan semua film dengan bintang dalam judul yang dirilis pada tahun 2000 atau sebelumnya, Anda dapat memasukkan: `(and title:'star' year:{,2000})`.

Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#). Untuk mengirimkan Lucene atau DisMax kueri, pilih parser kueri yang sesuai.

Anda dapat menentukan opsi tambahan untuk parser kueri yang dipilih untuk mengonfigurasi operator default dan mengontrol operator mana yang dapat digunakan dalam kueri. Untuk informasi selengkapnya, lihat [Parameter Permintaan Pencarian](#).

Anda dapat menyalin dan menempelkan URL permintaan untuk mengirimkan permintaan dan melihat respons dari browser Web. Permintaan dapat dikirim melalui HTTP atau HTTPS.

## Membangun Kueri Senyawa di Amazon CloudSearch

Anda dapat menggunakan parser kueri terstruktur untuk menggabungkan ekspresi kecocokan menggunakan Boolean `and`, `or`, dan `not` operator. Untuk memilih parser kueri terstruktur, Anda menyertakan `q.parser=structured` dalam kueri Anda. Operator kueri terstruktur ditentukan sebagai operator awalan. Sintaksnya adalah:

- `(and boost=N EXPRESSION1 EXPRESSION2 ... EXPRESSIONn)`
- `(or boost=N EXPRESSION1 EXPRESSION2 ... EXPRESSIONn)`
- `(not boost=N EXPRESSION)`

Misalnya, kueri berikut cocok dengan semua film dalam kumpulan data sampel yang berisi bintang dalam judul, dan Harrison Ford atau William Shatner muncul di `actors` lapangan, tetapi Zachary Quinto tidak.

```
(and title:'star' (or actors:'Harrison Ford' actors:'William Shatner')(not actors:'Zachary Quinto'))
```

Saat menggunakan operator kueri terstruktur, Anda menentukan nama operator, opsi untuk operator, dan kemudian ekspresi kecocokan yang dioperasikan, `(OPERATOR OPTIONS EXPRESSION)`. Ekspresi pencocokan dapat berupa string teks sederhana, atau subklausa dari kueri majemuk Anda. Opsi apa pun harus ditentukan sebelum persyaratan. Misalnya, `(and (not field=genres 'Sci-Fi')(or (term field=title boost=2 'star')(term field=plot 'star')))`.

Tanda kurung mengontrol urutan evaluasi ekspresi. Ketika ekspresi diapit dalam tanda kurung, ekspresi itu dievaluasi terlebih dahulu, dan kemudian nilai yang dihasilkan digunakan dalam evaluasi sisa kueri majemuk.

**⚠ Important**

Anda harus mengkodekan URL karakter khusus dalam string kueri. Misalnya, Anda harus menyandikan = operator dalam kueri terstruktur sebagai `%3D:(term+field%3Dtitle+'star')`. Amazon CloudSearch mengembalikan `InvalidQueryString` kesalahan jika karakter khusus tidak dikodekan URL. [Untuk referensi lengkap pengkodean URL, lihat Referensi Pengkodean URL HTML W3C.](#)

Misalnya, kueri berikut mencari `title` bidang untuk frasa `star wars` dan mengecualikan kecocokan yang memiliki nilai kurang dari 2000 di `year` bidang.

```
(and (phrase field='title' 'star wars') (not (range field=year {,2000})))
```

Untuk mengirimkan permintaan pencarian ini, Anda perlu menyandikan string kueri dan menentukan parser structured kueri dengan parameter. `q.parser`

```
http://search-movies-rr2f34ofg56xneuemujamut52i.us-east-1.cloudsearch.  
amazonaws.com/2013-01-01/search?q=(and+(phrase+field='title'+ 'star wars')+(not+(range  
+field%3Dyear+{,2000})))&q.parser=structured
```

Sintaks kueri terstruktur memungkinkan Anda untuk menggabungkan pencarian terhadap beberapa bidang. Jika Anda tidak menentukan bidang yang akan dicari, semua `text` dan `text-array` bidang akan dicari. Misalnya, kueri berikut mencari semua `text` dan `text-array` bidang untuk istilah bintang, dan mengecualikan dokumen yang berisi Zachary Quinto di bidang. `actors`

```
(and 'star' (not actors:'Zachary Quinto'))
```

Anda dapat menentukan boost nilai untuk meningkatkan pentingnya satu ekspresi dalam kueri majemuk dalam kaitannya dengan yang lain. Nilai boost meningkatkan skor dokumen yang cocok. Misalnya, kueri berikut meningkatkan kecocokan untuk istilah bintang jika muncul di `title` bidang, bukan di `description` bidang.

```
(and (range field=year [2013,}) (or (term field=title boost=2 'star') (term field=plot  
'star'))
```

Nilai boost harus lebih besar dari nol.



Selain and, dan ornot, sintaks pencarian CloudSearch terstruktur Amazon mendukung beberapa operator khusus:

- `matchall`—Cocokkan setiap dokumen di domain. Sintaks: `matchall`.
- `near`—Mendukung kueri frase ceroboh. `distance` Nilai menentukan jumlah maksimum kata yang dapat memisahkan kata-kata dalam frasa; misalnya, `(near field='plot' distance=4 'naval mutiny demonstration')`. Gunakan `near` operator untuk mengaktifkan pencocokan ketika istilah yang ditentukan berada dalam jarak dekat, tetapi tidak berdekatan. Untuk informasi selengkapnya tentang pencarian frase ceroboh, lihat [Mencari Frase](#) Sintaks: `(near field=FIELD distance=N boost=N 'STRING')`.
- `phrase`—Mencari frasa di `text` atau `text-array` bidang; misalnya, `(phrase field="title" 'teenage mutant ninja')` Mendukung peningkatan dokumen yang cocok dengan ekspresi. Untuk informasi selengkapnya tentang pencarian frasa, lihat [Mencari Frase](#). Sintaks: `(phrase field=FIELD boost=N 'STRING')`.
- `prefix`—Mencari bidang teks, teks-array, literal, atau array literal untuk awalan yang ditentukan diikuti oleh nol atau lebih karakter; misalnya, `(prefix field='title' 'wait')` Mendukung peningkatan dokumen yang cocok dengan ekspresi. Untuk informasi selengkapnya tentang penelusuran awalan, lihat [Mencari Prefiks](#) .Syntax: `(prefix field=FIELD boost=N 'STRING')`
- `range`—Mencari rentang nilai dalam bidang numerik; misalnya, `(range field=year [2000,2013])` Untuk informasi selengkapnya tentang penelusuran rentang, lihat [Mencari Rentang Nilai](#). Sintaks: `(range field=FIELD boost=N RANGE)`.
- `term`—Mencari istilah atau nilai individu di bidang apa pun; misalnya, `(and (term field=title 'star')(term field=year 1977))` Sintaks: `(term field=FIELD boost=N 'STRING' |VALUE)`.

Untuk informasi selengkapnya tentang mencari jenis data tertentu, lihat bagian berikut. Untuk informasi selengkapnya tentang sintaks penelusuran terstruktur, lihat [Sintaks Pencarian Terstruktur](#).

## Mencari Teks di Amazon CloudSearch

Anda dapat mencari bidang teks dan literal untuk string teks:

- `Text` dan `text-array` bidang selalu dapat dicari. Anda dapat mencari istilah dan frasa individual. Pencarian di dalam `text` dan `text-array` bidang tidak peka huruf besar/kecil.

- Literal dan `literal-array` bidang hanya dapat dicari jika pencarian diaktifkan di opsi pengindeksan domain. Anda dapat mencari kecocokan yang tepat dari string pencarian Anda. Pencarian di bidang literal peka huruf besar/kecil.

Jika Anda menggunakan parser kueri sederhana atau tidak menentukan bidang saat mencari dengan parser kueri terstruktur, secara default semua `text` dan `text-array` bidang dicari. Bidang literal tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari dengan `q.options` parameter.

Anda dapat mencari bidang ID dokumen unik seperti bidang teks apa pun. Untuk merujuk ke bidang ID dokumen dalam permintaan pencarian, Anda menggunakan nama `bidang_id`. ID dokumen selalu dikembalikan dalam hasil pencarian.

### Topik

- [Mencari Istilah Individu di Amazon CloudSearch](#)
- [Mencari Frasa di Amazon CloudSearch](#)
- [Mencari String Literal di Amazon CloudSearch](#)
- [Mencari Awalan di Amazon CloudSearch](#)

## Mencari Istilah Individu di Amazon CloudSearch

Saat Anda mencari `text` dan `text-array` bidang untuk istilah individual, Amazon CloudSearch menemukan semua dokumen yang berisi istilah pencarian di mana saja dalam bidang yang ditentukan, dalam urutan apa pun. Misalnya, dalam data film sampel, `title` bidang dikonfigurasi sebagai `text` bidang. Jika Anda mencari `title` bidang untuk bintang, Anda akan menemukan semua film yang berisi bintang di mana saja di `title` lapangan, seperti bintang, perang bintang, dan bintang lahir. Ini berbeda dari `literal` bidang pencarian, di mana nilai bidang harus identik dengan string pencarian untuk dianggap cocok.

Parser `simple` kueri menyediakan cara mudah untuk mencari `text` dan `text-array` bidang untuk satu atau lebih istilah. Parser `simple` kueri digunakan secara default kecuali Anda menggunakan `q.parser` parameter untuk menentukan parser kueri yang berbeda.

Misalnya, untuk mencari `katniss`, tentukan `katniss` dalam string kueri. Secara default, Amazon CloudSearch menyertakan semua bidang yang diaktifkan kembali dalam hasil pencarian. Anda dapat menentukan `return` parameter untuk menentukan bidang mana yang ingin Anda kembalikan.

```
https://search-domainname-domainid.us-east-1.cloudsearch.amazonaws.com/  
2013-01-01/search?q=katniss&return=title
```

Secara default, respons dikembalikan di JSON:

```
{  
  "status": {  
    "rid": "rd+5+r0oMAo6swY=",  
    "time-ms": 9  
  },  
  "hits": {  
    "found": 3,  
    "start": 0,  
    "hit": [  
      {  
        "id": "tt1951265",  
        "fields": {  
          "title": "The Hunger Games: Mockingjay - Part 1"  
        }  
      },  
      {  
        "id": "tt1951264",  
        "fields": {  
          "title": "The Hunger Games: Catching Fire"  
        }  
      },  
      {  
        "id": "tt1392170",  
        "fields": {  
          "title": "The Hunger Games"  
        }  
      }  
    ]  
  }  
}
```

Untuk menentukan beberapa istilah, pisahkan istilah dengan spasi. Misalnya: `star wars`. Saat Anda menentukan beberapa istilah penelusuran, secara default dokumen harus berisi semua istilah yang akan dianggap cocok. Istilah dapat terjadi di mana saja dalam bidang teks, dalam urutan apa pun.

Secara default, semua `text` dan `text-array` bidang dicari saat Anda menggunakan parser kueri sederhana. Anda dapat menentukan bidang mana yang ingin Anda cari dengan menentukan `q.options` parameter. Misalnya, kueri ini membatasi pencarian ke `description` bidang `title` dan dan meningkatkan pentingnya kecocokan di bidang dibandingkan kecocokan di `title` bidang. `description`

```
q=star wars&q.options={fields: ['title^5','description']}
```

Bila Anda menggunakan parser kueri sederhana, Anda dapat menggunakan awalan berikut untuk menunjuk istilah individual seperti yang diperlukan, opsional, atau untuk dikecualikan dari hasil pencarian:

- **+**—dokumen yang cocok harus berisi istilah. Ini adalah default—memisahkan istilah dengan spasi sama dengan mendahuluinya dengan awalan. **+**
- **-**—kecualikan dokumen yang berisi istilah dari hasil pencarian. **-**Operator hanya berlaku untuk persyaratan individual. Misalnya, untuk mengecualikan dokumen yang berisi istilah bintang di bidang pencarian default, tentukan: `-star`. Mencari `search?q=-star wars` mengambil semua dokumen yang tidak mengandung istilah bintang, tetapi mengandung istilah perang.
- **|**—termasuk dokumen yang berisi istilah dalam hasil pencarian, bahkan jika mereka tidak mengandung istilah lain. **|**Operator hanya berlaku untuk persyaratan individual. Misalnya, untuk menyertakan dokumen yang berisi salah satu dari dua istilah, tentukan: `term1 | term2`. Pencarian `search?q=star wars | trek` termasuk dokumen yang berisi bintang dan perang, atau istilah perjalanan.

Awalan ini hanya berlaku untuk istilah individual dalam kueri sederhana. Untuk membuat kueri majemuk, Anda perlu menggunakan parser kueri terstruktur, bukan parser kueri sederhana. Misalnya, untuk mencari istilah `star` dan `wars` menggunakan parser kueri terstruktur, Anda akan menentukan:

```
(and 'star' 'wars')
```

Perhatikan bahwa kueri ini cocok dengan dokumen yang berisi setiap istilah di salah satu bidang yang dicari. Persyaratan tidak harus berada di bidang yang sama untuk dianggap cocok. Namun, jika Anda menentukan `(and 'star wars' 'luke')`, bintang dan perang harus terjadi dalam bidang yang sama, dan `luke` dapat terjadi di salah satu bidang.

Jika Anda tidak menentukan bidang apa pun saat menggunakan parser `structured` kueri, semua `text` dan `text-array` bidang dicari secara default, sama seperti dengan parser. `simple` Demikian

pula, Anda dapat menggunakan `q.options` parameter untuk mengontrol bidang mana yang dicari dan untuk meningkatkan pentingnya bidang yang dipilih. Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).

Anda juga dapat melakukan pencarian fuzzy dengan parser kueri sederhana. Untuk melakukan pencarian fuzzy, tambahkan `~` operator dan nilai yang menunjukkan berapa banyak istilah dapat berbeda dari string kueri pengguna dan masih dianggap cocok. Misalnya, `planet~1` penelusuran yang menentukan untuk istilah planet dan memungkinkan kecocokan berbeda hingga satu karakter, yang berarti hasilnya akan mencakup hit untuk planet.

## Mencari Frasa di Amazon CloudSearch

Saat Anda mencari frasa, Amazon CloudSearch menemukan semua dokumen yang berisi frasa lengkap dalam urutan yang ditentukan. Anda juga dapat melakukan pencarian frase ceroboh di mana istilah muncul dalam jarak yang ditentukan satu sama lain.

Untuk mencocokkan frasa lengkap daripada istilah individual dalam frasa saat Anda mencari dengan parser kueri sederhana, lampirkan frasa dalam tanda kueri ganda. Misalnya, kueri berikut mencari frasa dengan cinta.

```
q="with love"
```

Untuk melakukan pencarian frase ceroboh dengan parser kueri sederhana, tambahkan `~` operator dan nilai jarak. Nilai jarak menentukan jumlah maksimum kata yang dapat memisahkan kata-kata dalam frasa. Misalnya, kueri berikut mencari istilah dengan cinta dalam tiga kata satu sama lain.

```
q="with love"~3
```

Dalam kueri majemuk, Anda menggunakan `phrase` operator untuk menentukan frasa yang ingin Anda cocokkan; misalnya:

```
(phrase field=title 'star wars')
```

Untuk melakukan pencarian frase ceroboh dalam kueri majemuk, Anda menggunakan `near` operator. `near` operator memungkinkan Anda untuk menentukan frasa yang Anda cari dan seberapa jauh jarak istilah dapat berada dalam bidang dan masih dianggap cocok. Misalnya, kueri berikut cocok dengan dokumen yang memiliki istilah bintang dan perang tidak lebih dari tiga kata terpisah di `title` lapangan.

```
(near field=title distance=3 'star wars')
```

Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).

## Mencari String Literal di Amazon CloudSearch

Saat Anda mencari bidang literal untuk string, Amazon hanya CloudSearch mengembalikan dokumen yang berisi kecocokan persis untuk string pencarian lengkap di bidang yang ditentukan, termasuk kasus. Misalnya, jika `title` bidang dikonfigurasi sebagai bidang literal dan Anda mencari Bintang, nilai `title` bidang harus Bintang untuk dianggap cocok— bintang, perang bintang, dan bintang lahir tidak akan disertakan dalam hasil pencarian. Ini berbeda dari bidang teks, di mana pencarian tidak peka huruf besar/kecil dan istilah pencarian yang ditentukan dapat muncul di mana saja dalam bidang dalam urutan apa pun.

Untuk mencari bidang literal, awali string pencarian dengan nama bidang literal yang ingin Anda cari, diikuti dengan titik dua. String pencarian harus dilampirkan dalam tanda kutip tunggal. Misalnya, kueri berikut mencari string literal Sci-Fi.

```
genres:'Sci-Fi'
```

Contoh ini mencari bidang genre dari setiap dokumen dan mencocokkan semua dokumen yang bidang genrenya berisi nilai Sci-Fi. Untuk menjadi kecocokan, nilai bidang harus sama persis untuk string pencarian, termasuk kasus. Misalnya, dokumen yang berisi nilai Sci-Fi di bidang genre tidak akan disertakan dalam hasil pencarian jika Anda mencari sci-fi atau sci-fi dewasa muda.

Dalam kueri majemuk, Anda menggunakan sintaks `term operator` untuk mencari bidang literal. Misalnya, (`term field=genres 'Sci-Fi'`). Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).

Anda dapat menggunakan bidang literal bersama dengan faset untuk memungkinkan pengguna menelusuri hasil sesuai dengan atribut faset. Untuk informasi lebih lanjut tentang faceting, lihat [Mendapatkan dan Menggunakan Informasi Facet di Amazon CloudSearch](#).

## Mencari Awalan di Amazon CloudSearch

Anda dapat mencari `texttext-array`, `literal`, dan `literal-array` bidang untuk awalan daripada istilah lengkap. Ini cocok dengan hasil yang berisi awalan diikuti oleh nol atau lebih karakter. Anda harus menentukan setidaknya satu karakter sebagai awalan. (Untuk mencocokkan

semua dokumen, gunakan `matchAll` operator dalam kueri terstruktur.) Secara umum, Anda harus menggunakan awalan yang berisi setidaknya dua karakter untuk menghindari pencocokan jumlah dokumen yang berlebihan.

Saat Anda mencari `text-array` bidang `text` atau, istilah yang cocok dengan awalan dapat muncul di mana saja di dalam konten bidang. Saat Anda mencari bidang literal, seluruh string pencarian, hingga dan termasuk karakter awalan, harus sama persis.

- Parser kueri sederhana—gunakan operator wildcard `*` (tanda bintang) untuk mencari awalan, misalnya `pre*`
- Parser kueri terstruktur—gunakan prefix operator untuk mencari awalan, misalnya prefix `'pre'`

Misalnya, kueri berikut mencari awalan `oce` di bidang judul dan mengembalikan judul setiap hit:

```
q=oce*&q.options={fields:['title']}&return=title
```

Jika Anda melakukan pencarian ini terhadap data sampel film, itu kembali sebagai `Ocean's Eleven` dan `Ocean's Twelve`:

```
{
  "status": {
    "rid": "hIbIxb8oRAo6swY=",
    "time-ms": 2
  },
  "hits": {
    "found": 2,
    "start": 0,
    "hit": [
      {
        "id": "tt0240772",
        "fields": {
          "title": "Ocean's Eleven"
        }
      },
      {
        "id": "tt0349903",
        "fields": {
          "title": "Ocean's Twelve"
        }
      }
    ]
  }
}
```

```
}  
  }  
] }  
}
```

Dalam kueri gabungan, Anda menggunakan prefix operator untuk mencari awalan. Misalnya, untuk mencari title bidang untuk awalan oce, Anda menentukan:

```
q.parser=structured&q=(prefix field%3Dtitle 'oce')
```

Perhatikan pengkodean URL. Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).

#### Note

Saat melakukan pencarian wildcard pada bidang teks, ingatlah bahwa Amazon CloudSearch memberi token pada bidang teks selama pengindeksan dan melakukan stemming sesuai dengan skema analisis yang dikonfigurasi untuk bidang tersebut. Biasanya, Amazon CloudSearch melakukan pemrosesan teks yang sama pada permintaan pencarian. Namun, saat Anda mencari awalan dengan operator wildcard (\*) atau prefix operator, tidak ada stemming yang dilakukan pada awalan. Ini berarti bahwa pencarian awalan yang diakhiri tidak akan cocok dengan versi tunggal dari istilah tersebut. Ini bisa terjadi untuk istilah apa pun yang berakhir, bukan hanya jamak. Misalnya, jika Anda mencari actor bidang dalam data film sampelAnders, ada tiga film yang cocok. Jika Anda mencariAnder\*, Anda mendapatkan film-film itu serta beberapa lainnya. Namun, jika Anda mencari tidak Anders\* ada kecocokan. Ini karena istilah disimpan dalam indeks sebagaiander, anders tidak muncul dalam indeks. Untuk informasi selengkapnya tentang cara Amazon CloudSearch memproses teks dan bagaimana hal itu dapat memengaruhi penelusuran, lihat[Pemrosesan Teks di Amazon CloudSearch](#).

## Mencari Angka di Amazon CloudSearch

Anda dapat menggunakan kueri terstruktur untuk mencari bidang numerik yang diaktifkan penelusuran untuk nilai atau [rentang nilai](#) tertentu. Amazon CloudSearch mendukung empat jenis bidang numerik:double,, double-arrayint, danint-array. Untuk informasi selengkapnya, lihat [configure indexing options](#).



Sintaks dasar untuk mencari bidang untuk satu nilai adalah **FIELD:VALUE**. Misalnya, `year:2010` mencari data film sampel untuk film yang dirilis pada tahun 2010.

Anda harus menggunakan parser kueri terstruktur untuk menggunakan sintaks bidang. Perhatikan bahwa nilai numerik tidak terlampir dalam tanda kutipan—tanda kutip menunjuk nilai sebagai string. Untuk mencari rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah, dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Untuk informasi selengkapnya, lihat [Mencari Rentang Nilai](#).

Dalam kueri majemuk, Anda menggunakan sintaks `term operator` untuk mencari nilai tunggal: (`term field=year 2010`).

## Mencari Tanggal dan Waktu di Amazon CloudSearch

Anda dapat menggunakan kueri terstruktur untuk mencari kolom tanggal yang diaktifkan penelusuran untuk tanggal dan waktu tertentu atau rentang [tanggal-waktu](#). Amazon CloudSearch mendukung dua jenis bidang tanggal, `date` dan `date-array`. Untuk informasi selengkapnya, lihat [configure indexing options](#).

Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut [IETF RFC3339](#): `yyyy-mm-ddTHH:mm:ss.SSSZ` Di UTC, misalnya, 17:00 23 Agustus 1970 adalah: `1970-08-23T17:00:00Z` Perhatikan bahwa Anda juga dapat menentukan detik pecahan saat menentukan waktu di UTC. Misalnya, `1967-01-31T23:20:50.650Z`.

Untuk mencari tanggal (atau waktu) di `date` bidang, Anda harus melampirkan string tanggal dalam tanda kutip tunggal. Misalnya, kedua kueri berikut mencari data film untuk semua film yang dirilis pada tengah malam pada tanggal 25 Desember 2001:

```
q.parser=structured&q=release_date:'2001-12-25T00:00:00Z'  
q.parser=structured&q=(term field%3Drelease_date '2001-12-25T00:00:00Z')
```

Untuk mencari sepanjang hari, lihat [the section called “Mencari Rentang Tanggal”](#).

## Mencari Rentang Nilai di Amazon CloudSearch

Anda dapat menggunakan kueri terstruktur untuk mencari bidang untuk rentang nilai. Untuk menentukan rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Penjepit persegi, [atau], menunjukkan bahwa ikatan termasuk dalam rentang, penjepit keriting, {atau}, tidak termasuk batas.

Misalnya, untuk mencari kumpulan data sampel untuk film yang dirilis dari 2008 hingga 2010 (inklusif), tentukan rentang sebagai `[2008, 2010]`.

Untuk menentukan rentang ujung terbuka, hilangkan batas. Misalnya, `year : [2002, ]` mencocokkan semua film yang dirilis dari tahun 2002 dan seterusnya, dan `year : { , 1970]` cocok dengan semua film yang dirilis hingga tahun 1970. Ketika Anda menghilangkan ikatan, Anda harus menggunakan penjepit keriting.

Dalam kueri majemuk, Anda menggunakan sintaks `range` operator untuk mencari berbagai nilai; misalnya: `(range field=year [1967, ])`.

## Mencari Rentang Tanggal

Untuk mencari rentang tanggal (atau waktu) di `date` bidang, Anda menggunakan sintaks rentang kurung yang sama yang Anda gunakan untuk nilai numerik, tetapi Anda harus melampirkan string tanggal dalam tanda kutip tunggal. Misalnya, permintaan berikut mencari data film untuk semua film dengan tanggal rilis 1 Januari 2013 atau lebih baru:

```
q.parser=structured&q=release_date:['2013-01-01T00:00:00Z',]
```

Gunakan sintaks berikut untuk mencari rentang tetap:

```
q.parser=structured&q=release_date:['2013-01-01T00:00:00Z', '2013-01-02T23:59:59Z']
```

## Mencari Rentang Lokasi

Anda dapat melakukan pencarian kotak pembatas dengan mencari berbagai lokasi. Untuk mencari rentang lokasi di `latlon` bidang, Anda menggunakan sintaks rentang kurung yang sama yang Anda gunakan untuk nilai numerik, tetapi Anda harus menyertakan pasangan garis lintang/bujur dalam tanda kutip tunggal.

Misalnya, jika Anda menyertakan `location` bidang di setiap dokumen, Anda dapat menentukan filter kotak pembatas Anda sebagai `location:['nn.n, nn.n', 'nn.n, nn.n']`. Dalam contoh berikut, kecocokan untuk restoran disaring sehingga hanya kecocokan di dalam area pusat kota Paso Robles, CA yang disertakan dalam hasil.

```
q='restaurant'&fq=location:
['35.628611, -120.694152', '35.621966, -120.686706']&q.parser=structured
```

Untuk informasi selengkapnya, lihat [location-based searching and sorting](#).

## Mencari Rentang Teks

Anda juga dapat mencari teks atau bidang literal untuk rentang nilai menggunakan sintaks rentang kurung. Seperti tanggal, string teks harus dilampirkan dalam tanda kutip tunggal. Misalnya, permintaan berikut akan mencari data film untuk berbagai ID dokumen. Untuk mereferensikan ID dokumen, Anda menggunakan nama bidang khusus `_id`.

```
_id:['tt1000000','tt1005000']
```

## Pencarian dan Peringkat Hasil berdasarkan Lokasi Geografis di Amazon CloudSearch

Jika Anda menyimpan lokasi dalam data dokumen menggunakan `latlon` bidang, Anda dapat menggunakan `haversin` fungsi dalam CloudSearch ekspresi Amazon untuk menghitung jarak antara dua lokasi. Menyimpan lokasi dengan data dokumen Anda juga memungkinkan Anda untuk dengan mudah mencari di dalam area tertentu.

Topik

- [Mencari Dalam Area di Amazon CloudSearch](#)
- [Menyortir Hasil berdasarkan Jarak di Amazon CloudSearch](#)

## Mencari Dalam Area di Amazon CloudSearch

Untuk mengaitkan lokasi dengan dokumen pencarian, Anda dapat menyimpan lintang dan bujur lokasi di `latlon` bidang menggunakan notasi derajat desimal. Nilai-nilai ditentukan sebagai daftar dipisahkan koma, `lat,lon` —misalnya. `35.628611,-120.694152` Mengaitkan lokasi dengan dokumen memungkinkan Anda untuk dengan mudah membatasi klik pencarian ke area tertentu dengan parameter. `fq`

Untuk menggunakan kotak pembatas untuk membatasi hasil ke area tertentu

1. Tentukan garis lintang dan bujur sudut kiri atas dan kanan bawah area yang Anda minati.
2. Gunakan `fq` parameter untuk memfilter dokumen yang cocok menggunakan koordinat kotak pembatas tersebut. Misalnya, jika Anda menyertakan `location` bidang di setiap

dokumen, Anda dapat menentukan filter kotak pembatas Anda sebagai `fq=location: ['nn.n,nn.n', 'nn.n,nn.n']` . Dalam contoh berikut, kecocokan untuk restoran disaring sehingga hanya kecocokan di dalam area pusat kota Paso Robles, CA yang disertakan dalam hasil.

```
q='restaurant'&fq=location:
['35.628611,-120.694152','35.621966,-120.686706']&q.parser=structured
```

## Menyortir Hasil berdasarkan Jarak di Amazon CloudSearch

Anda dapat menentukan ekspresi sebagai bagian dari permintaan pencarian Anda untuk mengurutkan hasil berdasarkan jarak. CloudSearch Ekspresi Amazon mendukung `haversin` fungsi, yang menghitung jarak lingkaran besar antara dua titik pada bola menggunakan garis lintang dan bujur setiap titik. (Untuk informasi lebih lanjut, lihat [rumus Haversine](#).) Jarak yang dihasilkan dikembalikan dalam kilometer.

Untuk menghitung jarak antara setiap dokumen yang cocok dan pengguna, Anda meneruskan lokasi pengguna ke dalam `haversin` fungsi dan mereferensikan lokasi dokumen yang disimpan di `latlon` bidang. Anda menentukan garis lintang dan bujur pengguna dalam notasi derajat desimal dan mengakses garis lintang dan bujur yang disimpan dalam `as` dan `latlon` `FIELD.latitude` `FIELD.longitude` Sebagai contoh, `expr.distance=haversin(userlat,userlon,location.latitude,location.longitude)`.

Untuk menggunakan ekspresi untuk mengurutkan hasil pencarian, Anda menentukan `sort` parameter.

Misalnya, kueri berikut mencari restoran dan mengurutkan hasil berdasarkan jarak dari pengguna.

```
q=restaurant&expr.distance=haversin(35.621966,-120.686706,location.latitude,location.longitude)
asc
```

Perhatikan bahwa Anda harus secara eksplisit menentukan arah pengurutan, `asc` atau `desc`

Anda dapat menyertakan jarak yang dihitung untuk setiap dokumen dalam hasil pencarian dengan menentukan nama ekspresi dengan `return` parameter. Sebagai contoh, `return=distance`.

Anda juga dapat menggunakan nilai jarak dalam ekspresi yang lebih kompleks untuk mempertimbangkan karakteristik lain, seperti `_score` relevansi dokumen. Dalam contoh berikut, ekspresi peringkat kedua menggunakan perhitungan dokumen `distance` dan relevansinya. `_score`

```
expr.distance=haversin(38.958687,-77.343149,latitude,longitude)&expr.myrank=_score/  
log10(distance)&sort=myrank+desc
```

### Tip

Agar contoh kueri ini berfungsi, Anda harus [mengonfigurasi indeks Anda](#) dengan `location` bidang dan memiliki `location` data dalam dokumen Anda:

```
{  
  "fields": {  
    "location": "40.05830,-74.40570"  
  }  
}
```

Jika bidang tidak ada, Anda mungkin menerima pesan galat berikut saat melakukan pencarian:

```
Syntax error in query: field (location) does not exist.
```

Untuk informasi selengkapnya tentang menggunakan ekspresi untuk mengurutkan hasil pencarian, lihat [Mengontrol Hasil Pencarian](#).

## Mencari Data DynamoDB dengan Amazon CloudSearch

Anda dapat menentukan tabel DynamoDB sebagai sumber saat mengonfigurasi opsi pengindeksan atau mengunggah data ke domain pencarian melalui konsol. Ini memungkinkan Anda untuk dengan cepat mengatur domain pencarian untuk bereksperimen dengan mencari data yang disimpan dalam tabel database DynamoDB.

Agar domain penelusuran tetap sinkron dengan perubahan pada tabel, Anda dapat mengirim pembaruan ke tabel dan domain pencarian Anda, atau Anda dapat memuat seluruh tabel secara berkala ke dalam domain pencarian baru.

### Topik

- [Mengkonfigurasi CloudSearch Domain Amazon untuk Mencari Data DynamoDB](#)
- [Mengunggah Data ke Amazon CloudSearch dari DynamoDB](#)

- [Menyinkronkan Domain Pencarian dengan Tabel DynamoDB](#)

## Mengkonfigurasi CloudSearch Domain Amazon untuk Mencari Data DynamoDB

Cara termudah untuk mengonfigurasi domain pencarian untuk mencari data DynamoDB adalah dengan menggunakan konsol Amazon. CloudSearch Wizard konfigurasi konsol menganalisis data tabel Anda dan menyarankan opsi pengindeksan berdasarkan atribut dalam tabel. Anda dapat memodifikasi konfigurasi yang disarankan untuk mengontrol atribut tabel mana yang diindeks.

### Note

Untuk mengunggah data dari DynamoDB, Anda harus memiliki izin untuk mengakses layanan dan sumber daya yang ingin Anda unggah. Untuk informasi selengkapnya, lihat [Menggunakan IAM untuk Mengontrol Akses ke Sumber Daya DynamoDB](#).

Bila Anda secara otomatis mengonfigurasi domain pencarian dari tabel DynamoDB, maksimal 200 atribut unik dapat dipetakan ke bidang indeks. (Anda tidak dapat mengonfigurasi lebih dari 200 bidang untuk domain pencarian, jadi Anda hanya dapat mengunggah data dari tabel DynamoDB dengan 200 atau lebih sedikit atribut.) Saat Amazon CloudSearch mendeteksi atribut yang memiliki sejumlah kecil nilai berbeda, bidang tersebut diaktifkan faset dalam konfigurasi yang disarankan.

### Important

Bila Anda menggunakan tabel DynamoDB untuk mengkonfigurasi domain, data tidak secara otomatis diunggah ke domain untuk pengindeksan. Anda harus mengunggah data untuk pengindeksan sebagai langkah terpisah setelah Anda mengonfigurasi domain.

## Mengonfigurasi Domain untuk Mencari DynamoDB menggunakan Konsol Amazon CloudSearch

Anda dapat menggunakan CloudSearch konsol Amazon untuk menganalisis data dari tabel DynamoDB untuk mengonfigurasi domain pencarian. Maksimal 5 MB dibaca dari tabel terlepas dari ukuran tabel. Secara default, Amazon CloudSearch membaca dari awal tabel. Anda dapat menentukan tombol mulai untuk mulai membaca dari item tertentu.

Untuk mengkonfigurasi domain pencarian menggunakan tabel DynamoDB

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Dari panel navigasi kiri, pilih Domain.
3. Pilih nama domain untuk membuka panel detailnya.
4. Buka tab Opsi pengindeksan dan pilih Wisaya konfigurasi.
5. Pilih Amazon DynamoDB.
6. Pilih tabel DynamoDB yang ingin Anda analisis.
  - Untuk membatasi unit kapasitas baca yang dapat dikonsumsi saat membaca dari tabel, masukkan persentase maksimum unit kapasitas baca yang ingin Anda gunakan.
  - Untuk mulai membaca dari item tertentu, tentukan kunci Mulai hash. Jika tabel menggunakan kunci primer tipe hash dan rentang, tentukan atribut hash dan atribut range untuk item tersebut.
7. Pilih Berikutnya.
8. Tinjau konfigurasi yang disarankan. Anda dapat mengedit bidang ini dan menambahkan bidang tambahan.
9. Setelah selesai, pilih Konfirmasi.
10. Jika Anda belum mengunggah data ke domain Anda, kosongkan kotak centang Jalankan pengindeksan sekarang untuk keluar tanpa pengindeksan. Jika Anda selesai membuat perubahan konfigurasi dan siap untuk mengindeks data Anda dengan konfigurasi baru, pastikan Run indexing now dipilih. Saat Anda siap menerapkan perubahan, pilih Selesai.

## Mengunggah Data ke Amazon CloudSearch dari DynamoDB

Anda dapat mengunggah data DynamoDB ke domain pencarian melalui konsol CloudSearch Amazon atau dengan alat baris perintah CloudSearch Amazon. Saat Anda mengunggah data dari tabel DynamoDB, CloudSearch Amazon mengonversinya menjadi kumpulan dokumen sehingga dapat diindeks. Anda memilih menentukan bidang indeks untuk setiap atribut dalam konfigurasi domain Anda. Untuk informasi selengkapnya, lihat [Mengkonfigurasi CloudSearch Domain Amazon untuk Mencari Data DynamoDB](#).

Anda dapat mengunggah data dari lebih dari satu tabel DynamoDB ke domain Amazon yang sama. CloudSearch Namun, perlu diingat bahwa Anda dapat mengunggah maksimal 200 atribut dari semua

tabel yang digabungkan. Jika item dengan kunci yang sama muncul di lebih dari satu tabel yang diunggah, item yang diterapkan terakhir akan menimpa semua versi sebelumnya.

Saat mengonversi data tabel ke kumpulan dokumen, Amazon CloudSearch menghasilkan dokumen untuk setiap item yang dibacanya dari tabel, dan mewakili setiap atribut item sebagai bidang dokumen. ID unik untuk setiap dokumen dibaca dari atribut `docid` item (jika ada) atau diberi nilai alfanumerik berdasarkan kunci utama.

Saat Amazon CloudSearch menghasilkan dokumen untuk item tabel:

- Set string dan set angka direpresentasikan sebagai bidang multi-nilai. Jika kumpulan DynamoDB berisi lebih dari 100 nilai, hanya 100 nilai pertama yang ditambahkan ke bidang multi-nilai.
- Atribut biner DynamoDB diabaikan.
- Nama atribut dimodifikasi agar sesuai dengan konvensi CloudSearch penamaan Amazon untuk nama bidang:
  - Semua huruf besar dikonversi menjadi huruf kecil.
  - Jika nama atribut DynamoDB tidak dimulai dengan huruf, nama bidang diawali dengan `f_`.
  - Karakter apa pun selain `a-z`, `0-9`, dan `_` (garis bawah) diganti dengan garis bawah. Jika transformasi ini menghasilkan nama bidang duplikat, nomor ditambahkan untuk membuat nama bidang unik. Misalnya, nama atribut `h_t`, `h-t`, `hát` akan dipetakan ke `h_t`, `h_t1`, dan `h_t2` masing-masing.
  - Jika nama atribut DynamoDB melebihi 64 karakter, 56 karakter pertama dari nama atribut digabungkan dengan hash MD5 8 karakter dari nama atribut lengkap untuk membentuk nama bidang.
  - Jika nama atribut adalah `body`, itu dipetakan ke nama `f_body` bidang.
  - Jika nama atribut `_score` itu dipetakan ke nama `f__score` bidang.
- Atribut angka dipetakan ke bidang CloudSearch int Amazon dan nilainya diubah menjadi bilangan bulat 32-bit yang tidak ditandatangani:
  - Jika atribut angka berisi nilai desimal, hanya bagian integral dari nilai yang disimpan. Semuanya di sebelah kanan titik desimal dijatuhkan.
  - Jika nilainya lebih besar dari yang dapat disimpan sebagai bilangan bulat yang tidak ditandatangani, nilainya terpotong.
  - Bilangan bulat negatif diperlakukan sebagai bilangan bulat positif yang tidak ditandatangani.



## Mengunggah Data DynamoDB ke Domain melalui Konsol Amazon CloudSearch

Anda dapat menggunakan CloudSearch konsol Amazon untuk mengunggah hingga 5 MB data dari tabel DynamoDB ke domain pencarian.

Untuk mengunggah data DynamoDB menggunakan konsol

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Dari panel navigasi kiri, pilih Domain.
3. Pilih nama domain untuk membuka konfigurasi.
4. Pilih Tindakan, Unggah dokumen.
5. Pilih Amazon DynamoDB.
6. Dari dropdown, pilih tabel DynamoDB yang berisi data Anda.
  - Untuk membatasi unit kapasitas baca yang dapat dikonsumsi saat membaca dari tabel, masukkan persentase maksimum unit kapasitas baca.
  - Untuk mulai membaca dari item tertentu, tentukan kunci Mulai hash. Jika tabel menggunakan kunci primer tipe hash dan rentang, tentukan atribut hash dan atribut range untuk item tersebut.
7. Ketika Anda selesai menentukan opsi tabel, pilih Berikutnya.
8. Tinjau item yang akan diunggah. Anda juga dapat menyimpan kumpulan dokumen yang dihasilkan dengan memilih Unduh kumpulan dokumen yang dihasilkan. Kemudian pilih Unggah dokumen.

## Menyinkronkan Domain Pencarian dengan Tabel DynamoDB

Agar domain penelusuran tetap sinkron dengan pembaruan ke tabel DynamoDB, Anda dapat melacak dan menerapkan pembaruan ke domain Anda secara terprogram, atau secara berkala membuat domain baru dan mengunggah seluruh tabel lagi. Jika Anda memiliki sejumlah besar data, yang terbaik adalah melacak dan menerapkan pembaruan secara terprogram.

### Pembaruan Sinkronisasi Pemrograman

Untuk menyinkronkan perubahan dan penambahan ke tabel DynamoDB Anda, Anda dapat membuat tabel pembaruan terpisah untuk melacak perubahan pada tabel yang Anda cari dan secara berkala mengunggah konten tabel pembaruan ke domain pencarian yang sesuai.

Untuk menghapus dokumen dari domain pencarian, Anda harus membuat dan mengunggah kumpulan dokumen yang berisi operasi penghapusan untuk setiap dokumen yang dihapus. Salah satu opsi adalah menggunakan tabel DynamoDB terpisah untuk melacak item yang dihapus, memproses tabel secara berkala untuk menghasilkan sekumpulan operasi penghapusan, dan mengunggah batch ke domain pencarian Anda.

Untuk memastikan bahwa Anda tidak kehilangan perubahan apa pun yang dilakukan selama pengunggahan data awal, Anda harus mulai mengumpulkan perubahan pelacakan sebelum pengunggahan data awal. Meskipun Anda mungkin memperbarui beberapa CloudSearch dokumen Amazon dengan data yang identik, Anda memastikan bahwa tidak ada perubahan yang hilang dan domain pencarian Anda berisi up-to-date versi dari setiap dokumen.

Seberapa sering Anda menyinkronkan pembaruan tergantung pada volume perubahan dan toleransi latensi pembaruan Anda. Salah satu pendekatannya adalah mengumpulkan perubahan selama periode waktu yang tetap dan pada akhir periode waktu unggah perubahan dan hapus tabel pelacakan periode tersebut.

Misalnya, untuk menyinkronkan perubahan dan penambahan sekali sehari, pada awal setiap hari Anda dapat membuat tabel bernama `Updates_YYY_MM_DD` untuk mengumpulkan pembaruan harian. Di penghujung hari, Anda mengunggah tabel `Updates_YYY_MM_DD` ke domain pencarian Anda. Setelah unggahan selesai, Anda dapat menghapus tabel pembaruan dan membuat yang baru untuk hari berikutnya.

## Beralih ke Domain Pencarian Baru

Jika Anda tidak ingin melacak dan menerapkan pembaruan individual ke tabel Anda, Anda dapat memuat seluruh tabel secara berkala ke domain pencarian baru dan kemudian mengalihkan lalu lintas kueri Anda ke domain baru.

Untuk beralih ke domain pencarian baru

1. Buat domain pencarian baru dan salin konfigurasi dari domain yang ada.
2. Unggah seluruh tabel DynamoDB ke domain baru. Untuk informasi selengkapnya, lihat [Mengunggah Data ke Amazon CloudSearch dari DynamoDB](#).
3. Setelah domain baru aktif, perbarui entri DNS yang mengarahkan lalu lintas kueri ke domain pencarian lama untuk menunjuk ke domain baru. Misalnya, jika Anda menggunakan [Amazon Route 53](#), Anda cukup memperbarui recordset dengan endpoint layanan pencarian baru Anda.
4. Hapus domain lama.

## Memfilter Dokumen Pencocokan di Amazon CloudSearch

Anda menggunakan fq parameter untuk memfilter dokumen yang cocok dengan kriteria pencarian yang ditentukan dengan q parameter tanpa memengaruhi skor relevansi dokumen yang disertakan dalam hasil pencarian. Menentukan filter hanya mengontrol dokumen pencocokan mana yang disertakan dalam hasil, itu tidak berpengaruh pada bagaimana mereka dinilai dan diurutkan.

fqParameter mendukung sintaks kueri terstruktur yang dijelaskan dalam [Search API](#).

Misalnya, Anda dapat menambahkan available bidang ke dokumen Anda untuk menunjukkan apakah suatu item dalam stok atau tidak, dan memfilter pada bidang itu untuk membatasi hasil untuk item dalam stok:

```
search?q=star+wars&fq=available:'true'&return=title
```

## Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch

Permintaan pencarian dapat menjadi sangat intensif sumber daya untuk diproses, yang dapat berdampak pada kinerja dan biaya menjalankan domain pencarian Anda. Secara umum, penelusuran yang mengembalikan volume klik yang besar dan kueri terstruktur yang kompleks lebih intensif sumber daya daripada kueri teks sederhana yang cocok dengan persentase kecil dokumen di domain penelusuran Anda.

Jika Anda mengalami waktu respons yang lambat, sering mengalami kesalahan server internal (biasanya kesalahan 507 atau 509), atau melihat jumlah jam instance yang dikonsumsi domain pencarian Anda meningkat tanpa peningkatan substansial dalam volume data yang Anda cari, Anda dapat menyempurnakan permintaan pencarian Anda untuk membantu mengurangi overhead pemrosesan. Bagian ini mengulas apa yang harus dicari dan langkah-langkah yang dapat Anda ambil untuk menyesuaikan permintaan pencarian Anda.

## Menganalisis Latensi Kueri

Sebelum Anda dapat menyetel permintaan Anda, Anda harus menganalisis kinerja pencarian Anda saat ini. Catat permintaan pencarian dan waktu respons Anda sehingga Anda dapat melihat permintaan mana yang paling lama diproses. Penelusuran yang lambat dapat secara tidak proporsional memengaruhi kinerja keseluruhan dengan mengikat sumber daya domain pencarian Anda. Mengoptimalkan permintaan pencarian paling lambat mempercepat semua pencarian Anda.

## Topik

- [Mengurangi Jumlah Hits](#)
- [Menyederhanakan Kueri Terstruktur](#)

## Mengurangi Jumlah Hits

Latensi kueri berbanding lurus dengan jumlah dokumen yang cocok. Pencarian yang cocok dengan sebagian besar dokumen umumnya paling lambat.

Menghilangkan dua jenis pencarian yang biasanya menghasilkan sejumlah besar dokumen yang cocok dapat secara signifikan meningkatkan kinerja keseluruhan:

- Kueri yang cocok dengan setiap dokumen di corpus (`() matchAll`) Anda. Meskipun ini bisa menjadi cara yang mudah untuk membuat daftar semua dokumen di domain Anda, ini adalah kueri intensif sumber daya. Jika Anda memiliki banyak dokumen, tidak hanya dapat menyebabkan permintaan lain habis waktu, kemungkinan akan habis sendiri.
- Awalan (wildcard) mencari dengan hanya satu atau dua karakter yang ditentukan. Jika Anda menggunakan jenis pencarian ini untuk memberikan hasil instan saat pengguna mengetik, tunggu hingga pengguna memasukkan setidaknya dua karakter sebelum Anda mulai mengirimkan permintaan dan menampilkan kemungkinan kecocokan.

Untuk mengurangi jumlah dokumen yang sesuai dengan permintaan Anda, Anda juga dapat melakukan hal berikut:

- Hilangkan kata-kata yang tidak relevan dari korpus Anda sehingga tidak digunakan untuk pencocokan. Cara termudah untuk melakukannya adalah dengan menambahkannya ke kamus daftar stopwords untuk skema analisis yang Anda gunakan. Atau, Anda dapat memproses data Anda untuk menghapus kata-kata yang tidak relevan. Menghilangkan kata-kata yang tidak relevan juga memiliki manfaat mengurangi ukuran indeks Anda, yang dapat membantu mengurangi biaya.
- Secara eksplisit memfilter hasil berdasarkan nilai bidang tertentu menggunakan parameter. `fq`

Jika Anda masih memiliki permintaan yang cocok dengan banyak dokumen, Anda dapat mengurangi latensi dengan meminimalkan jumlah pemrosesan yang harus dilakukan pada set hasil:

- Minimalkan informasi aspek yang Anda minta. Menghasilkan jumlah aspek menambah waktu yang diperlukan untuk memproses permintaan dan meningkatkan kemungkinan permintaan lain akan

habis waktu. Jika Anda meminta informasi aspek, ingatlah bahwa semakin banyak aspek yang Anda tentukan, semakin lama waktu yang dibutuhkan untuk memproses permintaan.

- Hindari menggunakan ekspresi Anda sendiri untuk menyortir. Pemrosesan tambahan yang diperlukan untuk mengurutkan hasil meningkatkan kemungkinan permintaan akan habis waktu. Jika Anda harus menyesuaikan bagaimana hasil diurutkan, umumnya lebih cepat menggunakan bidang daripada menggunakan ekspresi.

Perlu diingat bahwa mengembalikan sejumlah besar data dalam hasil pencarian dapat meningkatkan waktu transportasi dan memengaruhi latensi kueri. Minimalkan jumlah bidang pengembalian yang Anda gunakan untuk meningkatkan kinerja dan mengurangi ukuran indeks Anda.

## Menyederhanakan Kueri Terstruktur

Semakin banyak klausa yang ada dalam kriteria kueri, semakin lama waktu yang dibutuhkan untuk memproses kueri.

Jika Anda memiliki kueri terstruktur kompleks yang tidak berkinerja baik, Anda perlu menemukan cara untuk mengurangi jumlah klausa. Dalam beberapa kasus, Anda mungkin hanya dapat menetapkan batas atau merumuskan ulang kueri. Di tempat lain, Anda mungkin perlu memodifikasi konfigurasi domain Anda untuk mengakomodasi kueri yang lebih sederhana.

# Meminta Domain Pencarian Anda untuk Informasi Lebih Lanjut di Amazon CloudSearch

Saat Anda mengirimkan permintaan pencarian, Amazon CloudSearch mengembalikan koleksi dokumen yang sesuai dengan kriteria penelusuran Anda. Anda juga dapat mengambil:

- Isi bidang yang dipilih
- Informasi facet yang memungkinkan Anda untuk mengkategorikan hasil
- Statistik untuk nilai yang terkandung dalam bidang numerik
- Sorotan yang menampilkan klik pencarian di data bidang
- Saran pelengkapan otomatis

## Topik

- [Mengambil Data dari Bidang Indeks di Amazon CloudSearch](#)
- [Mendapatkan Statistik untuk Bidang Numerik di Amazon CloudSearch](#)
- [Mendapatkan dan Menggunakan Informasi Facet di Amazon CloudSearch](#)
- [Menyoroti Pencarian Hits di Amazon CloudSearch](#)
- [Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch](#)

## Mengambil Data dari Bidang Indeks di Amazon CloudSearch

Secara default, hasil pencarian mencakup semua bidang yang diaktifkan kembali. Untuk mengembalikan subset bidang yang diaktifkan kembali atau mengembalikan nilai ekspresi untuk dokumen yang cocok, Anda dapat menentukan `return` parameternya. Untuk mengembalikan hanya ID dokumen untuk dokumen yang cocok, tentukan `return=_no_fields`. Untuk mengambil skor relevansi yang dihitung untuk setiap dokumen, tentukan `return=_score`. Anda menentukan beberapa bidang kembali sebagai daftar dipisahkan koma. Misalnya, `return=title,_score` mengembalikan hanya judul dan skor relevansi dari setiap dokumen yang cocok.

Hanya bidang yang dikonfigurasi untuk diaktifkan kembali yang dapat disertakan dalam hasil pencarian. Membuat field `return` diaktifkan meningkatkan ukuran indeks Anda, yang dapat meningkatkan biaya menjalankan domain Anda. Anda hanya boleh menyimpan data dokumen dalam indeks pencarian dengan membuat bidang kembali diaktifkan ketika sulit atau mahal

untuk mengambil data menggunakan cara lain. Karena perlu beberapa waktu untuk menerapkan pembaruan dokumen di seluruh domain, Anda harus mengambil data penting seperti informasi harga dengan menggunakan ID dokumen yang dikembalikan alih-alih dikembalikan dari indeks.

Misalnya, untuk menyertakan judul dan relevansi `_score` dalam hasil pencarian, tentukan yang berikut ini:

```
search?q=star -wars&return=title,_score&size=3
```

Bidang yang ditentukan disertakan dengan setiap hit di hasil pencarian:

```
{
  "status" : {
    "rid" : "y9Dzhs8oEwqMHnk=",
    "time-ms" : 2
  },
  "hits" : {
    "found" : 76,
    "start" : 0,
    "hit" : [ {
      "id" : "tt1411664",
      "fields" : {
        "title" : "Bucky Larson: Born to Be a Star",
        "_score" : "9.231539"
      }
    }, {
      "id" : "tt1911658",
      "fields" : {
        "title" : "The Penguins of Madagascar",
        "_score" : "7.1051397"
      }
    }, {
      "id" : "tt0120601",
      "fields" : {
        "title" : "Being John Malkovich",
        "_score" : "6.206055"
      }
    }
  ]
}
```

# Mendapatkan Statistik untuk Bidang Numerik di Amazon CloudSearch

Amazon CloudSearch dapat mengembalikan statistik berikut untuk bidang numerik berkemampuan aspek:

- `count`—Jumlah dokumen yang berisi nilai di bidang yang ditentukan.
- `max`—Nilai maksimum yang ditemukan di bidang yang ditentukan.
- `mean`—Rata-rata nilai yang ditemukan di bidang yang ditentukan.
- `min`—Nilai minimum yang ditemukan di bidang yang ditentukan.
- `missing`—Jumlah dokumen yang tidak mengandung nilai di bidang yang ditentukan.
- `stddev`—Ukuran untuk mengukur jumlah penyimpangan, atau variasi, dalam nilai bidang. Standar deviasi yang rendah menunjukkan bahwa nilai di semua dokumen mendekati rata-rata. Deviasi standar yang tinggi menunjukkan bahwa nilainya tersebar dalam rentang yang luas. Standar deviasi dihitung dengan mengambil akar kuadrat dari varians, yang merupakan rata-rata perbedaan kuadrat dari mean.
- `sum`—Jumlah nilai bidang di semua dokumen.
- `sumOfSquares`—Jumlah semua nilai bidang kuadrat.

Untuk mendapatkan statistik untuk bidang Anda menggunakan `stats.FIELD` parameter. `FIELD` adalah nama bidang numerik yang mendukung aspek. Anda menentukan objek JSON kosong, `stats.FIELD={}`, untuk mendapatkan semua statistik yang tersedia untuk bidang tertentu. (`stats.FIELD` parameter tidak mendukung opsi apa pun; Anda harus melewati objek JSON kosong.) Anda dapat meminta statistik untuk beberapa bidang dalam permintaan yang sama.

Anda bisa mendapatkan statistik hanya untuk bidang numerik berkemampuan aspek: `date,,, date-array double-double-array, int` atau `int-array`. Perhatikan bahwa hanya `count,, max,min,` dan `missing` statistik yang dikembalikan untuk `date` dan `date-array` bidang. Untuk informasi selengkapnya tentang mengaktifkan bidang untuk mengembalikan aspek, lihat [configure indexing options](#)

Misalnya, untuk mencari bintang dan mendapatkan statistik untuk bidang tahun, tentukan yang berikut ini:

```
search?q=star&stats.year={}
```



# Mendapatkan dan Menggunakan Informasi Facet di Amazon CloudSearch

## Topik

- [Mendapatkan Informasi Facet di Amazon CloudSearch](#)
- [Menggunakan Informasi Facet di Amazon CloudSearch](#)

Facet adalah bidang indeks yang mewakili kategori yang ingin Anda gunakan untuk menyaring dan memfilter hasil pencarian. Saat mengirimkan permintaan penelusuran ke Amazon CloudSearch, Anda dapat meminta informasi aspek untuk mengetahui berapa banyak dokumen yang memiliki nilai yang sama di bidang tertentu. Anda dapat menampilkan informasi ini bersama dengan hasil pencarian, dan menggunakannya untuk memungkinkan pengguna untuk secara interaktif menyempurnakan pencarian mereka. (Ini sering disebut sebagai navigasi segi atau pencarian segi.)

Anda bisa mendapatkan informasi aspek untuk setiap bidang yang mendukung aspek dengan menentukan `facet.FIELD` parameter dalam permintaan pencarian Anda. Secara default, Amazon CloudSearch mengembalikan jumlah facet untuk 10 nilai teratas. Untuk informasi selengkapnya tentang mengaktifkan bidang untuk mengembalikan aspek, lihat [configure indexing options](#). Untuk deskripsi `facet.FIELD` parameter, lihat [Parameter Permintaan Pencarian](#) di referensi API Pencarian.

Anda dapat menentukan opsi faset untuk mengontrol pengurutan nilai faset untuk setiap bidang, membatasi jumlah nilai faset yang dikembalikan, atau memilih nilai faset apa yang akan dihitung dan dikembalikan.

## Mendapatkan Informasi Facet di Amazon CloudSearch

Untuk mendapatkan informasi facet untuk bidang, Anda menggunakan `facet.FIELD` parameter. `FIELD` adalah nama bidang yang mendukung aspek. Anda menentukan opsi faset sebagai objek JSON. Jika objek JSON kosong (`facet.FIELD={}`), jumlah faset dihitung untuk semua nilai bidang, faset diurutkan berdasarkan jumlah faset, dan 10 aspek teratas dikembalikan dalam hasil. Anda dapat meminta informasi aspek untuk beberapa bidang dalam permintaan yang sama.

Anda dapat mengambil informasi facet dengan dua cara:

- `sort`—Mengembalikan informasi facet diurutkan baik berdasarkan jumlah faset atau nilai facet.
- `buckets`—Mengembalikan informasi facet untuk nilai atau rentang aspek tertentu.

## Menyortir Informasi Aspet

Anda menentukan `sort` opsi untuk mengontrol bagaimana informasi faset diurutkan. Ada dua opsi jenis: `count` dan `bucket`:

- Gunakan `count` untuk mengurutkan aspek berdasarkan jumlah faset. Misalnya, `facet.year={sort:'count'}` menghitung jumlah kecocokan yang memiliki nilai tahun yang sama dan mengurutkan informasi faset dengan angka itu.
- Gunakan `bucket` untuk mengurutkan aspek dengan nilai faset. Misalnya, `facet.year={sort:'bucket'}`.

Saat Anda menggunakan `sort` opsi, Anda dapat menentukan `size` opsi untuk mengontrol jumlah maksimum nilai faset yang dikembalikan dalam hasil. `size` Opsi ini hanya valid ketika Anda menggunakan `sort` opsi.

Dalam contoh berikut, informasi faset dihitung untuk `genres` bidang, genre diurutkan berdasarkan nilai faset, dan 5 genre pertama dikembalikan dalam hasil:

```
facet.genres={sort:'bucket', size:5}
```

## Informasi Aspek Bucketing

Anda dapat secara eksplisit menentukan nilai faset atau rentang yang ingin Anda hitung dengan menggunakan opsi. `buckets` Bucket ditentukan sebagai array nilai atau rentang, misalnya, `facet.color={buckets:["red","green","blue"]}`.

Untuk menentukan rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Braket persegi, `[atau]`, menunjukkan bahwa ikatan termasuk dalam rentang, penjepit keriting, `{atau}`, tidak termasuk batas. Anda dapat menghilangkan batas atas atau bawah untuk menentukan rentang terbuka. Saat menghilangkan ikatan, Anda harus menggunakan penjepit keriting. Sebagai contoh, `facet.year={buckets:["[1970,1979]","[1980,1989]","[1990,1999]","[2000,2009]","[2010,}"]}`. Untuk stempel waktu, Anda bisa menggunakannya. `q=-poet&facet.release_date={buckets:["[\'1980-01-01T00:00:00Z\'','\'1986-01-01T00:00:01Z\'"]}`

`size` Opsi `sort` dan tidak valid jika Anda menentukan ember.

Amazon CloudSearch mendukung dua metode untuk menghitung jumlah bucket, `filter` dan `interval`. Secara default, `filter` metode ini digunakan, yang hanya mengirimkan kueri `filter` tambahan untuk setiap bucket untuk mendapatkan jumlah bucket. Meskipun ini berfungsi dengan baik dalam banyak kasus, jika Anda memiliki tingkat pembaruan yang tinggi atau mengambil sejumlah besar aspek, kinerja dapat terganggu karena kueri tersebut tidak dapat memanfaatkan mekanisme caching internal.

Jika Anda mengalami kinerja kueri yang lambat untuk aspek yang disetel, coba setel metode bucket, yang `interval` mem-post-process set hasil daripada mengirimkan beberapa kueri:

```
facet.year={buckets:["[1970,1979]","[1980,1989]","[1990,1999]"],method:"interval"}
```

Sebaiknya lakukan pengujian kinerja Anda sendiri untuk menentukan metode mana yang terbaik untuk aplikasi Anda. Secara umum, `filter` metode ini lebih cepat jika Anda memiliki tingkat pembaruan yang cukup rendah dan tidak mengambil sejumlah besar bucket. Namun, jika Anda memiliki tingkat pembaruan yang tinggi atau banyak bucket, menggunakan `interval` metode ini untuk mem-post-process kumpulan hasil dapat menghasilkan kinerja kueri yang jauh lebih cepat.

## Menggunakan Informasi Facet di Amazon CloudSearch

Anda dapat menampilkan informasi aspek untuk memungkinkan pengguna untuk lebih mudah menelusuri hasil pencarian dan mengidentifikasi informasi yang mereka minati. Misalnya, jika pengguna mencoba menemukan salah satu film Star Trek, tetapi tidak dapat mengingat judul lengkapnya, ia mungkin mulai dengan mencari bintang. Jika Anda ingin menampilkan aspek teratas untuk genre, Anda akan menyertakan `facet.FIELD` dalam kueri, bersama dengan jumlah nilai faset yang ingin Anda ambil untuk setiap aspek:

```
search?q=star&facet.genres={sort:'count',size:5}&format=xml&return=_no_fields
```

Contoh sebelumnya memberi Anda informasi berikut dalam respons pencarian:

```
<results>
  <status rid="v7r9hs8oFQqMHnk=" time-ms="3"/>
  <hits found="85" start="0">
    <hit id="tt1411664"/>
    <hit id="tt1911658"/>
    <hit id="tt0086190"/>
    <hit id="tt0120601"/>
    <hit id="tt2141761"/>
    <hit id="tt1674771"/>
```

```

    <hit id="tt0056687"/>
    <hit id="tt0397892"/>
    <hit id="tt0258153"/>
    <hit id="tt0796366"/>
  </hits>
  <facets>
    <facet name="genres">
      <bucket value="Comedy" count="41"/><bucket value="Drama" count="35"/>
      <bucket value="Adventure" count="29"/>
      <bucket value="Sci-Fi" count="24"/>
      <bucket value="Action" count="20"/>
    </facet>
  </facets>
</results>

```

## Aspek Multi-Pilihan di Amazon CloudSearch

Jika Anda ingin menampilkan aspek yang tersedia dan memungkinkan pengguna memilih beberapa nilai untuk menyempurnakan hasil, Anda dapat mengirimkan satu permintaan untuk mendapatkan dokumen yang cocok dengan batasan aspek dan permintaan tambahan untuk mendapatkan jumlah faset.

Misalnya, dalam data film sampel, `genresrating`, dan `year` bidang diaktifkan faset. Jika pengguna mencari istilah penyair, Anda dapat mengirimkan permintaan berikut untuk mendapatkan film yang cocok dan jumlah aspek untuk `genres`, `rating` dan bidang: `year`

```
q=poet&facet.genres={}&facet.rating={}&facet.year={}&return=_no_fields
```

Karena tidak ada `facet.FIELD` opsi yang ditentukan, Amazon CloudSearch menghitung semua nilai faset dan mengembalikan 10 nilai teratas untuk setiap aspek:

```

{
  "status" : {
    "rid" : "it3T8tIoDgrUSvA=",
    "time-ms" : 5
  },
  "hits" : {
    "found" : 14,
    "start" : 0,
    "hit" : [
      {"id" : "tt0097165"},
      {"id" : "tt0059113"},

```

```
{ "id" : "tt0108174"},
{"id" : "tt1067765"},
{ "id" : "tt1311071"},
{"id" : "tt0810784"},
{"id" : "tt0819714"},
{"id" : "tt0203009"},
{"id" : "tt0114702"},
{"id" : "tt0107840" ]
},
"facets" : {
  "genres" : {
    "buckets" : [
      {"value" : "Drama","count" : 12},
      {"value" : "Romance","count" : 9},
      {"value" : "Biography", "count" : 4},
      {"value" : "Comedy","count" : 2},
      {"value" : "Thriller","count" : 2},
      {"value" : "War","count" : 2},
      {"value" : "Crime","count" : 1},
      {"value" : "History","count" : 1},
      {"value" : "Musical","count" : 1} ]
    },
  "rating" : {
    "buckets" : [
      {"value" : "6.3","count" : 3},
      {"value" : "6.2","count" : 2},
      {"value" : "7.1","count" : 2},
      {"value" : "7.9","count" : 2},
      {"value" : "5.3","count" : 1},
      {"value" : "6.1""count" : 1},
      {"value" : "6.4","count" : 1},
      {"value" : "6.9","count" : 1},
      {"value" : "7.6","count" : 1} ]
    },
  "year" : {
    "buckets" : [
      {"value" : "2013","count" : 3},
      {"value" : "1993","count" : 2},
      {"value" : "1965","count" : 1},
      {"value" : "1989","count" : 1},
      {"value" : "1995","count" : 1},
      {"value" : "2001","count" : 1},
      {"value" : "2004","count" : 1},
      {"value" : "2006","count" : 1},
```

```

        {"value" : "2008","count" : 1},
        {"value" : "2009","count" : 1} ]
    }
}
}

```

Saat pengguna menyempurnakan pencarian dengan memilih nilai facet, Anda menggunakan pilihan facet tersebut untuk memfilter hasil. Misalnya, jika pengguna memilih 2013, 2012, dan 1993, permintaan berikut akan merilis film yang cocok selama tahun-tahun tersebut:

```

q=poet&fq=(or year:2013 year:2012 year:1993)&facet.genres={}&facet.rating={}&facet.year={}&return=_no_fields

```

Ini mendapatkan dokumen yang cocok dengan pilihan pengguna dan faset dihitung dengan filter yang diterapkan:

```

{
  "status" : {
    "rid" : "zMP38tIoDwrUSvA=",
    "time-ms" : 6
  },
  "hits" : {
    "found" : 6,
    "start" : 0,
    "hit" : [
      {"id" : "tt0108174"},
      {"id" : "tt1067765"},
      {"id" : "tt1311071"},
      {"id" : "tt0107840"},
      {"id" : "tt1462411"},
      {"id" : "tt0455323"} ]
    },
  "facets" : {
    "genres" : {
      "buckets" : [
        {"value" : "Drama","count" : 4},
        {"value" : "Romance","count" : 3},
        {"value" : "Comedy","count" : 2},
        {"value" : "Thriller","count" : 2},
        {"value" : "Biography","count" : 1},
        {"value" : "Crime","count" : 1} ]
      }
    },
  }
}

```

```

"rating" : {
  "buckets" : [
    {"value" : "6.3", "count" : 2},
    {"value" : "5.3", "count" : 1},
    {"value" : "6.2", "count" : 1},
    {"value" : "6.4", "count" : 1},
    {"value" : "7.1", "count" : 1} ]
},
"year" : {
  "buckets" : [
    {"value" : "2013", "count" : 3},
    {"value" : "1993", "count" : 2},
    {"value" : "2012", "count" : 1} ]
}
}
}

```

Inilah yang ingin Anda tunjukkan untuk genre dan peringkat. Namun, untuk memungkinkan pengguna mengubah filter tahun, Anda perlu mendapatkan jumlah aspek untuk tahun yang tidak dipilih. Untuk melakukan ini, Anda mengirimkan permintaan kedua untuk mengambil jumlah faset untuk bidang tahun tanpa filter:

```
q=poet&facet.year={}&size=0
```

Tidak perlu mengambil dokumen yang cocok, jadi `size` parameter disetel ke nol untuk meminimalkan latensi permintaan. Permintaan hanya mengembalikan informasi facet untuk `year` bidang:

```

{
  "status" : {
    "rid" : "x/7r0NIoRwqlHfo=",
    "time-ms" : 4
  },
  "hits" : {
    "found" : 14,
    "start" : 0,
    "hit" : [ ]
  },
  "facets" : {
    "year" : {
      "buckets" : [
        {"value" : "2013", "count" : 3},

```

```
    {"value" : "1993", "count" : 2},
    {"value" : "1965", "count" : 1},
    {"value" : "1989", "count" : 1},
    {"value" : "1995", "count" : 1},
    {"value" : "2001", "count" : 1},
    {"value" : "2004", "count" : 1},
    {"value" : "2006", "count" : 1},
    {"value" : "2008", "count" : 1},
    {"value" : "2009", "count" : 1} ]
  }
}
```

Untuk meminimalkan waktu respons, Anda dapat mengirim permintaan ini secara paralel dengan permintaan untuk mendapatkan hasil yang difilter. Namun, perlu diingat bahwa permintaan tambahan ini dapat memengaruhi kinerja kueri Anda secara keseluruhan, dan mungkin perlu meningkatkan skala domain Anda untuk menangani lalu lintas tambahan. (Untuk informasi lebih lanjut tentang penskalaan, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).)

Jika pengguna lebih menyempurnakan pencarian dengan memilih genre atau peringkat, Anda menambahkannya ke kriteria filter untuk mendapatkan dokumen yang cocok. Misalnya, permintaan berikut mendapatkan film yang dirilis pada tahun 2013, 2012, atau 1993 yang memiliki peringkat 6,3:

```
q=poet&fq=(and rating:6.3 (or year:2013 year:2012
year:1993))&facet.genres={}&return=_no_fields
```

Mendapatkan informasi aspek untuk genre dalam permintaan ini mengembalikan jumlah aspek dengan filter peringkat dan tahun yang diterapkan:

```
{
  "status" : {
    "rid" : "l66b89IoEA rUSvA=",
    "time-ms" : 6
  },
  "hits" : {
    "found" : 2,
    "start" : 0,
    "hit" : [
      {"id" : "tt1462411"},
      {"id" : "tt0455323"} ]
    },
  "facets" : {
```



```
"genres" : {
  "buckets" : [
    {"value" : "Drama","count" : 2} ]
  }
}
```

Untuk memungkinkan pengguna memilih peringkat yang berbeda, Anda mengirimkan permintaan tambahan untuk mendapatkan jumlah aspek peringkat dengan hanya filter tahun yang diterapkan:

```
q=poet&fq=(or year:2013 year:2012 year:1993)&facet.rating={}&size=0
```

Permintaan ini mendapat respons berikut:

```
{
  "status" : {
    "rid" : "jqWj89IoEQrUSvA=",
    "time-ms" : 5
  },
  "hits" : {
    "found" : 6,
    "start" : 0,
    "hit" : [ ]
  },
  "facets" : {
    "rating" : {
      "buckets" : [
        {"value" : "6.3","count" : 2},
        {"value" : "5.3","count" : 1},
        {"value" : "6.2","count" : 1},
        {"value" : "6.4","count" : 1},
        {"value" : "7.1","count" : 1} ]
      }
    }
}
```

Demikian pula, Anda memerlukan permintaan lain untuk mendapatkan jumlah aspek tahun dengan hanya filter peringkat yang diterapkan:

```
q=poet&fq=rating:6.3&facet.year={}&size=0
```

Permintaan ini mendapat respons berikut:

```
{
  "status" : {
    "rid" : "4L6F8NIoDQrUSvA=",
    "time-ms" : 4
  },
  "hits" : {
    "found" : 3,
    "start" : 0,
    "hit" : [ ]
  },
  "facets" : {
    "year" : {
      "buckets" : [
        {"value" : "1995", "count" : 1},
        {"value" : "2012", "count" : 1},
        {"value" : "2013", "count" : 1} ]
    }
  }
}
```

## Menyoroti Pencarian Hits di Amazon CloudSearch

Amazon CloudSearch dapat mengembalikan kutipan dengan hasil pencarian untuk menunjukkan di mana istilah pencarian muncul dalam bidang tertentu dari dokumen yang cocok. Misalnya, dalam kutipan berikut istilah pencarian luke skywalker disorot dalam bidang: plot

```
highlights": {
  "plot": "After the rebels have been brutally overpowered by the Empire on
their newly established base, *Luke* *Skywalker* takes advanced Jedi
training with Master Yoda, while his friends are pursued by Darth Vader
as part of his plan to capture *Luke*."
}
```

Jika Anda mencari frasa, dokumen yang cocok harus berisi frasa itu. Namun, ketika Anda mengambil sorotan, istilah dalam frasa disorot satu per satu. Jika Anda mencari frasa "Luke Skywalker" dan mengambil sorotan untuk plot bidang seperti yang ditunjukkan pada contoh sebelumnya, istilah Luke ini disorot bahkan ketika tidak diikuti oleh Skywalker. Sorotan dikembalikan untuk 10 KB data pertama di bidang. Jika bidang berisi lebih dari 10 KB data dan istilah pencarian muncul melewati batas 10 KB, mereka tidak disorot.

Anda bisa mendapatkan sorotan untuk setiap bidang yang diaktifkan sorotan dengan menentukan `highlight.FIELD` parameter dalam permintaan pencarian Anda. Misalnya, untuk mendapatkan sorotan untuk `plot` bidang yang ditampilkan, Anda dapat menentukan yang berikut:

```
search?q=star wars&highlight.plot={}
```

Untuk informasi selengkapnya tentang mengaktifkan bidang untuk menampilkan sorotan, lihat [configure indexing options](#).

Anda dapat mengontrol berapa banyak kemunculan istilah pencarian dalam kutipan yang disorot, bagaimana mereka harus disorot, dan apakah kutipan dikembalikan sebagai teks biasa atau HTML. Ketika Amazon CloudSearch mengembalikan kutipan sebagai HTML, karakter non-alfanumerik diloloskan dengan pengkodean entitas HTML. Ini dilakukan untuk meminimalkan risiko yang terkait dengan penyematan konten HTML yang tidak tepercaya, karena bidang tersebut mungkin awalnya diisi dengan konten buatan pengguna.

Anda menentukan opsi sorotan sebagai objek JSON. Jika objek JSON kosong `highlight.FIELD={}`, Amazon CloudSearch menyoroti semua kemunculan istilah pencarian dengan melampirkannya dalam tag penekanan HTML, `<em>istilah</em>`, dan kutipan dikembalikan sebagai HTML.

- Untuk menentukan apakah kutipan harus dikembalikan sebagai `text` atau `html`, gunakan format opsi; misalnya, `highlight.plot={format:'text'}`
- Untuk menentukan jumlah maksimum kemunculan istilah pencarian yang ingin Anda sorot, gunakan `max_phrases` opsi; misalnya, `highlight.plot={max_phrases:3}`. Defaultnya adalah 1, maksimum adalah 5.
- Untuk menentukan string yang akan ditambahkan ke setiap istilah yang disorot, gunakan `pre_tag` opsi; misalnya, `highlight.plot={pre_tag:'<strong>', post_tag:'</strong>'}`
- Untuk menentukan string yang akan ditambahkan ke setiap istilah yang disorot, gunakan `post_tag` opsi; misalnya, `highlight.plot={pre_tag:'<strong>', post_tag:'</strong>'}`.

# Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch

Bagian ini menjelaskan cara mengonfigurasi sugester sehingga Anda dapat mengambil saran. Saran mungkin cocok untuk kueri penelusuran yang tidak lengkap—saran memungkinkan Anda menampilkan kemungkinan kecocokan sebelum pengguna selesai mengetik kueri mereka. Di Amazon CloudSearch, saran didasarkan pada isi bidang teks tertentu. Saat Anda meminta saran, Amazon CloudSearch menemukan semua dokumen yang nilainya di bidang sugester dimulai dengan string kueri yang ditentukan—awal bidang harus cocok dengan string kueri agar dianggap cocok. Data yang dikembalikan mencakup nilai bidang dan ID dokumen untuk setiap kecocokan. Anda dapat mengonfigurasi pemberi saran untuk menemukan kecocokan untuk string kueri yang tepat, atau untuk melakukan pencocokan string perkiraan (pencocokan fuzzy) untuk mengoreksi kesalahan ketik dan salah eja.

Untuk informasi selengkapnya tentang API yang disarankan, lihat [Sarankan](#) di [Search API](#).

## Topik

- [Mengkonfigurasi Suggesters untuk Amazon CloudSearch](#)
- [Mengambil Saran di Amazon CloudSearch](#)

## Mengkonfigurasi Suggesters untuk Amazon CloudSearch

Saat Anda mengonfigurasi sugester, Anda harus menentukan nama bidang teks yang ingin Anda cari kemungkinan kecocokan dan nama unik untuk pemberi saran. Bidang yang digunakan untuk saran harus diaktifkan kembali. Hanya 512 byte data pertama di lapangan yang digunakan untuk menghasilkan saran.

Nama yang disarankan harus dimulai dengan huruf dan setidaknya tiga dan panjangnya tidak lebih dari 64 karakter. Karakter yang diizinkan adalah: a-z (huruf kecil), 0-9, dan \_ (garis bawah). Nama sugester ditentukan dalam string kueri saat Anda mengambil saran, jadi yang terbaik adalah menggunakan nama pendek. Skor nama dicadangkan dan tidak dapat digunakan sebagai nama sugester.

Suggesters juga mendukung dua opsi:

- `FuzzyMatching`—Anda dapat mengatur tingkat ketidakjelasan yang diizinkan saat menyarankan kecocokan untuk string ke tidak ada, rendah, atau tinggi. Dengan `none`, string yang ditentukan

diperlakukan sebagai awalan yang tepat. Dengan rendah, saran harus berbeda dari string yang ditentukan dengan tidak lebih dari satu karakter. Dengan tinggi, saran dapat berbeda hingga dua karakter. Defaultnya tidak ada.

- **SortExpression**—Anda dapat mengonfigurasi ekspresi ini untuk menghitung skor untuk setiap saran untuk mengontrol bagaimana mereka diurutkan. Skor dibulatkan ke bilangan bulat terdekat, dengan lantai 0 dan langit-langit  $2^{31}-1$ . Skor relevansi dokumen tidak dihitung untuk saran, jadi ekspresi pengurutan tidak dapat mereferensikan nilainya. `_score` Untuk mengurutkan saran menggunakan bidang numerik atau ekspresi yang ada, cukup tentukan nama bidang atau ekspresi. Jika tidak ada ekspresi yang dikonfigurasi untuk sugester, saran diurutkan dalam urutan abjad. Perhatikan bahwa ekspresi yang didefinisikan dalam sugester tidak dapat direferensikan dalam permintaan penelusuran atau ekspresi lainnya. Jika ingin menggunakan ekspresi untuk tujuan lain, tambahkan ke konfigurasi domain Anda dan rujuk dengan nama dari sugester. Untuk informasi selengkapnya tentang ekspresi, lihat [Mengkonfigurasi Ekspresi](#).

Jika Anda ingin mendapatkan saran dari beberapa bidang teks, Anda menentukan sugester untuk setiap bidang dan mengirimkan permintaan saran terpisah untuk mendapatkan kecocokan dari setiap pemberi saran. Anda dapat mengonfigurasi hingga sepuluh sugester. Sugester dapat mengkonsumsi sejumlah besar memori dan ruang disk, terutama jika Anda menggunakan bidang sumber yang berat teks dan mengatur pencocokan fuzzy ke high.

#### Tip

Alih-alih mengonfigurasi pemberi saran untuk menggunakan semua kemungkinan dari semua dokumen, pertimbangkan untuk mengindeks 1.000 atau 10.000 kueri penelusuran paling populer dan mengonfigurasi pemberi saran untuk menggunakannya. Anda dapat menyimpan kueri dalam CloudSearch indeks Amazon terpisah atau di bidang yang hanya digunakan untuk saran.

Cara termudah untuk mendefinisikan pemberi saran adalah melalui halaman [Suggesters di konsol Amazon](#). CloudSearch Anda juga dapat menentukan sugester menggunakan AWS SDK atau AWS CLI

#### Important

Setelah Anda menambahkan sugester ke domain pencarian Anda, Anda harus menjalankan pengindeksan sebelum Anda dapat menggunakannya untuk mengambil saran. Saat Anda

menambahkan dan menghapus dokumen, Anda harus secara berkala membangun kembali indeks Anda untuk memperbarui saran. Saran tidak akan mencerminkan dokumen yang ditambahkan atau dihapus sampai Anda menelepon `IndexDocuments`.

## Mengonfigurasi Suggester melalui Konsol Amazon CloudSearch

Anda dapat dengan mudah menambahkan, memperbarui, dan menghapus suggester melalui konsol Amazon CloudSearch .

Untuk menambahkan suggester

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih Domain.
3. Pilih nama domain untuk membuka konfigurasinya.
4. Buka tab Opsi pencarian lanjutan.
5. Di panel Suggesters, pilih Add suggester .s
6. Masukkan nama untuk suggester baru.
7. Untuk bidang Sumber, tentukan bidang teks yang akan digunakan untuk saran.
8. Untuk menyertakan saran yang mengoreksi kesalahan ejaan kecil atau kesalahan ketik, atur pencocokan Fuzzy ke Rendah atau Tinggi. Saat disetel ke rendah, saran menyertakan istilah yang berbeda dari string kueri pengguna dengan satu karakter. Ketika diatur ke tinggi, saran termasuk istilah yang berbeda hingga dua karakter.
9. Untuk mengontrol cara saran diurutkan, masukkan ekspresi numerik di bidang ekspresi Sortir. Ekspresi hanya dapat berupa nama bidang numerik yang ingin Anda gunakan untuk mengurutkan saran, nama ekspresi yang ada, atau ekspresi yang valid. Untuk informasi selengkapnya tentang ekspresi, lihat [Mengkonfigurasi Ekspresi](#).
10. Klik Simpan perubahan.
11. Setelah selesai mengonfigurasi suggester untuk domain penelusuran Anda, Anda harus mengindeks ulang domain Anda sebelum dapat menggunakan pemberi saran. Untuk menjalankan pengindeksan, buka dasbor domain dan pilih Tindakan, Jalankan pengindeksan.

## Mengkonfigurasi Suggesters dengan AWS CLI

Anda dapat menambahkan atau memperbarui sugester dengan perintah `aws cloudsearch define-suggester`. Untuk menghapus sugester, Anda menggunakan `aws cloudsearch delete-suggester`.

Untuk menambah atau memperbarui sugester

- Jalankan perintah `aws cloudsearch define-suggester`. Anda menentukan konfigurasi sugester di JSON dengan opsi `--suggester`. Konfigurasi sugester harus diapit tanda kutip dan semua tanda kutip dalam konfigurasi harus diloloskan dengan garis miring terbalik. Untuk format konfigurasi sugester, lihat [define-suggester](#) di Command Reference. AWS CLI Misalnya, perintah berikut mengonfigurasi sugester yang dipanggil `mysuggester` untuk mengembalikan saran berdasarkan bidang `title`

```
aws cloudsearch define-suggester --domain-name movies --suggester "{\"SuggesterName\": \"mysuggester\", \"DocumentSuggesterOptions\": {\"SourceField\": \"title\"}}\"
{
  \"Suggester\": {
    \"Status\": {
      \"PendingDeletion\": false,
      \"State\": \"RequiresIndexDocuments\",
      \"CreationDate\": \"2014-06-26T17:26:43Z\",
      \"UpdateVersion\": 27,
      \"UpdateDate\": \"2014-06-26T17:26:43Z\"
    },
    \"Options\": {
      \"DocumentSuggesterOptions\": {
        \"SourceField\": \"title\"
      },
      \"SuggesterName\": \"mysuggester\"
    }
  }
}
```

Anda dapat menggunakan `--fuzzy-matching` opsi untuk menyertakan saran yang mengoreksi kesalahan ejaan kecil atau kesalahan ketik. Nilai yang valid untuk pencocokan fuzzy adalah `none`, `low`, dan `high` (Defaultnya adalah `none`.) Saat diatur ke `low`, saran akan menyertakan istilah yang berbeda dari string kueri pengguna dengan satu karakter. Ketika diatur ke `high`, saran akan mencakup istilah yang berbeda hingga dua karakter. Misalnya, perintah

berikut mengkonfigurasi `mysuggester` untuk menyertakan saran yang berbeda dari string kueri pengguna hanya dengan satu karakter:

```
aws cloudsearch --name mysuggester --source title
--fuzzy-matching low
```

Anda dapat menggunakan `--sort-expression` opsi untuk mengontrol bagaimana saran yang dikembalikan diurutkan. Anda dapat menggunakan ekspresi yang valid untuk menyortir. (Seringkali, ini hanya akan menjadi nama bidang numerik atau ekspresi yang telah ditentukan.) Misalnya, untuk mengurutkan saran yang dikembalikan `mysuggester` menurut nilai di `year` bidang, tentukan:

```
aws cloudsearch define-suggester --name mysuggester --source title
--fuzzy-matching low --sort-expression year
```

Untuk menghapus suggester

- Jalankan `aws cloudsearch delete-suggester` perintah dan tentukan `--name` opsi. Misalnya, untuk menghapus `mysuggester`:

```
aws cloudsearch delete-suggester --name mysuggester --delete
```

## Mengonfigurasi Suggester Menggunakan AWS SDK

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk. [DefineSuggester](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Mengambil Saran di Amazon CloudSearch

Anda mengambil saran dengan mengirimkan permintaan ke `suggest` sumber daya pada titik akhir pencarian domain melalui HTTP GET. Sebagai contoh:

```
http://search-movies-rr2f34ofg56xneuemujamut52i.us-east-1.cloudsearch.
amazonaws.com/2013-01-01/suggest?q=oce&suggester=mysuggester
```



Anda harus menentukan versi API dalam permintaan dan string kueri harus dikodekan URL. Ukuran maksimum permintaan saran adalah 8190 byte, termasuk metode HTTP, URI, dan versi protokol.

Sumber `suggest` daya mendukung empat parameter:

- `q`—String yang ingin Anda dapatkan saran.
- `suggester`—Nama sugester yang ingin Anda gunakan.
- `size`—Jumlah saran untuk diambil. Secara default, sepuluh saran teratas dikembalikan. (Saran diurutkan menurut ekspresi pengurutan yang ditentukan dalam sugester. Jika tidak ada ekspresi pengurutan yang didefinisikan dalam sugester, saran diurutkan dalam urutan abjad.)
- `format`—Jenis konten respons, `json` atau `xml`. Secara default, saran dikembalikan dalam JSON.

`suggester` parameter `q` dan harus ditentukan. Tidak ada saran yang dikembalikan jika Anda meminta saran untuk string kosong. `format` parameter `size` dan bersifat opsional.

Contoh berikut mendapat saran untuk string `oce` berdasarkan isi `title` bidang.

```
http://search-imdb2-m2brrr7ex7z6sqhgwsjdmcvud4.us-east-1.cloudsearch.amazonaws.com/2013-01-01/suggest?q=oce&suggester=title
{
  "status": {
    "rid": "646f5s0oDAr8pVk=",
    "time-ms": 2
  },
  "suggest": {
    "query": "oce",
    "found": 3,
    "suggestions": [{
      "suggestion": "Ocean's Eleven",
      "score": 0,
      "id": "tt0054135"
    },
    {
      "suggestion": "Ocean's Thirteen",
      "score": 0,
      "id": "tt0496806"
    },
    {
      "suggestion": "Ocean's Twelve",
      "score": 0,
      "id": "tt0349903"
    }
  ]
}
```

```
}  
  ]  
  }  
}
```

# Mengontrol Bagaimana Hasil Pencarian Dikembalikan di Amazon CloudSearch

Anda dapat menentukan parameter dalam permintaan pencarian untuk mengontrol bagaimana hasil pencarian diurutkan, mengembalikan hasil dalam XMLbukan JSON, dan paginasi melalui kumpulan hasil. Anda dapat menentukan ekspresi yang menghitung nilai kustom yang dapat digunakan untuk menentukan batasan pencarian atau mengurutkan hasil.

## Topik

- [Menyortir Hasil di Amazon CloudSearch](#)
- [Menggunakan Pembobotan Bidang Relatif untuk Menyesuaikan Peringkat Relevansi di Amazon CloudSearch](#)
- [Mengkonfigurasi Ekspresi di Amazon CloudSearch](#)
- [Mendapatkan Hasil sebagai XHTML di Amazon CloudSearch](#)
- [Hasil Paginasi di Amazon CloudSearch](#)

## Menyortir Hasil di Amazon CloudSearch

Secara default, hasil pencarian diurutkan sesuai dengan relevansinya dengan permintaan pencarian. Skor relevansi dokumen (`_score`) didasarkan pada seberapa sering istilah pencarian muncul dalam dokumen dibandingkan dengan seberapa umum istilah tersebut di semua dokumen dalam domain. Skor relevansi adalah nilai positif yang dapat sangat bervariasi tergantung pada data dan kueri Anda. Skor untuk setiap klausa dalam kueri Anda bersifat aditif, jadi kueri dengan lebih banyak klausa secara alami akan memiliki skor yang lebih tinggi daripada kueri hanya dengan satu atau dua. Jika Anda tahu seperti apa kueri khas Anda, Anda dapat melakukan beberapa kueri tes untuk mendapatkan gambaran tentang kisaran skor yang mungkin Anda lihat.

Untuk mengubah cara hasil pencarian diurutkan, Anda dapat:

- Gunakan `literal` bidang `text` atau untuk mengurutkan hasil menurut abjad. Perhatikan bahwa Amazon CloudSearch mengurutkan berdasarkan titik kode Unicode, jadi angka datang sebelum huruf dan huruf besar muncul sebelum huruf kecil. Angka diurutkan sebagai string, bukan berdasarkan nilai; misalnya, 10 akan datang sebelum 2.
- Gunakan `double` kolom `int` atau untuk mengurutkan hasil secara numerik.

- Gunakan `date` bidang untuk mengurutkan hasil berdasarkan tanggal.
- Gunakan ekspresi kustom untuk mengurutkan hasil.

Untuk menggunakan bidang untuk mengurutkan hasil pencarian, Anda harus mengonfigurasi bidang yang akan menjadi `SortEnabled`. Hanya bidang nilai tunggal yang dapat berupa `SortEnabled` — Anda tidak dapat menggunakan bidang tipe array untuk menyortir. Untuk informasi selengkapnya tentang mengonfigurasi bidang, lihat [configure indexing options](#).

Untuk menggunakan ekspresi untuk menyortir, Anda membuat ekspresi numerik menggunakan `int` bidang, ekspresi lain, skor relevansi dokumen, serta operator dan fungsi numerik. Anda dapat menentukan ekspresi dalam konfigurasi domain Anda, atau dalam permintaan pencarian. Untuk informasi selengkapnya tentang mengonfigurasi ekspresi, lihat [Mengkonfigurasi Ekspresi](#).

#### Tip

Untuk mengurutkan hasil secara acak, Anda dapat menggunakan `_rand` ekspresi sederhana:

```
/2013-01-01/search?expr.r=_rand&q=test&return=r%2Cplot%2Ctitle&sort=r+desc
```

Ekspresi ini stabil, yang memungkinkan Anda melakukan halaman bolak-balik tanpa kehilangan pengurutan acak awal. Jika Anda ingin menggunakan pengurutan acak yang berbeda, Anda dapat menambahkan `a-z` dan `0-9` karakter setelah `_rand` nilai, seperti:

```
/2013-01-01/search?expr.r=_rand1a2b3c&q=test&return=r%2Cplot%2Ctitle&sort=r+desc
```

Anda menggunakan `sort` parameter untuk menentukan bidang atau ekspresi yang ingin Anda gunakan untuk mengurutkan hasil. Anda harus secara eksplisit menentukan arah pengurutan bersama dengan nama bidang atau ekspresi. Misalnya, `sort=year asc` atau `sort=year desc`.

Saat Anda menggunakan bidang untuk menyortir, dokumen tanpa nilai di bidang tersebut akan dicantumkan terakhir. Jika Anda menentukan daftar kolom atau ekspresi yang dipisahkan koma, bidang atau ekspresi pertama digunakan sebagai kriteria pengurutan primer, yang kedua digunakan sebagai kriteria pengurutan sekunder, dan seterusnya.

Jika Anda tidak menentukan `sort` parameter, hasil pencarian diberi peringkat menggunakan skor relevansi default dokumen dengan dokumen dengan skor tertinggi yang tercantum terlebih dahulu. Ini setara dengan menentukan `sort=_score desc`

Anda dapat menggunakan `q.options` parameter untuk menentukan bobot bidang yang akan diterapkan saat menghitung `_score` relevansi dokumen. Untuk informasi selengkapnya, lihat [Menggunakan Pembobotan Bidang Relatif untuk Menyesuaikan Relevansi Teks](#).

## Menggunakan Pembobotan Bidang Relatif untuk Menyesuaikan Peringkat Relevansi di Amazon CloudSearch

Anda dapat menetapkan bobot ke bidang yang dipilih sehingga Anda dapat meningkatkan relevansi `_score` dokumen dengan kecocokan di bidang utama seperti `title` bidang, dan meminimalkan dampak kecocokan di bidang yang kurang penting. Secara default semua bidang memiliki berat 1.

Bobot bidang diatur dengan `q.options fields` opsi. Anda menentukan bidang sebagai array string. Untuk mengatur bobot bidang, Anda menambahkan tanda sisipan (^) dan nilai numerik positif ke nama bidang. Anda tidak dapat mengatur berat bidang ke nol atau menggunakan fungsi atau ekspresi matematika untuk menentukan berat bidang.

Misalnya, jika Anda ingin kecocokan di dalam `title` lapangan mendapat skor lebih tinggi daripada kecocokan di dalam `title` bidang, Anda dapat mengatur berat bidang menjadi 2 dan berat `plot` bidang menjadi 0,5: `plot`

```
q.options={fields:['title^2','plot^0.5']}
```

Selain mengontrol bobot bidang, `fields` opsi mendefinisikan kumpulan bidang yang dicari secara default jika Anda menggunakan parser kueri sederhana atau tidak menentukan bidang di bagian ekspresi gabungan saat menggunakan parser kueri terstruktur. Untuk informasi selengkapnya, lihat [Parameter Permintaan Pencarian](#) di Referensi API Pencarian.

Untuk mereferensikan skor relevansi tertimbang dalam definisi ekspresi, Anda menggunakan `_score`. Anda dapat menggunakan `_score` nilai tertimbang dalam hubungannya dengan bidang numerik, ekspresi lain, dan operator dan fungsi numerik standar. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Ekspresi](#).

# Mengkonfigurasi Ekspresi di Amazon CloudSearch

Anda dapat menentukan ekspresi numerik dan menggunakannya untuk mengurutkan hasil pencarian. Ekspresi juga dapat dikembalikan dalam hasil pencarian. Anda dapat menambahkan ekspresi ke konfigurasi domain atau menentukan ekspresi dalam permintaan pencarian.

## Topik

- [Menulis Ekspresi untuk Amazon CloudSearch](#)
- [Mendefinisikan CloudSearch Ekspresi Amazon dalam Permintaan Penelusuran](#)
- [Mengonfigurasi Ekspresi yang Dapat Digunakan Kembali untuk Domain Penelusuran di Amazon CloudSearch](#)
- [Membandingkan Ekspresi di Amazon CloudSearch](#)

## Menulis Ekspresi untuk Amazon CloudSearch

CloudSearch Ekspresi Amazon dapat berisi:

- Nilai tunggal, urutan bidang numerik yang diaktifkan (`int`,`double`,`date`). (Anda harus menentukan bidang tertentu, wildcard tidak didukung.)
- Ekspresi lainnya
- `_scoreVariabel`, yang mereferensikan skor relevansi dokumen
- `_timeVariabel`, yang mereferensikan waktu zaman saat ini
- `_randVariabel`, yang mengembalikan nilai yang dihasilkan secara acak
- Integer, floating point, hex, dan literal oktal
- Operator aritmatika: `+` `-` `*` `/` `%`
- Operator Bitwise: `|` `&` `^` `~` `<<` `>>` `>>>`
- Operator Boolean (termasuk operator ternary): `&&` `||` `!` `?:`
- Operator perbandingan: `<` `<=` `==` `>=` `>`
- Fungsi matematika: `abs` `ceil` `exp` `floor` `ln` `log10` `logn` `max` `min` `pow` `sqrt`
- Fungsi trigonometri: `acos` `acosh` `asin` `asinh` `atan` `atan2` `atanh` `cos` `cosh` `sin` `sinh` `tanh` `tan`
- Fungsi `haversin` jarak

[JavaScript urutan aturan prioritas berlaku](#) untuk operator. Anda dapat mengganti prioritas operator dengan menggunakan tanda kurung.

Evaluasi pintasan digunakan saat mengevaluasi ekspresi logis—jika nilai ekspresi dapat ditentukan setelah mengevaluasi argumen pertama, argumen kedua tidak dievaluasi. Misalnya, dalam ekspresi `a || b`, hanya `b` dievaluasi jika tidak `a` benar.

Ekspresi selalu mengembalikan nilai integer dari 0 ke nilai integer bertanda 64-bit maksimum ( $2^{63} - 1$ ). Hasil antara dihitung sebagai nilai floating point presisi ganda dan nilai pengembalian dibulatkan ke bilangan bulat terdekat. Jika ekspresi tidak valid atau mengevaluasi ke nilai negatif, ia mengembalikan 0. Jika ekspresi mengevaluasi nilai yang lebih besar dari maksimum, ia mengembalikan nilai maksimum.

Nama ekspresi harus dimulai dengan huruf dan setidaknya 3 dan panjangnya tidak lebih dari 64 karakter. Karakter berikut diperbolehkan: a-z (huruf kecil), 0-9, dan `_` (garis bawah). Skor nama dicadangkan dan tidak dapat digunakan sebagai nama ekspresi.

Misalnya, jika Anda menentukan `int` bidang bernama `popularity` untuk domain Anda, Anda dapat menggunakan bidang tersebut bersama dengan relevansi default `_score` untuk membuat ekspresi kustom.

```
(0.3*popularity)+(0.7*_score)
```

Perhatikan bahwa contoh sederhana ini mengasumsikan bahwa peringkat popularitas dan nilai `_score` relevansi berada dalam kisaran yang hampir sama. Untuk menyetel ekspresi Anda untuk hasil peringkat, Anda perlu melakukan beberapa pengujian untuk menentukan cara menimbang komponen ekspresi Anda untuk mendapatkan hasil yang Anda inginkan.

## Menggunakan Bidang Tanggal di CloudSearch Ekspresi Amazon

Nilai dari `date` bidang disimpan sebagai waktu epoch dengan resolusi milidetik. Ini berarti Anda dapat menggunakan operator matematika dan perbandingan untuk membangun ekspresi menggunakan tanggal yang disimpan dalam dokumen Anda dan waktu epoch saat ini (`_time`). Misalnya, menggunakan ekspresi berikut untuk mengurutkan hasil penelusuran dari domain film mendorong film dengan tanggal rilis terbaru ke bagian atas daftar.

```
_score/(_time - release_date)
```

## Mendefinisikan CloudSearch Ekspresi Amazon dalam Permintaan Penelusuran

Anda dapat menentukan dan menggunakan ekspresi secara langsung dalam permintaan pencarian sehingga Anda dapat mengulangi dengan cepat saat Anda menyempurnakan ekspresi yang Anda gunakan untuk mengurutkan hasil. Dengan mendefinisikan ekspresi dalam permintaan pencarian, Anda juga dapat memasukkan informasi kontekstual ke dalam ekspresi, seperti lokasi geografis pengguna. Anda dapat mengganti ekspresi yang ditentukan dalam konfigurasi domain dengan mendefinisikan ekspresi dengan nama yang sama dalam permintaan penelusuran.

Ketika Anda menentukan ekspresi dalam permintaan pencarian, itu tidak disimpan sebagai bagian dari konfigurasi domain Anda. Jika Anda ingin menggunakan ekspresi dalam permintaan lain, Anda harus menentukan ekspresi di setiap permintaan atau menambahkan ekspresi ke konfigurasi domain Anda. Mendefinisikan ekspresi di setiap permintaan daripada menambahkannya ke konfigurasi domain meningkatkan overhead permintaan, yang dapat mengakibatkan waktu respons lebih lambat dan berpotensi meningkatkan biaya menjalankan domain Anda. Untuk informasi tentang menambahkan ekspresi ke konfigurasi domain, lihat [Mengkonfigurasi Ekspresi](#).

Anda dapat menentukan dan menggunakan beberapa ekspresi dalam permintaan pencarian. Definisi ekspresi dapat mereferensikan ekspresi lain yang didefinisikan dalam permintaan, serta ekspresi yang dikonfigurasi sebagai bagian dari konfigurasi domain.

Tidak ada batasan tentang bagaimana Anda dapat menggunakan ekspresi yang Anda tentukan dalam permintaan pencarian. Anda dapat menggunakan ekspresi untuk mengurutkan hasil pencarian, menentukan ekspresi lain, atau mengembalikan informasi yang dihitung dalam hasil pencarian.

Untuk menentukan ekspresi dalam permintaan pencarian

1. Gunakan `expr.NAME` parameter, di mana *NAME* adalah nama ekspresi yang Anda definisikan. Sebagai contoh:

```
expr.rank1=log10(clicks)*_score
```

2. Untuk menggunakan ekspresi untuk mengurutkan hasil, tentukan nama ekspresi dengan `sort` parameter:

```
search?q=terminator&expr.rank1=log10(clicks)*_score&sort=rank1 desc
```



3. Untuk menyertakan nilai yang dihitung dalam hasil pencarian, tambahkan ekspresi ke daftar `return` bidang:

```
search?q=terminator&expr.rank1=log10(clicks)*_score&sort=rank1 desc&return=rank1
```

Misalnya, permintaan berikut membuat dua ekspresi yang digunakan untuk mengurutkan hasil dan mengembalikan salah satunya dalam hasil pencarian:

```
search?q=terminator&expr.rank1=sin( _score)&expression.rank2=cos( _score)&sort=rank1 desc,rank2 desc&return=title,_score,rank2
```

## Mengonfigurasi Ekspresi yang Dapat Digunakan Kembali untuk Domain Penelusuran di Amazon CloudSearch

Saat Anda menentukan ekspresi dalam konfigurasi domain, Anda dapat mereferensikan ekspresi dalam permintaan penelusuran apa pun. Menambahkan ekspresi ke konfigurasi domain mengurangi overhead untuk menentukannya di setiap permintaan, dan membantu memaksimalkan waktu respons dan meminimalkan biaya.

Saat Anda menambahkan ekspresi ke konfigurasi domain Anda, perlu beberapa waktu agar perubahan diproses dan ekspresi baru menjadi aktif. Untuk menguji perubahan ekspresi dengan cepat, Anda dapat menentukan dan menggunakan ekspresi secara langsung dalam permintaan pencarian, seperti yang dijelaskan dalam [query time expressions](#). Setelah Anda selesai menguji dan menyetel ekspresi, Anda harus menemukannya ke konfigurasi domain Anda.

### Topik

- [Mengonfigurasi Ekspresi Menggunakan Konsol Amazon CloudSearch](#)
- [Mengkonfigurasi CloudSearch Ekspresi Amazon Menggunakan AWS CLI](#)
- [Mengkonfigurasi Ekspresi Menggunakan API CloudSearch Konfigurasi Amazon](#)

## Mengonfigurasi Ekspresi Menggunakan Konsol Amazon CloudSearch

Untuk mengkonfigurasi ekspresi

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Dari panel navigasi kiri, pilih Domain.

3. Pilih nama domain untuk membuka konfigurasinya.
4. Buka tab Opsi pencarian lanjutan.
5. Di panel Ekspresi, pilih Tambahkan ekspresi.
6. Masukkan nama untuk ekspresi baru.
7. Untuk Nilai, masukkan ekspresi numerik yang ingin Anda evaluasi pada waktu pencarian. Anda dapat memilih Sisipkan untuk menambahkan nilai khusus dan fungsi matematika dan trigonometri.
8. Pilih Simpan.

## Mengkonfigurasi CloudSearch Ekspresi Amazon Menggunakan AWS CLI

Anda menggunakan `aws cloudsearch define-expression` perintah untuk mendefinisikan ekspresi terkomputasi untuk domain.

Untuk mengkonfigurasi ekspresi

- Jalankan `aws cloudsearch define-expression` perintah untuk mendefinisikan ekspresi baru. Anda menentukan nama untuk ekspresi dengan `--name` opsi, dan ekspresi numerik yang ingin Anda evaluasi dengan `--expression` opsi. Misalnya, permintaan berikut membuat ekspresi `popularhits` yang disebut yang memperhitungkan dokumen `popularity` dan relevansi. `_score`

```
aws cloudsearch define-expression --domain-name movies --name popularhits --
expression '((0.3*popularity)/10.0)+(0.7* _score)'
```

```
{
  "Expression": {
    "Status": {
      "PendingDeletion": false,
      "State": "Processing",
      "CreationDate": "2014-05-01T01:15:18Z",
      "UpdateVersion": 52,
      "UpdateDate": "2014-05-01T01:15:18Z"
    },
    "Options": {
      "ExpressionName": "popularhits",
      "ExpressionValue": "((0.3*popularity)/10.0)+(0.7* _score)"
    }
  }
}
```

```
}
```

## Mengkonfigurasi Ekspresi Menggunakan API CloudSearch Konfigurasi Amazon

AWS SDK (kecuali SDK Android dan iOS) mendukung semua CloudSearch tindakan Amazon yang ditentukan dalam API CloudSearch Konfigurasi Amazon, termasuk [DefineExpression](#) Untuk informasi selengkapnya tentang menginstal dan menggunakan AWS SDK, lihat [AWS Software Development Kits](#).

## Membandingkan Ekspresi di Amazon CloudSearch

Anda dapat menggunakan CloudSearch konsol Amazon untuk membandingkan ekspresi dan melihat bagaimana perubahan pada ekspresi dan bobot bidang memengaruhi cara Amazon CloudSearch mengurutkan hasil penelusuran.

Untuk membandingkan ekspresi

1. Buka CloudSearch konsol Amazon di <https://console.aws.amazon.com/cloudsearch/home>.
2. Di panel navigasi kiri, pilih Domain.
3. Pilih nama domain untuk membuka konfigurasinya.
4. Pilih Tindakan, Bandingkan ekspresi.
5. Di kotak Pencarian, masukkan istilah yang ingin Anda cari. Amazon CloudSearch memberi peringkat hasil pencarian menggunakan ekspresi dan bobot yang ditentukan. Ini menyegarkan hasil setiap kali Anda membuat perubahan pada ekspresi atau bobot.
6. Di setiap editor ekspresi, tentukan ekspresi peringkat untuk membandingkan. Anda dapat menambahkan ekspresi baru atau memilih ekspresi yang ada dari menu Ekspresi tersimpan. Amazon CloudSearch mengevaluasi ekspresi baru saat Anda mengirimkan permintaan pencarian.
7. Tentukan bobot bidang yang akan digunakan untuk setiap ekspresi. Anda juga dapat mengedit bobot bidang secara langsung dalam ekspresi. Bobot bidang harus dalam kisaran 0,0 hingga 10,0, inklusif. Secara default, bobot untuk semua bidang diatur ke 1.0. Anda dapat mengatur bobot bidang individual untuk mengontrol seberapa banyak kecocokan dalam teks tertentu atau bidang literal memengaruhi `_score` relevansi dokumen. Anda juga dapat mengubah bobot default.

 Note

Menyesuaikan bobot bidang hanya memengaruhi peringkat hasil jika ekspresi mereferensikan `_score` nilai. Anda dapat memodifikasi ekspresi untuk mengubah bagaimana relevansi bobot `_score` berkontribusi pada peringkat keseluruhan dokumen. Untuk informasi selengkapnya, lihat [Menggunakan Pembobotan Bidang Relatif untuk Menyesuaikan Relevansi Teks](#).

8. Pilih Jalankan.
9. Hasil pencarian untuk dua ekspresi ditampilkan side-by-side. (Jika ekspresi kosong, hasilnya diurutkan menurut `_score` relevansi default.) Empat ikon menyoroti perbedaan:



hijau ke atas

Panah

Dokumen diberi peringkat lebih tinggi dalam hasil pencarian menggunakan ekspresi kedua.



merah ke bawah

Panah

Dokumen diberi peringkat lebih rendah dalam hasil pencarian menggunakan ekspresi kedua.



plus

Kuning

Dokumen disertakan dalam hasil pencarian menggunakan ekspresi kedua, tetapi dihilangkan dari hasil pencarian menggunakan ekspresi pertama.



minus

Merah

Dokumen dihilangkan dari hasil pencarian menggunakan ekspresi kedua, tetapi disertakan dalam hasil pencarian menggunakan ekspresi pertama.

**Note**

Anda dapat menyimpan ekspresi ke konfigurasi domain langsung dari panel Bandingkan ekspresi. Untuk menyimpan salah satu ekspresi, pilih Simpan ekspresi.

## Mendapatkan Hasil sebagai XHTML di Amazon CloudSearch

Secara default, respons CloudSearch penelusuran Amazon diformat dalam JSON. Untuk mendapatkan hasil sebagai XHTML, tentukan parameter kueri `format=xml` dalam permintaan pencarian Anda:

```
search?q=star wars&return=_no_fields&format=xml
```

Respons penelusuran yang diformat dalam XML berisi informasi yang sama persis dengan respons JSON:

```
<results>
  <status rid="3abhhs8oEAqMHnk=" time-ms="2"/>
  <hits found="9" start="0">
    <hit id="tt0076759"/>
    <hit id="tt0086190"/>
    <hit id="tt0121766"/>
    <hit id="tt2488496"/>
    <hit id="tt1408101"/>
    <hit id="tt0489049"/>
    <hit id="tt0120915"/>
    <hit id="tt0080684"/>
    <hit id="tt0121765"/>
  </hits>
</results>
```

Untuk informasi rinci tentang format respons JSON dan XML untuk permintaan penelusuran, lihat.

[Respon Pencarian](#)

## Hasil Paginasi di Amazon CloudSearch

Secara default, Amazon CloudSearch mengembalikan sepuluh klik teratas sesuai dengan urutan pengurutan yang ditentukan. Untuk mengontrol jumlah klik yang dikembalikan dalam kumpulan hasil, Anda menggunakan `size` parameter.

Untuk mendapatkan set hit berikutnya yang dimulai dari offset tertentu, Anda dapat menggunakan `start` parameter. Perhatikan bahwa himpunan hasil berbasis nol — hasil pertama adalah pada indeks 0. Anda bisa mendapatkan 10.000 klik pertama menggunakan `start` parameter `size` dan. Untuk halaman melalui lebih dari 10.000 klik, gunakan `cursor` parameter. Untuk informasi selengkapnya, lihat [Paging Dalam Melampaui 10.000 Hits](#).

Misalnya, `search?q=wolverine` mengembalikan 10 hit pertama yang berisi wolverine, dimulai dari indeks 0. Contoh berikut menetapkan `start` parameter ke 10 untuk mendapatkan set sepuluh hit berikutnya.

```
search?q=wolverine&start=10
```

Jika Anda ingin mengambil 25 klik sekaligus, atur `size` parameternya ke 25. Untuk mendapatkan set hit pertama, Anda tidak perlu mengatur `start` parameter.

```
search?q=wolverine&size=25
```

Untuk permintaan berikutnya, gunakan `start` parameter untuk mengambil set klik yang Anda inginkan. Misalnya, untuk mendapatkan batch ketiga dari 25 hits, tentukan yang berikut ini:

```
search?q=wolverine&size=25&start=50
```

## Deep Paging Melampaui 10.000 Hits di Amazon CloudSearch

Menggunakan `size` dan `start` ke halaman melalui hasil bekerja dengan baik jika Anda hanya perlu mengakses beberapa halaman pertama hasil. Namun, jika Anda perlu halaman melalui ribuan klik, menggunakan `cursor` lebih efisien. Untuk halaman melalui lebih dari 10.000 klik, Anda harus menggunakan `filecursor`. (Anda hanya dapat mengakses 10.000 klik pertama menggunakan `size` parameter `start` dan.)

Untuk halaman melalui hasil menggunakan `cursor`, Anda menentukan `cursor=initial` dalam permintaan pencarian awal Anda dan menyertakan `size` parameter untuk menentukan berapa

banyak klik yang ingin Anda dapatkan. Amazon CloudSearch mengembalikan nilai kursor dalam respons yang Anda gunakan untuk mendapatkan set klik berikutnya. Kursor mengembalikan set hit berurutan; Namun, Anda dapat menggunakannya untuk mensimulasikan akses acak halaman dalam jika perlu. Perlu diingat bahwa kursor dimaksudkan untuk digunakan ke halaman melalui hasil yang ditetapkan dalam jumlah waktu yang wajar dari permintaan awal. Menggunakan kursor basi dapat mengembalikan hasil basi jika pembaruan telah diposting ke indeks untuk sementara.

### Important

Saat Anda menggunakan kursor ke halaman melalui kumpulan hasil yang diurutkan berdasarkan skor dokumen (`_score`), Anda bisa mendapatkan hasil yang tidak konsisten jika indeks diperbarui di antara permintaan. Ini juga dapat terjadi jika jumlah replikasi domain Anda lebih besar dari satu, karena pembaruan diterapkan secara konsisten di seluruh instance dalam domain. Jika ini menjadi masalah, hindari menyortir hasil berdasarkan skor. Anda dapat menggunakan `sort` opsi untuk mengurutkan berdasarkan bidang tertentu, atau menggunakan `fq` alih-alih `q` untuk menentukan kriteria pencarian Anda. (Skor dokumen tidak dihitung untuk kueri filter.)

Misalnya, permintaan berikut menetapkan `cursor` nilai ke `initial` dan `size` parameter `100` untuk mendapatkan set hit pertama.

```
search?q=-star&cursor=initial&size=100
```

Kursor untuk set hit berikutnya disertakan dalam respons.

```
{
  "status": {
    "rid": "z67+3L0oHgo6swY=",
    "time-ms": 7
  },
  "hits": {
    "found": 1649,
    "start": 0,
    "cursor": "Vb-HSS4YQW9JSVFKeFpvQ2wwZERBek16SXp0ems9Aw",
    "hit": [
      {
        "id": "tt0397892"
      },
      .
    ]
  }
}
```

```
    .  
    .  
    {  
      "id": "tt0332379"  
    }  
  ]  
}  
}
```

Dalam permintaan berikutnya, `cursor` parameter menentukan nilai kursor yang dikembalikan.

```
search?q=-star&cursor=Vb-HSS4YQW9JSVFKeFpvQ2wwZERBek16SXp0ems9Aw&size=100
```



# Mengintegrasikan Amazon CloudSearch dengan API Gateway

Bab ini memberikan informasi tentang mengintegrasikan Amazon CloudSearch dengan Amazon API Gateway. API Gateway memungkinkan Anda membuat dan meng-host REST API yang melakukan panggilan ke layanan lain. Beberapa kasus penggunaan untuk menggunakan API Gateway dengan Amazon CloudSearch termasuk yang berikut:

- Lebih lanjut mengamankan titik akhir CloudSearch penelusuran Amazon menggunakan kunci API atau kumpulan pengguna Amazon Cognito
- Menggunakan CloudWatch untuk memantau dan mencatat panggilan pencarian ke CloudSearch domain Amazon
- Membatasi pengguna ke subset Amazon API yang lebih terbatas CloudSearch
- Menegakkan batas tarif pada jumlah permintaan

Untuk mempelajari lebih lanjut tentang manfaat API Gateway, lihat [Panduan Pengembang API Gateway](#).

## Topik

- [Prasyarat](#)
- [Membuat dan Mengonfigurasi API \(Konsol\)](#)
- [Menguji API \(Konsol\)](#)

## Prasyarat

Sebelum mengintegrasikan Amazon CloudSearch dengan API Gateway, Anda harus memiliki sumber daya berikut.

Prasyarat	Deskripsi
CloudSearch	Untuk tujuan pengujian, domain harus memiliki beberapa data yang dapat dicari. Data film IMDb adalah pilihan yang sangat baik.
Domain Amazon	Domain harus memiliki kebijakan akses berikut:

Prasyarat	Deskripsi
	<pre data-bbox="298 212 1507 842">{   "Version": "2012-10-17",   "Statement": [     {       "Effect": "Allow",       "Principal": {         "AWS": "arn:aws:iam:: 123456789012 :role/my-api-gateway-role "       },       "Action": [         "cloudsearch:search",         "cloudsearch:suggest"       ]     }   ] }</pre> <p data-bbox="298 884 1507 1062">Kebijakan ini mengonfigurasi CloudSearch domain Amazon sehingga hanya API Gateway (dan mungkin pemilik akun) yang dapat mengaksesnya. Untuk mempelajari lebih lanjut, lihat <a href="#">the section called “Membuat Domain Pencarian”</a> dan <a href="#">Mengonfigurasi Akses untuk Amazon CloudSearch</a>.</p>

Prasyarat	Deskripsi
IAM Role	<p>Peran ini mendelegasikan izin ke API Gateway dan memungkinkannya membuat permintaan ke Amazon. CloudSearch Peran ini disebut sebagai <i>my-api-gateway-role</i> dalam Bab ini dan harus memiliki izin berikut:</p> <pre data-bbox="300 394 1507 1071">{   "Version": "2012-10-17",   "Statement": [{     "Effect": "Allow",     "Action": [       "logs:CreateLogGroup",       "logs:CreateLogStream",       "logs:DescribeLogGroups",       "logs:DescribeLogStreams",       "logs:PutLogEvents",       "logs:GetLogEvents",       "logs:FilterLogEvents"     ],     "Resource": "*"   }] }</pre> <p>Peran tersebut juga harus memiliki hubungan kepercayaan berikut:</p> <pre data-bbox="300 1182 1507 1654">{   "Version": "2012-10-17",   "Statement": [{     "Sid": "",     "Effect": "Allow",     "Principal": {       "Service": "apigateway.amazonaws.com"     },     "Action": "sts:AssumeRole"   }] }</pre> <p>Untuk mempelajari selengkapnya, lihat <a href="#">Membuat Peran</a> di Panduan Pengguna IAM.</p>

## Membuat dan Mengonfigurasi API (Konsol)

Langkah-langkah yang terlibat dalam membuat API bervariasi tergantung pada apakah permintaan menggunakan parameter, memerlukan badan permintaan, membutuhkan header tertentu, dan banyak faktor lainnya. Prosedur berikut membuat API yang memiliki satu fungsi: melakukan pencarian di CloudSearch domain Amazon. Untuk informasi selengkapnya tentang mengonfigurasi API, lihat [Membuat API di Amazon API Gateway](#).

Untuk membuat API (konsol)

1. Masuk keAWS Management Console, dan buka konsol API Gateway di <https://console.aws.amazon.com/apigateway>.
2. Pilih Buat API (atau pilih Memulai jika ini adalah pertama kalinya Anda menggunakan API Gateway).
3. Pilih Build di bawah REST API (bukan pribadi).
4. Berikan nama dan deskripsi opsional, lalu pilih Buat API.
5. Pilih Tindakan, Buat Metode. Dari menu tarik-turun, pilih GET dan konfirmasi.
6. Untuk jenis Integrasi, pilih AWSLayanan.
7. Untuk AWSWilayah, pilih Wilayah tempat CloudSearch domain Amazon Anda berada.
8. Untuk AWSLayanan, pilih CloudSearch.
9. Untuk AWSSubdomain, tentukan subdomain untuk titik akhir penelusuran CloudSearch domain Amazon Anda.

Misalnya, jika titik akhir pencarian domain `Andasearch-my-test-asdf5asdfasdfasdf5asdfg.us-west-1.cloudsearch.amazonaws.com`, tentukan `search-my-test-asdf5ambgebbgmmodhhq5asdfg`.

10. Untuk Metode HTTP, pilih GET.
11. Untuk Action Type, pilih Use path override dan enter `/2013-01-01/search`.
12. Untuk peran Eksekusi, tentukan ARN untuk `my-api-gateway-role`, seperti `arn:aws:iam::123456789012:role/my-api-gateway-role`
13. Untuk Penanganan Konten, pilih Passthrough, gunakan batas waktu default, lalu pilih Simpan.
14. Pilih Permintaan Metode.
15. Untuk Request Validator, pilih Validasi parameter string kueri dan header, lalu konfirmasi.

16. Perluas Parameter String Kueri URL. Pilih Tambahkan string kueri, beri nama stringq, dan konfirmasi. Tandai string kueri seperti yang diperlukan.
17. Pilih Metode Eksekusi untuk kembali ke ringkasan metode.
18. Pilih Permintaan Integrasi.
19. Perluas Parameter String Kueri URL. Pilih Tambahkan string kueri, beri nama stringq, berikan pemetaan `method.request.querystring.q`, lalu konfirmasi.

## Menguji API (Konsol)

Pada titik ini, Anda telah membuat API yang memiliki satu metode. Sebelum menerapkan API, Anda harus mengujinya.

Untuk menguji API (konsol)

1. Arahkan ke halaman Eksekusi Metode.
2. Pilih Uji.
3. Di bawah Query Strings, masukkan string kueri yang akan cocok dengan beberapa data di CloudSearch domain Amazon. Jika Anda menggunakan data film IMDb, coba `q=thor`.
4. Pilih Uji.
5. Verifikasi bahwa badan respons berisi hasil penelusuran, seperti berikut ini:

```
{
  "status": {
    "rid": "rcWTo8IsviEK+own",
    "time-ms": 1
  },
  "hits": {
    "found": 7,
    "start": 0,
    "hit": [
      {
        "id": "tt0800369",
        "fields": {
          "rating": "7.0",
          "genres": [
            "Action",
            "Adventure",
            "Fantasy"
          ]
        }
      }
    ]
  }
}
```

```
    ],
    "title": "Thor",
    "release_date": "2011-04-21T00:00:00Z",
    "plot": "The powerful but arrogant god Thor is cast out of Asgard to
live amongst humans in Midgard (Earth), where he soon becomes one of their finest
defenders.",
    "rank": "135",
    "running_time_secs": "6900",
    "directors": [
      "Kenneth Branagh",
      "Joss Whedon"
    ],
    "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTYxMjA5NDMzNV5BMl5BanBnXkFtZTcwOTk2Mjk3NA@@._V1_SX400_.jpg",
    "year": "2011",
    "actors": [
      "Chris Hemsworth",
      "Anthony Hopkins",
      "Natalie Portman"
    ]
  }
},
...
]
}
}
```

Pada titik ini, Anda memiliki API fungsional. Anda dapat menambahkan metode untuk mengaktifkan permintaan penelusuran yang lebih kuat, menerapkan API dan mengonfigurasi pembatasan kecepatan, membuat dan memerlukan penggunaan kunci API, menambahkan autentikasi kumpulan pengguna Amazon Cognito, dan banyak lagi. Untuk informasi selengkapnya, lihat [Panduan Pengembang API Gateway](#).

# Menangani Kesalahan di Amazon CloudSearch

Bagian ini memberikan informasi tentang cara menangani kesalahan saat berinteraksi dengan Amazon secara CloudSearch terprogram. Untuk informasi tentang kode kesalahan tertentu yang ditampilkan oleh CloudSearch layanan Amazon, lihat:

- [Kesalahan Layanan Pencarian](#)
- [dokumen/kode status batch](#)
- [Kesalahan Umum Layanan Konfigurasi](#). Untuk kesalahan spesifik yang dapat dikembalikan dari tindakan tertentu, lihat dokumentasi untuk [tindakan](#) tersebut.

## Topik

- [Jenis Kesalahan di Amazon CloudSearch](#)
- [Mencoba Kembali Permintaan di Amazon CloudSearch](#)

## Jenis Kesalahan di Amazon CloudSearch

Kode status HTTP yang dikembalikan oleh Amazon CloudSearch API menunjukkan apakah permintaan berhasil diselesaikan, atau jika terjadi kesalahan klien atau server saat memproses permintaan:

- Kode status 2xx menunjukkan bahwa permintaan klien berhasil diproses.
- Kode status 4xx menunjukkan bahwa ada masalah dengan permintaan klien. Kesalahan permintaan klien yang umum termasuk memberikan kredensi yang tidak valid dan menghilangkan parameter yang diperlukan. Ketika Anda mendapatkan kesalahan 4xx, Anda perlu memperbaiki masalah dan mengirimkan kembali permintaan klien yang dibentuk dengan benar.
- Kode status 5xx menunjukkan bahwa kesalahan server terjadi saat memproses permintaan klien. Kesalahan server biasanya bersifat sementara dan seringkali merupakan hasil dari batas waktu server, pembatasan, atau keterbatasan kapasitas. Kami merekomendasikan untuk menangkap dan mencoba kembali semua kesalahan 5xx.

Kode status HTTP dikembalikan untuk setiap permintaan. Selain itu, badan respons memberikan peringatan tambahan dan informasi kesalahan.

Pesan dalam search respons menunjukkan tingkat keparahan, peringatan atau kode kesalahan, dan deskripsi masalah dengan permintaan pencarian. Untuk daftar peringatan dan kesalahan yang dapat dikembalikan oleh layanan pencarian, lihat [Cari Properti Respon \(JSON\)](#) atau [Elemen Respon Pencarian \(XHTML\)](#).

Kesalahan dan peringatan dalam documents/batch respons memberikan informasi tentang masalah parsing dan validasi yang dihadapi saat memproses data dokumen. Untuk informasi selengkapnya, lihat [dokumen/Batch Response \(JSON\)](#) atau [dokumen/Batch Response \(XHTML\)](#).

Kesalahan yang dikembalikan dalam respons layanan konfigurasi memberikan informasi tentang apa yang menyebabkan permintaan mengembalikan kode status 4xx atau 5xx. Untuk informasi tentang kesalahan umum yang digunakan semua tindakan, lihat [Kesalahan Umum](#). Kesalahan khusus tindakan tercantum dalam topik tindakan di [Referensi API Konfigurasi untuk Amazon CloudSearch](#)

## Mencoba Kembali Permintaan di Amazon CloudSearch

Agar aplikasi Anda berjalan dengan lancar, Anda perlu membangun logika untuk menangkap dan merespons kesalahan. Salah satu pendekatan tipikal adalah mengimplementasikan permintaan Anda dalam blok coba atau pernyataan jika-maka.

Kami merekomendasikan untuk menangkap dan mencoba kembali semua kesalahan server (5xx). Karena kesalahan dapat dihasilkan dari mana saja dalam pipeline permintaan, Anda harus menerapkan fallback untuk kesalahan 5xx yang tidak terduga selain penanganan khusus untuk kode status tertentu.

Kesalahan 507 dan 509 biasanya menunjukkan bahwa layanan pencarian Anda kelebihan beban. Hal ini dapat disebabkan oleh volume atau kompleksitas permintaan pencarian yang Anda kirimkan. Amazon CloudSearch biasanya menskalakan secara otomatis untuk menangani beban. Karena membutuhkan waktu untuk menerapkan instance penelusuran tambahan, sebaiknya gunakan kebijakan coba ulang backoff eksponensial untuk mengurangi tingkat permintaan sementara dan meminimalkan kegagalan permintaan. Untuk informasi selengkapnya, lihat [Error Retries and Exponential Backoff](#).

Pola penggunaan tertentu, seperti mengirimkan kueri penelusuran yang kompleks ke satu contoh penelusuran kecil, terkadang dapat menghasilkan batas waktu tanpa memicu penskalaan otomatis. Jika Anda berulang kali mengalami tingkat kesalahan yang tinggi, Anda dapat secara eksplisit meminta kapasitas tambahan melalui formulir [Permintaan Batas CloudSearch Layanan](#) Amazon.



Kesalahan klien (4xx) biasanya menunjukkan bahwa Anda perlu merevisi permintaan untuk memperbaiki masalah — cukup mencoba kembali permintaan yang sama kemungkinan besar akan menghasilkan kesalahan yang sama. 409 kesalahan yang dikembalikan oleh layanan konfigurasi dapat menunjukkan bahwa permintaan ditolak karena batas sumber daya telah tercapai. Lihat informasi yang lebih lengkap di [Batas](#).

# Amazon CloudSearch API Referensi

Anda menggunakan tiga API untuk berinteraksi dengan Amazon CloudSearch:

- [API Konfigurasi](#) —Siapkan dan kelola domain penelusuran Anda.
- [Document Service API](#) —Kirim data yang ingin Anda cari.
- [API Pencarian](#) —Cari domain Anda.

## Referensi API Konfigurasi untuk Amazon CloudSearch

Anda menggunakan API CloudSearch Konfigurasi Amazon untuk membuat, mengonfigurasi, dan mengelola domain penelusuran. Untuk informasi selengkapnya mengonfigurasi domain penelusuran, lihat [Membuat dan Mengelola Domain Pencarian](#)

API lain yang Anda gunakan untuk berinteraksi dengan Amazon CloudSearch adalah:

- [Document Service API](#)—Kirim data yang ingin Anda cari.
- [Search API](#)—Cari domain Anda.

### Topik

- [Mengirimkan Permintaan Konfigurasi di Amazon CloudSearch](#)
- [Tindakan](#)
- [Jenis Data](#)
- [Parameter Umum](#)
- [Kesalahan Umum](#)

## Mengirimkan Permintaan Konfigurasi di Amazon CloudSearch

### Important

Cara termudah untuk mengirimkan permintaan konfigurasi adalah dengan menggunakan CloudSearch konsol Amazon, alat baris CloudSearch perintah Amazon, atau AWS SDK for Java JavaScript, .NET, PHP, Ruby, atau Python (Boto). Alat baris perintah dan SDK

menangani proses penandatanganan untuk Anda dan memastikan bahwa permintaan CloudSearch konfigurasi Amazon dibentuk dengan benar. Untuk informasi selengkapnya tentang AWS SDK, lihat [AWS Software Development Kit](#).

Anda mengirimkan permintaan CloudSearch konfigurasi Amazon ke CloudSearch titik akhir Amazon untuk wilayah Anda menggunakan protokol AWS Query. Untuk daftar wilayah dan titik akhir yang didukung saat ini, lihat [Wilayah dan Titik Akhir](#).

Permintaan AWS Query adalah permintaan HTTP atau HTTPS yang dikirimkan melalui HTTP GET atau POST dengan parameter Query bernama Action. Anda harus menentukan versi API di semua permintaan konfigurasi dan versi tersebut harus cocok dengan versi API yang ditentukan saat domain dibuat.

Anda harus menyertakan parameter otorisasi dan tanda tangan digital dalam setiap permintaan. Amazon CloudSearch mendukung AWS Signature Versi 4. Untuk petunjuk penandatanganan terperinci, lihat [Proses Penandatanganan V4 Tanda Tangan](#) di Referensi Umum AWS.

#### Note

Amazon CloudSearch membatasi permintaan berlebihan ke layanan konfigurasi. Pelambatan terjadi karena tindakan, sehingga DescribeDomains permintaan yang berlebihan tidak menyebabkan Amazon CloudSearch membatasi DescribeIndexFields permintaan. Batas permintaan berubah berdasarkan kebutuhan layanan, tetapi memungkinkan banyak panggilan ke setiap tindakan per jam.

## Struktur Permintaan Konfigurasi

Referensi ini menunjukkan permintaan CloudSearch konfigurasi Amazon sebagai URL, yang dapat digunakan langsung di browser. (Meskipun permintaan GET ditampilkan sebagai URL, nilai parameter ditampilkan tanpa kode untuk membuatnya lebih mudah dibaca. Ingatlah bahwa Anda harus menyalin URL nilai parameter saat mengirimkan permintaan.) URL berisi tiga bagian:

- Endpoint—titik masuk layanan Web untuk ditindaklanjuti, `c1oudsearch.us-east-1.amazonaws.com`
- Action—tindakan konfigurasi CloudSearch Amazon yang ingin Anda lakukan. Untuk daftar lengkap tindakan, lihat [Tindakan](#).

- Parameter—parameter permintaan apa pun yang diperlukan untuk tindakan yang ditentukan. Setiap permintaan kueri juga harus menyertakan beberapa parameter umum untuk menangani otentikasi. Untuk informasi selengkapnya, lihat [Minta Otentikasi](#).

Anda harus menentukan `Version` parameter di setiap permintaan CloudSearch konfigurasi Amazon. Versi Amazon CloudSearch API saat ini adalah 2013-01-01.

Misalnya, permintaan GET berikut membuat domain pencarian baru yang disebut film:

```
https://cloudsearch.us-east-1.amazonaws.com
?Action=CreateDomain
&DomainName=movies
&Version=2013-01-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE/20120712/us-east-1/cloudsearch/aws4
_request
&X-Amz-Date=2012-07-12T21:41:29.094Z
&X-Amz-SignedHeaders=host
&X-Amz-Signature=c7600a00fea082dac002b247f9d6812f25195fbaf7f0a6fc4ce08a39666c6a10
3c8dcb
```

## Minta Otentikasi

Permintaan yang dikirimkan ke API Konfigurasi diautentikasi menggunakan kunci akses AWS Anda. Anda harus menyertakan parameter otorisasi dan tanda tangan digital dalam setiap permintaan. Amazon CloudSearch mendukung AWS Signature Versi 4. Untuk petunjuk penandatanganan terperinci, lihat [Proses Penandatanganan V4 Tanda Tangan](#) di Referensi Umum AWS.

### Note

Jika Anda baru mulai menandatangani permintaan AWS Anda sendiri, lihat bagaimana SDK menerapkan penandatanganan. Sumber untuk sebagian besar AWS SDK tersedia di <https://github.com/aws>.

Misalnya, untuk membuat `CreateDomain` permintaan, Anda memerlukan informasi berikut:

```
Region name: us-east-1
Service name: cloudsearch
API version: 2013-01-01
```

```
Date: 2014-03-12T21:41:29.094Z
Access key: AKIAIOSFODNN7EXAMPLE
Secret key: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Action: CreateDomain
Action Parameters: DomainName=movies
```

String kueri kanonik untuk CreateDomain permintaan terlihat seperti ini:

```
Action=CreateDomain
&DomainName=movies
&Version=2013-01-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE/20120712/us-east-1/cloudsearch/aws4
_request
&X-Amz-Date=2012-07-12T21:41:29.094Z
&X-Amz-SignedHeaders=host
```

Permintaan terakhir yang ditandatangani terlihat seperti ini:

```
https://cloudsearch.us-east-1.amazonaws.com
?Action=CreateDomain
&DomainName=movies
&Version=2013-01-01
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE/20120712/us-east-1/cloudsearch/aws4
_request
&X-Amz-Date=2014-03-12T21:41:29.094Z
&X-Amz-SignedHeaders=host
&X-Amz-Signature=c7600a00fea082dac002b247f9d6812f25195fbaf7f0a6fc4ce08a39666c6a10
```

## Tindakan

Tindakan berikut didukung:

- [BuildSuggesters](#)
- [CreateDomain](#)
- [DefineAnalysisScheme](#)
- [DefineExpression](#)
- [DefineIndexField](#)
- [DefineSuggester](#)

- [DeleteAnalysisScheme](#)
- [DeleteDomain](#)
- [DeleteExpression](#)
- [DeleteIndexField](#)
- [DeleteSuggester](#)
- [DescribeAnalysisSchemes](#)
- [DescribeAvailabilityOptions](#)
- [DescribeDomains](#)
- [DescribeExpressions](#)
- [DescribeIndexFields](#)
- [DescribeScalingParameters](#)
- [DescribeServiceAccessPolicies](#)
- [DescribeSuggesters](#)
- [IndexDocuments](#)
- [ListDomainNames](#)
- [UpdateAvailabilityOptions](#)
- [UpdateScalingParameters](#)
- [UpdateServiceAccessPolicies](#)

## BuildSuggesters

### Deskripsi

Mengindeks saran pencarian. Untuk informasi selengkapnya, lihat [Mengonfigurasi Saran di Panduan Pengembang Amazon CloudSearch](#).

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `BuildSuggestersResult`.

### FieldNames

Daftar nama bidang.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

#### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

#### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## CreateDomain

### Deskripsi

Membuat domain pencarian baru. Untuk informasi selengkapnya, lihat [Membuat Domain Pencarian](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

Nama untuk domain yang Anda buat. Karakter yang diizinkan adalah a-z (huruf kecil), 0-9, dan tanda hubung (-). Nama domain harus dimulai dengan huruf atau angka dan setidaknya 3 dan tidak lebih dari 28 karakter.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `CreateDomainResult`.

### DomainStatus

Status domain pencarian saat ini.

Tipe: [DomainStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## DefineAnalysisScheme

### Deskripsi

Mengkonfigurasi skema analisis yang dapat diterapkan ke `text-array` bidang `text` atau untuk menentukan opsi pemrosesan teks khusus bahasa. Untuk informasi selengkapnya, lihat [Mengonfigurasi Skema Analisis](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### AnalysisScheme

Informasi konfigurasi untuk skema analisis. Setiap skema analisis memiliki nama yang unik dan menentukan bahasa teks yang akan diproses. Opsi berikut dapat dikonfigurasi untuk skema analisis: `Synonyms`, `Stopwords`, `StemmingDictionary`, `JapaneseTokenizationDictionary` dan `AlgorithmicStemming`.

Tipe: [AnalysisScheme](#)

Diperlukan: Ya

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DefineAnalysisSchemeResult`.

### AnalysisScheme

Status dan konfigurasi `fileAnalysisScheme`.

Tipe: [AnalysisSchemeStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DefineExpression

### Deskripsi

Mengkonfigurasi `Expression` untuk domain pencarian. Digunakan untuk membuat ekspresi baru dan memodifikasi yang sudah ada. Jika ekspresi ada, konfigurasi baru menggantikan yang lama. Untuk informasi selengkapnya, lihat [Mengonfigurasi Ekspresi](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Ekspresi

Ekspresi bernama yang dapat dievaluasi pada waktu pencarian. Dapat digunakan untuk mengurutkan hasil pencarian, menentukan ekspresi lain, atau mengembalikan informasi yang dihitung dalam hasil pencarian.

Tipe: [Ekspresi](#)

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DefineExpressionResult`.

### Ekspresi

Nilai `Expression` dan statusnya saat ini.

Tipe: [ExpressionStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DefineIndexField

### Deskripsi

Mengkonfigurasi `IndexField` untuk domain pencarian. Digunakan untuk membuat bidang baru dan memodifikasi yang sudah ada. Anda harus menentukan nama domain yang Anda konfigurasi dan konfigurasi bidang indeks. Konfigurasi bidang indeks menentukan nama unik, jenis bidang indeks, dan opsi yang ingin Anda konfigurasi untuk bidang tersebut. Opsi yang dapat Anda tentukan tergantung pada `IndexFieldType` . Jika bidang ada, konfigurasi baru menggantikan yang lama. Untuk informasi selengkapnya, lihat [Mengonfigurasi Bidang Indeks](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### IndexField

Bidang indeks dan opsi bidang yang ingin Anda konfigurasi.

Tipe: [IndexField](#)

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DefineIndexFieldResult`.

### IndexField

Nilai dari sebuah `IndexField` dan statusnya saat ini.

Tipe: [IndexFieldStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## DefineSuggester

### Deskripsi

Mengonfigurasi suggester untuk domain. Sebuah suggester memungkinkan Anda untuk menampilkan kemungkinan kecocokan sebelum pengguna selesai mengetik kueri mereka. Saat Anda mengonfigurasi suggester, Anda harus menentukan nama bidang teks yang ingin Anda cari kemungkinan kecocokan dan nama unik untuk pemberi saran. Untuk informasi selengkapnya, lihat [Mendapatkan Saran Penelusuran](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Suggester

Informasi konfigurasi untuk penelusuran penelusuran. Setiap suggester memiliki nama unik dan menentukan bidang teks yang ingin Anda gunakan untuk saran. Opsi berikut dapat dikonfigurasi untuk suggester: `FuzzyMatching`, `SortExpression`.

Tipe: [Suggester](#)

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DefineSuggesterResult`.

### Suggester

Nilai a Suggester dan statusnya saat ini.

Tipe: [SuggesterStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DeleteAnalysisScheme

### Deskripsi

Menghapus skema analisis. Untuk informasi selengkapnya, lihat [Mengonfigurasi Skema Analisis](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

#### AnalysisSchemeName

Nama skema analisis yang ingin Anda hapus.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

#### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DeleteAnalysisSchemeResult`.

#### AnalysisScheme

Status skema analisis dihapus.

Tipe: [AnalysisSchemeStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DeleteDomain

### Deskripsi

Secara permanen menghapus domain pencarian dan semua datanya. Setelah domain dihapus, itu tidak dapat dipulihkan. Untuk informasi selengkapnya, lihat [Menghapus Domain Pencarian](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

Nama domain yang ingin Anda hapus secara permanen.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DeleteDomainResult`.

### DomainStatus

Status domain pencarian saat ini.

Tipe: [DomainStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## DeleteExpression

### Deskripsi

Menghapus `Expression` dari domain pencarian. Untuk informasi selengkapnya, lihat [Mengonfigurasi Ekspresi](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### ExpressionName

Nama yang akan `Expression` dihapus.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DeleteExpressionResult`.

### Ekspresi

Status ekspresi yang dihapus.

Tipe: [ExpressionStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## DeleteIndexField

### Deskripsi

Menghapus `IndexField` dari domain pencarian. Untuk informasi selengkapnya, lihat [Mengonfigurasi Bidang Indeks](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### IndexFieldName

Nama bidang indeks yang ingin Anda hapus dari opsi pengindeksan domain.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DeleteIndexFieldResult`.

### IndexField

Status bidang indeks yang dihapus.

Tipe: [IndexFieldStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DeleteSuggester

### Deskripsi

Menghapus suggester. Untuk informasi selengkapnya, lihat [Mendapatkan Saran Penelusuran](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### SuggesterName

Menentukan nama suggester yang ingin Anda hapus.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DeleteSuggesterResult`.

### Suggester

Status suggester dihapus.

Tipe: [SuggesterStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeAnalysisSchemes

### Deskripsi

Mendapatkan skema analisis yang dikonfigurasi untuk domain. Skema analisis mendefinisikan opsi pemrosesan teks khusus bahasa untuk suatu bidang. `text` Dapat dibatasi pada skema analisis spesifik berdasarkan nama. Secara default, menampilkan semua skema analisis dan menyertakan perubahan yang tertunda pada konfigurasi. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mengonfigurasi Skema Analisis](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

#### AnalysisSchemeNames.member.n

Skema analisis yang ingin Anda gambarkan.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

#### Dikerahkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

#### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

## Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeAnalysisSchemesResult`.

### AnalysisSchemes

Deskripsi skema analisis.

Jenis: [AnalysisSchemeStatus](#) daftar

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeAvailabilityOptions

### Deskripsi

Mendapatkan opsi ketersediaan yang dikonfigurasi untuk domain. Secara default, menampilkan konfigurasi dengan perubahan yang tertunda. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Ketersediaan](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### Dikerahkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeAvailabilityOptionsResult`.

### AvailabilityOptions

Opsi ketersediaan dikonfigurasi untuk domain. Menunjukkan apakah Multi-AZ diaktifkan untuk domain.

Tipe: [AvailabilityOptionsStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

#### DisabledOperation

Permintaan ditolak karena mencoba operasi yang tidak diaktifkan.

Kode Status HTTP: 409

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

### LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## DescribeDomains

### Deskripsi

Mendapat informasi tentang domain pencarian yang dimiliki oleh akun ini. Dapat dibatasi untuk domain tertentu. Menampilkan semua domain secara default. Untuk mendapatkan jumlah dokumen yang dapat dicari di domain, gunakan konsol atau kirimkan `matchall` permintaan ke titik akhir penelusuran domain Anda: `q=matchall&q.parser=structured&size=0` Untuk informasi selengkapnya, lihat [Mendapatkan Informasi tentang Domain Pencarian](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainNames.member.n

Nama-nama domain yang ingin Anda sertakan dalam respons.

Jenis: Daftar string

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Tidak

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeDomainsResult`.

### DomainStatusList

Daftar yang berisi status setiap domain yang diminta.

Jenis: [DomainStatus](#) daftar

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

## Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

## Kode Status HTTP: 500

## DescribeExpressions

### Deskripsi

Mendapat ekspresi yang dikonfigurasi untuk domain pencarian. Dapat dibatasi pada ekspresi tertentu berdasarkan nama. Secara default, menampilkan semua ekspresi dan menyertakan perubahan yang tertunda pada konfigurasi. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mengonfigurasi Ekspresi](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### Dikerahkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### ExpressionNames.member.n

Membatasi `DescribeExpressions` respons terhadap ekspresi yang ditentukan. Jika tidak ditentukan, semua ekspresi ditampilkan.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeExpressionsResult`.

### Ekspresi

Ekspresi dikonfigurasi untuk domain.

Jenis: [ExpressionStatus](#) daftar

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeIndexFields

### Deskripsi

Mendapat informasi tentang bidang indeks yang dikonfigurasi untuk domain pencarian. Dapat dibatasi untuk bidang tertentu berdasarkan nama. Secara default, menampilkan semua bidang dan menyertakan perubahan yang tertunda pada konfigurasi. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mendapatkan Informasi Domain](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### Dikerahkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### FieldNames.member.n

Daftar bidang indeks yang ingin Anda jelaskan. Jika tidak ditentukan, informasi dikembalikan untuk semua bidang indeks yang dikonfigurasi.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeIndexFieldsResult`.

### IndexFields

Bidang indeks dikonfigurasi untuk domain.

Jenis: [IndexFieldStatus](#) daftar

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeScalingParameters

### Deskripsi

Mendapatkan parameter penskalaan yang dikonfigurasi untuk domain. Parameter penskalaan domain menentukan jenis instance pencarian dan jumlah replikasi yang diinginkan. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Penskalaan](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeScalingParametersResult`.

### ScalingParameters

Status dan konfigurasi parameter penskalaan domain pencarian.

Tipe: [ScalingParametersStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

#### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

#### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## DescribeDomainEndpointOptions

### Deskripsi

Mengembalikan opsi endpoint domain, khususnya apakah semua permintaan ke domain harus tiba melalui HTTPS. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Titik Akhir Domain](#) di Panduan CloudSearchPengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain.

Tipe: String

Diperlukan: Ya

### dikerahkan

Apakah akan mengambil konfigurasi terbaru (yang mungkin dalam Processing status) atau konfigurasi aktif saat ini (?deployed=true).

Tipe: Boolean

Wajib: Tidak

### Elemen Respons

#### DomainEndpointOptions

Status dan konfigurasi opsi endpoint domain pencarian.

Tipe: [DomainEndpointOptionsStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeServiceAccessPolicies

### Deskripsi

Mendapatkan informasi tentang kebijakan akses yang mengontrol akses ke dokumen domain dan titik akhir pencarian. Secara default, menampilkan konfigurasi dengan perubahan yang tertunda. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mengonfigurasi Akses untuk Domain Penelusuran](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### Dikembangkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeServiceAccessPoliciesResult`.

### AccessPolicies

Aturan akses yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

Tipe: [AccessPoliciesStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## DescribeSuggesters

### Deskripsi

Mendapatkan sugester yang dikonfigurasi untuk domain. Sebuah sugester memungkinkan Anda untuk menampilkan kemungkinan kecocokan sebelum pengguna selesai mengetik kueri mereka. Dapat dibatasi untuk pemberi saran tertentu berdasarkan nama. Secara default, menampilkan semua pemberi saran dan menyertakan perubahan yang tertunda pada konfigurasi. Tetapkan `Deployed` opsi `true` untuk menampilkan konfigurasi aktif dan mengecualikan perubahan yang tertunda. Untuk informasi selengkapnya, lihat [Mendapatkan Saran Penelusuran](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### Dikerahkan

Apakah akan menampilkan konfigurasi yang diterapkan (`true`) atau menyertakan perubahan yang tertunda (`false`). Default ke `false`.

Tipe: Boolean

Wajib: Tidak

### DomainName

Nama domain yang ingin Anda gambarkan.

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### SuggesterNames.member.n

Saran yang ingin Anda gambarkan.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `DescribeSuggestersResult`.

## Penyarankan

Para sugester dikonfigurasi untuk domain yang ditentukan dalam permintaan.

Jenis: [SuggesterStatus](#) daftar

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## IndexDocuments

### Deskripsi

Memberitahu domain pencarian untuk mulai mengindeks dokumennya menggunakan opsi pengindeksan terbaru. Operasi ini harus dipanggil untuk mengaktifkan opsi yang [OptionStatus](#) `adaRequiresIndexDocuments`.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `IndexDocumentsResult`.

### FieldNames

Nama-nama bidang yang saat ini sedang diindeks.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

#### Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

#### ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409



## ListDomainNames

### Deskripsi

Daftar semua domain pencarian yang dimiliki oleh akun.

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `ListDomainNamesResult`.

### DomainNames

Nama-nama domain pencarian yang dimiliki oleh akun.

Tipe: Peta String ke String

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## UpdateAvailabilityOptions

### Deskripsi

Mengkonfigurasi opsi ketersediaan untuk domain. Mengaktifkan opsi Multi-AZ memperluas CloudSearch domain Amazon ke Availability Zone tambahan di Wilayah yang sama untuk meningkatkan toleransi kesalahan jika terjadi gangguan layanan. Perubahan pada opsi Multi-AZ dapat memakan waktu sekitar setengah jam untuk menjadi aktif. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Ketersediaan](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### MultiAZ

Anda memperluas domain pencarian yang ada ke Availability Zone kedua dengan menyetel opsi Multi-AZ ke `true`. Demikian pula, Anda dapat mematikan opsi Multi-AZ untuk menurunkan versi domain ke Availability Zone tunggal dengan menyetel opsi Multi-AZ ke `false`.

Jenis: Boolean

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `UpdateAvailabilityOptionsResult`.

## AvailabilityOptions

Opsi ketersediaan yang baru dikonfigurasi. Menunjukkan apakah Multi-AZ diaktifkan untuk domain.

Tipe: [AvailabilityOptionsStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## DisabledOperation

Permintaan ditolak karena mencoba operasi yang tidak diaktifkan.

Kode Status HTTP: 409

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

## Kode Status HTTP: 409

## UpdateScalingParameters

### Deskripsi

Mengkonfigurasi parameter penskalaan untuk domain. Parameter penskalaan domain menentukan jenis instance pencarian dan jumlah replikasi yang diinginkan. Amazon CloudSearch akan tetap secara otomatis menskalakan domain Anda berdasarkan volume data dan lalu lintas, tetapi tidak di bawah jenis instans dan jumlah replikasi yang diinginkan. Jika opsi Multi-AZ diaktifkan, nilai-nilai ini mengontrol sumber daya yang digunakan per Availability Zone. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Penskalaan](#) di Panduan CloudSearch Pengembang Amazon.

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### ScalingParameters

Jenis instance yang diinginkan dan jumlah replika yang diinginkan dari setiap partisi indeks.

Tipe: [ScalingParameters](#)

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `UpdateScalingParametersResult`.

### ScalingParameters

Status dan konfigurasi parameter penskalaan domain pencarian.

Tipe: [ScalingParametersStatus](#)

## Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## UpdateDomainEndpointOptions

### Deskripsi

Memperbarui opsi endpoint domain, khususnya apakah semua permintaan ke domain harus tiba melalui HTTPS. Untuk informasi selengkapnya, lihat [Mengonfigurasi Opsi Titik Akhir Domain](#) di Panduan CloudSearchPengembang Amazon.

### Parameter Permintaan

#### DomainName

String yang mewakili nama domain.

Tipe: String

Diperlukan: Ya

#### DomainEndpointOptions

Kontainer untuk opsi titik akhir.

Tipe: [DomainEndpointOptions](#)

Diperlukan: Ya

### Elemen Respons

#### DomainEndpointOptionsStatus

Status dan konfigurasi opsi endpoint domain.

Tipe: [DomainEndpointOptionsStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

### Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## ValidationException

Permintaan berisi input yang tidak valid atau tidak ada input yang diperlukan.

Kode status HTTP 400.

## DisabledOperation

Permintaan ditolak karena mencoba operasi yang tidak diaktifkan.

Kode Status HTTP: 409



## UpdateServiceAccessPolicies

### Deskripsi

Mengkonfigurasi aturan akses yang mengontrol akses ke dokumen domain dan titik akhir pencarian. Untuk informasi selengkapnya, lihat [Mengonfigurasi Akses untuk CloudSearch Domain Amazon](#).

### Parameter Permintaan

Untuk informasi tentang parameter umum yang digunakan semua tindakan, lihat [Parameter Umum](#).

### AccessPolicies

Aturan akses yang ingin Anda konfigurasi. Aturan-aturan ini menggantikan aturan yang ada.

Tipe: String

Diperlukan: Ya

### DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

### Elemen Respons

Elemen berikut dikembalikan dalam struktur bernama `UpdateServiceAccessPoliciesResult`.

### AccessPolicies

Aturan akses yang dikonfigurasi untuk domain.

Tipe: [AccessPoliciesStatus](#)

### Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

## Basis

Terjadi kesalahan saat memproses permintaan.

Kode Status HTTP: 400

## Internal

Terjadi kesalahan internal saat memproses permintaan. Jika masalah ini berlanjut, laporkan masalah dari [Service Health Dashboard](#).

Kode Status HTTP: 500

## InvalidType

Permintaan ditolak karena ditentukan definisi tipe yang tidak valid.

Kode Status HTTP: 409

## LimitExceeded

Permintaan ditolak karena batas sumber daya telah dipenuhi.

Kode Status HTTP: 409

## ResourceNotFound

Permintaan ditolak karena berusaha mereferensikan sumber daya yang tidak ada.

Kode Status HTTP: 409

## Jenis Data

Amazon CloudSearch Configuration Service API berisi beberapa tipe data yang digunakan berbagai tindakan. Bagian ini menjelaskan setiap tipe data secara detail.

### Note

Urutan setiap elemen dalam respons tidak dijamin. Aplikasi tidak harus mengambil urutan tertentu.

Tipe data berikut didukung:

- [AccessPoliciesStatus](#)
- [AnalysisOptions](#)
- [AnalysisScheme](#)
- [AnalysisSchemeStatus](#)
- [AvailabilityOptionsStatus](#)
- [BuildSuggestersResult](#)
- [CreateDomainResult](#)
- [DateArrayOptions](#)
- [DateOptions](#)
- [DefineAnalysisSchemeResult](#)
- [DefineExpressionResult](#)
- [DefineIndexFieldResult](#)
- [DefineSuggesterResult](#)
- [DeleteAnalysisSchemeResult](#)
- [DeleteDomainResult](#)
- [DeleteExpressionResult](#)
- [DeleteIndexFieldResult](#)
- [DeleteSuggesterResult](#)
- [DescribeAnalysisSchemesResult](#)
- [DescribeAvailabilityOptionsResult](#)
- [DescribeDomainsResult](#)
- [DescribeExpressionsResult](#)
- [DescribeIndexFieldsResult](#)
- [DescribeScalingParametersResult](#)
- [DescribeServiceAccessPoliciesResult](#)
- [DescribeSuggestersResult](#)
- [DocumentSuggesterOptions](#)
- [DomainStatus](#)
- [DoubleArrayOptions](#)
- [DoubleOptions](#)

- [Ekspresi](#)
- [ExpressionStatus](#)
- [IndexDocumentsResult](#)
- [IndexField](#)
- [IndexFieldStatus](#)
- [IntArrayOptions](#)
- [IntOptions](#)
- [LatLonOptions](#)
- [Batas](#)
- [ListDomainNamesResult](#)
- [LiteralArrayOptions](#)
- [LiteralOptions](#)
- [BuildSuggestersResult](#)
- [OptionStatus](#)
- [ScalingParameters](#)
- [ScalingParametersStatus](#)
- [ServiceEndpoint](#)
- [Suggester](#)
- [SuggesterStatus](#)
- [TextArrayOptions](#)
- [TextOptions](#)
- [UpdateAvailabilityOptionsResult](#)
- [UpdateScalingParametersResult](#)
- [UpdateServiceAccessPoliciesResult](#)

## AccessPoliciesStatus

### Deskripsi

Aturan akses yang dikonfigurasi untuk dokumen domain dan titik akhir pencarian, dan status saat ini dari aturan tersebut.

## Daftar Isi

### Pilihan

Aturan akses untuk dokumen domain atau titik akhir layanan pencarian. Untuk informasi selengkapnya, lihat [Mengonfigurasi Akses untuk Domain Penelusuran](#) di Panduan CloudSearch Pengembang Amazon. Ukuran maksimum dokumen kebijakan adalah 100 KB.

Tipe: String

Diperlukan: Ya

### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## AnalysisOptions

### Deskripsi

Sinonim, stopwords, dan opsi stemming untuk skema analisis. Termasuk kamus tokenisasi untuk bahasa Jepang.

### Daftar Isi

#### AlgorithmicStemming

Tingkat pembatalan algoritmik untuk melakukan: `none`, `minimal`, `light` atau `full` Tingkat yang tersedia bervariasi tergantung pada bahasa. Untuk informasi selengkapnya, lihat [Pengaturan Pemrosesan Teks Khusus Bahasa](#) di Panduan CloudSearch Pengembang Amazon

Jenis: String

Nilai yang Valid: `none` | `minimal` | `light` | `full`

Diperlukan: Tidak

## JapaneseTokenizationDictionary

Array JSON yang berisi kumpulan istilah, token, bacaan, dan bagian pidato untuk Tokenizaiton Jepang. Kamus tokenisasi Jepang memungkinkan Anda untuk mengganti tokenisasi default untuk istilah yang dipilih. Ini hanya berlaku untuk bidang bahasa Jepang.

Tipe: String

Wajib: Tidak

## StemmingDictionary

Objek JSON yang berisi kumpulan pasangan string: nilai yang masing-masing memetakan istilah ke batangnya. Misalnya, {"term1": "stem1", "term2": "stem2", "term3": "stem3"}. Kamus stemming diterapkan selain stemming algoritmik apa pun. Ini memungkinkan Anda untuk mengesampingkan hasil pembatalan algoritmik untuk memperbaiki kasus-kasus tertentu dari overstemming atau understemming. Ukuran maksimum kamus stemming adalah 500 KB.

Tipe: String

Wajib: Tidak

## Stopwords

Sebuah array JSON istilah untuk mengabaikan selama pengindeksan dan pencarian. Misalnya, ["a", "an", "the", "of"]. Kamus stopwords harus secara eksplisit mencantumkan setiap kata yang ingin Anda abaikan. Wildcard dan ekspresi reguler tidak didukung.

Tipe: String

Wajib: Tidak

## Sinonim

Objek JSON yang mendefinisikan grup sinonim dan alias. Grup sinonim adalah array array, di mana setiap sub-array adalah sekelompok istilah di mana setiap istilah dalam grup dianggap sebagai sinonim dari setiap istilah lain dalam grup. Nilai alias adalah objek yang berisi kumpulan pasangan string:value di mana string menentukan istilah dan array nilai menentukan masing-masing alias untuk istilah tersebut. Alias dianggap sebagai sinonim dari istilah yang ditentukan, tetapi istilah tersebut tidak dianggap sebagai sinonim dari alias. Untuk informasi selengkapnya tentang menentukan sinonim, lihat [Sinonim](#) di Panduan CloudSearch Pengembang Amazon.

Tipe: String

Wajib: Tidak

## AnalysisScheme

### Deskripsi

Informasi konfigurasi untuk skema analisis. Setiap skema analisis memiliki nama yang unik dan menentukan bahasa teks yang akan diproses. Opsi berikut dapat dikonfigurasi untuk skema analisis: `Synonyms`, `Stopwords`, `StemmingDictionary`, `JapaneseTokenizationDictionary` dan `AlgorithmicStemming`.

### Daftar Isi

#### AnalysisOptions

Sinonim, stopwords, dan opsi stemming untuk skema analisis. Termasuk kamus tokenisasi untuk bahasa Jepang.

Tipe: [AnalysisOptions](#)

Diperlukan: Tidak

#### AnalysisSchemeLanguage

Kode bahasa [IETF RFC 4646](#) atau `mul` untuk beberapa bahasa.

Jenis: String

Nilai yang Valid: `ar | bg | ca | cs | da | de | el | en | es | eu | fa | fi | fr | ga | gl | he | hi | hu | hy | id | it | ja | ko | lv | mul | nl | no | pt | ro | ru | sv | th | tr | zh-Hans | zh-Hant`

Diperlukan: Ya

#### AnalysisSchemeName

Nama harus dimulai dengan huruf dan dapat berisi karakter berikut: `a-z` (huruf kecil), `0-9`, dan `_` (garis bawah).

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## AnalysisSchemeStatus

### Deskripsi

Status dan konfigurasi fileAnalysisScheme.

### Daftar Isi

### Pilihan

Informasi konfigurasi untuk skema analisis. Setiap skema analisis memiliki nama yang unik dan menentukan bahasa teks yang akan diproses. Opsi berikut dapat dikonfigurasi untuk skema analisis:Synonyms,Stopwords,StemmingDictionary,JapaneseTokenizationDictionary danAlgorithmicStemming.

Tipe: [AnalysisScheme](#)

Diperlukan: Ya

### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## AvailabilityOptionsStatus

### Deskripsi

Status dan konfigurasi opsi ketersediaan domain.

### Daftar Isi

### Pilihan

Opsi ketersediaan yang dikonfigurasi untuk domain.



Jenis: Boolean

Diperlukan: Ya

## Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## BuildSuggestersResult

### Deskripsi

Hasil BuildSuggester permintaan. Berisi daftar bidang yang digunakan untuk saran.

### Daftar Isi

#### FieldNames

Daftar nama bidang.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## CreateDomainResult

### Deskripsi

Hasil dari aCreateDomainRequest. Berisi status domain yang baru dibuat.

### Daftar Isi

#### DomainStatus

Status domain pencarian saat ini.

Tipe: [DomainStatus](#)

Diperlukan: Tidak

## DateArrayOptions

### Deskripsi

Opsi untuk bidang yang berisi larik tanggal. Hadir jika `IndexFieldType` menentukan bidang adalah `typedate-array`. Semua opsi diaktifkan secara default.

### Daftar Isi

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

#### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

#### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

#### SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

## SourceFields

Daftar bidang sumber untuk dipetakan ke bidang.

Tipe: String

Wajib: Tidak

## DateOptions

### Deskripsi

Opsi untuk bidang tanggal. Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut IETF RFC3339: T 00:00:00 Z. yyyy-mm-dd Hadir jika `IndexFieldType` menentukan bidang adalah `tipe date`. Semua opsi diaktifkan secara default.

### Daftar Isi

### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

## SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SourceField

String yang mewakili nama bidang indeks. CloudSearch mendukung bidang indeks reguler serta bidang dinamis. Nama bidang dinamis mendefinisikan pola yang dimulai atau diakhiri dengan wildcard. Bidang dokumen apa pun yang tidak dipetakan ke bidang indeks biasa tetapi cocok dengan pola bidang dinamis dikonfigurasi dengan opsi pengindeksan bidang dinamis.

Nama bidang reguler dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan \_ (garis bawah). Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (\*). Wildcard juga bisa menjadi satu-satunya karakter dalam nama bidang dinamis. Beberapa wildcard, dan wildcard yang disematkan dalam string tidak didukung.

Nama `score` dicadangkan dan tidak dapat digunakan sebagai nama bidang. Untuk merujuk ke ID dokumen, Anda dapat menggunakan nama `tersebut_id`.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## DefineAnalysisSchemeResult

### Deskripsi

Hasil dari `DefineAnalysisScheme` permintaan. Berisi status skema analisis yang baru dikonfigurasi.

## Daftar Isi

### AnalysisScheme

Status dan konfigurasi fileAnalysisScheme.

Tipe: [AnalysisSchemeStatus](#)

Diperlukan: Ya

### DefineExpressionResult

#### Deskripsi

Hasil dari DefineExpression permintaan. Berisi status ekspresi yang baru dikonfigurasi.

#### Daftar Isi

#### Ekspresi

Nilai dari sebuah Expression dan statusnya saat ini.

Tipe: [ExpressionStatus](#)

Diperlukan: Ya

### DefineIndexFieldResult

#### Deskripsi

Hasil dari DefineIndexField permintaan. Berisi status bidang indeks yang baru dikonfigurasi.

#### Daftar Isi

#### IndexField

Nilai dari sebuah IndexField dan statusnya saat ini.

Tipe: [IndexFieldStatus](#)

Diperlukan: Ya

## DefineSuggesterResult

### Deskripsi

Hasil dari DefineSuggester permintaan. Berisi status suggester yang baru dikonfigurasi.

### Daftar Isi

### Penyarankan

Nilai a Suggester dan statusnya saat ini.

Tipe: [SuggesterStatus](#)

Diperlukan: Ya

## DeleteAnalysisSchemeResult

### Deskripsi

Hasil dari DeleteAnalysisScheme permintaan. Berisi status skema analisis yang dihapus.

### Daftar Isi

### AnalysisScheme

Status skema analisis dihapus.

Tipe: [AnalysisSchemeStatus](#)

Diperlukan: Ya

## DeleteDomainResult

### Deskripsi

Hasil dari DeleteDomain permintaan. Berisi status domain yang baru dihapus, atau tidak ada status jika domain telah sepenuhnya dihapus.

### Daftar Isi

### DomainStatus

Status domain pencarian saat ini.

Tipe: [DomainStatus](#)

Diperlukan: Tidak

## DeleteExpressionResult

Deskripsi

Hasil DeleteExpression permintaan. Menentukan ekspresi yang dihapus.

Daftar Isi

Ekspresi

Status ekspresi yang dihapus.

Tipe: [ExpressionStatus](#)

Diperlukan: Ya

## DeleteIndexFieldResult

Deskripsi

Hasil DeleteIndexField permintaan.

Daftar Isi

IndexField

Status bidang indeks yang dihapus.

Tipe: [IndexFieldStatus](#)

Diperlukan: Ya

## DeleteSuggesterResult

Deskripsi

Hasil dari DeleteSuggester permintaan. Berisi status suggester yang dihapus.

## Daftar Isi

### Penyarankan

Status sugester dihapus.

Tipe: [SuggesterStatus](#)

Diperlukan: Ya

## DescribeAnalysisSchemesResult

### Deskripsi

Hasil dari `DescribeAnalysisSchemes` permintaan. Berisi skema analisis yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

### Daftar Isi

#### AnalysisSchemes

Deskripsi skema analisis.

Jenis: [AnalysisSchemeStatus](#) daftar

Wajib: Ya

## DescribeAvailabilityOptionsResult

### Deskripsi

Hasil dari `DescribeAvailabilityOptions` permintaan. Menunjukkan apakah opsi Multi-AZ diaktifkan atau tidak untuk domain yang ditentukan dalam permintaan.

### Daftar Isi

#### AvailabilityOptions

Opsi ketersediaan yang dikonfigurasi untuk domain. Menunjukkan apakah Multi-AZ diaktifkan untuk domain.

Tipe: [AvailabilityOptionsStatus](#)



Diperlukan: Tidak

## DescribeDomainsResult

### Deskripsi

Hasil dari `DescribeDomains` permintaan. Berisi status domain yang ditentukan dalam permintaan atau semua domain yang dimiliki oleh akun.

### Daftar Isi

#### DomainStatusList

Daftar yang berisi status setiap domain yang diminta.

Jenis: [DomainStatus](#) daftar

Wajib: Ya

## DescribeExpressionsResult

### Deskripsi

Hasil dari `DescribeExpressions` permintaan. Berisi ekspresi yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

### Daftar Isi

#### Ekspresi

Ekspresi dikonfigurasi untuk domain.

Jenis: [ExpressionStatus](#) daftar

Wajib: Ya

## DescribeIndexFieldsResult

### Deskripsi

Hasil dari `DescribeIndexFields` permintaan. Berisi bidang indeks yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

## Daftar Isi

### IndexFields

Bidang indeks dikonfigurasi untuk domain.

Jenis: [IndexFieldStatus](#) daftar

Wajib: Ya

## DescribeScalingParametersResult

### Deskripsi

Hasil `DescribeScalingParameters` permintaan. Berisi parameter penskalaan yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

### Daftar Isi

#### ScalingParameters

Status dan konfigurasi parameter penskalaan domain pencarian.

Tipe: [ScalingParametersStatus](#)

Diperlukan: Ya

## DescribeServiceAccessPoliciesResult

### Deskripsi

Hasil dari `DescribeServiceAccessPolicies` permintaan.

### Daftar Isi

#### AccessPolicies

Aturan akses yang dikonfigurasi untuk domain yang ditentukan dalam permintaan.

Tipe: [AccessPoliciesStatus](#)

Diperlukan: Ya

## DescribeSuggestersResult

### Deskripsi

Hasil dari DescribeSuggesters permintaan.

### Daftar Isi

### Penyarankan

Para sugester dikonfigurasi untuk domain yang ditentukan dalam permintaan.

Jenis: [SuggesterStatus](#) daftar

Wajib: Ya

## DocumentSuggesterOptions

### Deskripsi

Opsi untuk penelusuran penelusuran.

### Daftar Isi

### FuzzyMatching

Tingkat ketidakjelasan diperbolehkan saat menyarankan kecocokan untuk string:none,, low atau high Dengan none, string yang ditentukan diperlakukan sebagai awalan yang tepat. Dengan rendah, saran harus berbeda dari string yang ditentukan dengan tidak lebih dari satu karakter. Dengan tinggi, saran dapat berbeda hingga dua karakter. Defaultnya tidak ada.

Jenis: String

Nilai yang Valid: none | low | high

Diperlukan: Tidak

### SortExpression

Ekspresi yang menghitung skor untuk setiap saran untuk mengontrol bagaimana mereka diurutkan. Skor dibulatkan ke bilangan bulat terdekat, dengan lantai 0 dan langit-langit  $2^{31}-1$ . Skor relevansi dokumen tidak dihitung untuk saran, jadi ekspresi pengurutan tidak dapat

mereferensikan nilainya. `_score` Untuk mengurutkan saran menggunakan bidang numerik atau ekspresi yang ada, cukup tentukan nama bidang atau ekspresi. Jika tidak ada ekspresi yang dikonfigurasi untuk sugester, saran diurutkan dengan kecocokan terdekat yang terdaftar terlebih dahulu.

Tipe: String

Wajib: Tidak

SourceField

Nama bidang indeks yang ingin Anda gunakan untuk saran.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## DomainEndpointOptions

Deskripsi

Apakah mengharuskan semua permintaan ke domain tiba melalui HTTPS. Kami merekomendasikan `Policy-Min-TLS-1-2-2019-07` untuk `TLSecurityPolicy`. Untuk kompatibilitas dengan klien lama, defaultnya adalah `Policy-Min-TLS-1-0-2019-07`.

Daftar Isi

`enforceHttps`

Mengaktifkan atau menonaktifkan persyaratan bahwa semua permintaan ke domain tiba melalui HTTPS.

Jenis: Boolean

Nilai yang Valid: `true` | `false`

Diperlukan: Tidak

TLS SecurityPolicy

Versi TLS minimum yang diperlukan.

Jenis: String

Nilai yang Valid: Policy-Min-TLS-1-2-2019-07 | Policy-Min-TLS-1-0-2019-07

Diperlukan: Tidak

## DomainEndpointOptionsStatus

Deskripsi

Konfigurasi dan status opsi endpoint domain.

Daftar Isi

Pilihan

Konfigurasi saat ini.

Tipe: [DomainEndpointOptions](#)

Status

Status opsi konfigurasi.

Tipe: [OptionStatus](#)

## DomainStatus

Deskripsi

Status domain pencarian saat ini.

Daftar Isi

ARN

Nama Sumber Daya Amazon (ARN) dari domain pencarian. Lihat [Pengidentifikasi Entitas IAM](#) dalam Menggunakan AWS Identity and Access Management untuk informasi selengkapnya.

Tipe: String

Wajib: Tidak

## Dibuat

Benar jika domain pencarian dibuat. Diperlukan beberapa menit untuk menginisialisasi domain saat [CreateDomain](#) dipanggil. Domain pencarian yang baru dibuat dikembalikan [DescribeDomains](#) dengan nilai palsu untuk Dibuat hingga pembuatan domain selesai.

Tipe: Boolean

Wajib: Tidak

## Dihapus

Benar jika domain pencarian telah dihapus. Sistem harus membersihkan sumber daya yang didedikasikan untuk domain pencarian ketika [DeleteDomain](#) dipanggil. Domain pencarian yang baru dihapus dikembalikan [DescribeDomains](#) dengan nilai sebenarnya IsDeleted selama beberapa menit hingga pembersihan sumber daya selesai.

Tipe: Boolean

Wajib: Tidak

## DocService

Endpoint layanan untuk memperbarui dokumen dalam domain pencarian.

Tipe: [ServiceEndpoint](#)

Diperlukan: Tidak

## DomainId

Pengidentifikasi unik yang dihasilkan secara internal untuk domain.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## DomainName

String yang mewakili nama domain. Nama domain unik di seluruh domain yang dimiliki oleh akun dalam wilayah AWS. Nama domain dimulai dengan huruf atau angka dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan - (tanda hubung).

Jenis: String

Kendala panjang: Panjang minimum 3. Panjang maksimum 28.

Diperlukan: Ya

#### Batas

Tipe: [Batas](#)

Diperlukan: Tidak

#### Pengolahan

Benar jika pemrosesan sedang dilakukan untuk mengaktifkan konfigurasi domain saat ini.

Tipe: Boolean

Wajib: Tidak

#### RequiresIndexDocuments

Benar jika [IndexDocuments](#) perlu dipanggil untuk mengaktifkan konfigurasi domain saat ini.

Jenis: Boolean

Diperlukan: Ya

#### SearchInstanceCount

Jumlah instance pencarian yang tersedia untuk memproses permintaan pencarian.

Tipe: Integer

Wajib: Tidak

#### SearchInstanceType

Jenis instance yang digunakan untuk memproses permintaan pencarian.

Tipe: String

Wajib: Tidak

#### SearchPartitionCount

Jumlah partisi di mana indeks pencarian tersebar.

Tipe: Integer

Wajib: Tidak

## SearchService

Titik akhir layanan untuk meminta hasil pencarian dari domain pencarian.

Tipe: [ServiceEndpoint](#)

Diperlukan: Tidak

## DoubleArrayOptions

### Deskripsi

Opsi untuk bidang yang berisi larik nilai floating point 64-bit presisi ganda. Hadir jika `IndexFieldType` menentukan bidang adalah `jenisdouble-array`. Semua opsi diaktifkan secara default.

### Daftar Isi

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Tipe: Ganda

Diperlukan: Tidak

#### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

#### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

#### SearchEnabled

Apakah isi bidang dapat dicari.



Tipe: Boolean

Wajib: Tidak

### SourceFields

Daftar bidang sumber untuk dipetakan ke bidang.

Tipe: String

Wajib: Tidak

### DoubleOptions

#### Deskripsi

Opsi untuk bidang floating point 64-bit presisi ganda. Hadir jika `IndexFieldType` menentukan bidang adalah `tipedouble`. Semua opsi diaktifkan secara default.

#### Daftar Isi

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen. Ini bisa menjadi penting jika Anda menggunakan bidang dalam ekspresi dan bidang itu tidak ada di setiap dokumen.

Tipe: Ganda

Diperlukan: Tidak

#### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

#### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

SourceField

Nama bidang sumber untuk dipetakan ke bidang.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## Ekspresi

Deskripsi

Ekspresi bernama yang dapat dievaluasi pada waktu pencarian. Dapat digunakan untuk mengurutkan hasil pencarian, menentukan ekspresi lain, atau mengembalikan informasi yang dihitung dalam hasil pencarian.

Daftar Isi

ExpressionName

Nama harus dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan \_ (garis bawah).

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## ExpressionValue

Ekspresi untuk mengevaluasi untuk penyortiran saat memproses permintaan pencarian.

`ExpressionSintaks` didasarkan pada JavaScript ekspresi. Untuk informasi selengkapnya, lihat [Mengonfigurasi Ekspresi](#) di Panduan CloudSearch Pengembang Amazon.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum 10240.

Diperlukan: Ya

## ExpressionStatus

### Deskripsi

Nilai dari sebuah `Expression` dan statusnya saat ini.

### Daftar Isi

#### Pilihan

Ekspresi yang dievaluasi untuk menyortir saat memproses permintaan pencarian.

Tipe: [Ekspresi](#)

Diperlukan: Ya

#### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## IndexDocumentsResult

### Deskripsi

Hasil dari `IndexDocuments` permintaan. Berisi status operasi pengindeksan, termasuk bidang yang diindeks.

## Daftar Isi

### FieldNames

Nama-nama bidang yang saat ini sedang diindeks.

Jenis: Daftar string

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

### IndexField

#### Deskripsi

Informasi konfigurasi untuk bidang dalam indeks, termasuk nama, jenis, dan opsinya. Opsi yang didukung tergantung pada `IndexFieldType` .

#### Daftar Isi

### DateArrayOptions

Opsi untuk bidang yang berisi larik tanggal. Hadir jika `IndexFieldType` menentukan bidang adalah `typedate-array`. Semua opsi diaktifkan secara default.

Tipe: [DateArrayOptions](#)

Diperlukan: Tidak

### DateOptions

Opsi untuk bidang tanggal. Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut IETF RFC3339: T 00:00:00 Z. yyyy-mm-dd Hadir jika `IndexFieldType` menentukan bidang adalah `typedate`. Semua opsi diaktifkan secara default.

Tipe: [DateOptions](#)

Diperlukan: Tidak

### DoubleArrayOptions

Opsi untuk bidang yang berisi larik nilai floating point 64-bit presisi ganda. Hadir jika `IndexFieldType` menentukan bidang adalah `tipedouble-array`. Semua opsi diaktifkan secara default.

Tipe: [DoubleArrayOptions](#)

Diperlukan: Tidak

## DoubleOptions

Opsi untuk bidang floating point 64-bit presisi ganda. Hadir jika `IndexFieldType` menentukan bidang adalah `tipedouble`. Semua opsi diaktifkan secara default.

Tipe: [DoubleOptions](#)

Diperlukan: Tidak

## IndexFieldName

String yang mewakili nama bidang indeks. CloudSearch mendukung bidang indeks reguler serta bidang dinamis. Nama bidang dinamis mendefinisikan pola yang dimulai atau diakhiri dengan wildcard. Bidang dokumen apa pun yang tidak dipetakan ke bidang indeks biasa tetapi cocok dengan pola bidang dinamis dikonfigurasi dengan opsi pengindeksan bidang dinamis.

Nama bidang reguler dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan `_` (garis bawah). Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (`*`). Wildcard juga bisa menjadi satu-satunya karakter dalam nama bidang dinamis. Beberapa wildcard, dan wildcard yang disematkan dalam string tidak didukung.

Nama `score` dicadangkan dan tidak dapat digunakan sebagai nama bidang. Untuk mereferensikan ID dokumen, Anda dapat menggunakan nama tersebut `_id`.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## IndexFieldType

Jenis bidang. Opsi yang valid untuk bidang tergantung pada jenis bidang. Untuk informasi selengkapnya tentang jenis bidang yang didukung, lihat [Mengonfigurasi Bidang Indeks](#) di Panduan CloudSearch Pengembang Amazon.

Jenis: String

Nilai yang Valid: `int` | `double` | `literal` | `text` | `date` | `latlon` | `int-array` | `double-array` | `literal-array` | `text-array` | `date-array`

Diperlukan: Ya

### IntArrayOptions

Opsi untuk bidang yang berisi larik bilangan bulat bertanda 64-bit. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `int-array`. Semua opsi diaktifkan secara default.

Tipe: [IntArrayOptions](#)

Diperlukan: Tidak

### IntOptions

Opsi untuk bidang integer bertanda 64-bit. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `int`. Semua opsi diaktifkan secara default.

Tipe: [IntOptions](#)

Diperlukan: Tidak

### LatLonOptions

Opsi untuk bidang `latlon`. Bidang `latlon` berisi lokasi yang disimpan sebagai pasangan nilai lintang dan bujur. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `latlon`. Semua opsi diaktifkan secara default.

Tipe: [LatLonOptions](#)

Diperlukan: Tidak

### LiteralArrayOptions

Pilihan untuk bidang yang berisi array string literal. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `literal-array`. Semua opsi diaktifkan secara default.

Tipe: [LiteralArrayOptions](#)

Diperlukan: Tidak

### LiteralOptions

Opsi untuk bidang literal. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `literal`. Semua opsi diaktifkan secara default.

Tipe: [LiteralOptions](#)

Diperlukan: Tidak

## TextArrayOptions

Pilihan untuk bidang yang berisi array string teks. Hadir jika `IndexFieldType` menentukan bidang adalah `tipetext-array`. `text-array` Bidang selalu dapat dicari. Semua opsi diaktifkan secara default.

Tipe: [TextArrayOptions](#)

Diperlukan: Tidak

## TextOptions

Opsi untuk bidang teks. Hadir jika `IndexFieldType` menentukan bidang adalah `tipetext`. `text` Bidang selalu dapat dicari. Semua opsi diaktifkan secara default.

Tipe: [TextOptions](#)

Diperlukan: Tidak

## IndexFieldStatus

### Deskripsi

Nilai dari sebuah `IndexField` dan statusnya saat ini.

### Daftar Isi

### Pilihan

Informasi konfigurasi untuk bidang dalam indeks, termasuk nama, jenis, dan opsinya. Opsi yang didukung tergantung pada `IndexFieldType`.

Tipe: [IndexField](#)

Diperlukan: Ya

### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## IntArrayOptions

### Deskripsi

Opsi untuk bidang yang berisi larik bilangan bulat bertanda 64-bit. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `int-array`. Semua opsi diaktifkan secara default.

### Daftar Isi

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Tipe: Panjang

Diperlukan: Tidak

#### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

#### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

#### SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak



## SourceFields

Daftar bidang sumber untuk dipetakan ke bidang.

Tipe: String

Wajib: Tidak

## IntOptions

### Deskripsi

Opsi untuk bidang integer bertanda 64-bit. Hadir jika `IndexFieldType` menentukan bidang adalah jenis `int`. Semua opsi diaktifkan secara default.

### Daftar Isi

### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen. Ini bisa menjadi penting jika Anda menggunakan bidang dalam ekspresi dan bidang itu tidak ada di setiap dokumen.

Tipe: Panjang

Diperlukan: Tidak

### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

## SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SourceField

Nama bidang sumber untuk dipetakan ke bidang.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## LatLonOptions

### Deskripsi

Opsi untuk bidang latlon. Bidang latlon berisi lokasi yang disimpan sebagai pasangan nilai lintang dan bujur. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `latlon`. Semua opsi diaktifkan secara default.

### Daftar Isi

### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

SourceField

String yang mewakili nama bidang indeks. CloudSearch mendukung bidang indeks reguler serta bidang dinamis. Nama bidang dinamis mendefinisikan pola yang dimulai atau diakhiri dengan wildcard. Bidang dokumen apa pun yang tidak dipetakan ke bidang indeks biasa tetapi cocok dengan pola bidang dinamis dikonfigurasi dengan opsi pengindeksan bidang dinamis.

Nama bidang reguler dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan \_ (garis bawah). Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (\*). Wildcard juga bisa menjadi satu-satunya karakter dalam nama bidang dinamis. Beberapa wildcard, dan wildcard yang disematkan dalam string tidak didukung.

Nama `score` dicadangkan dan tidak dapat digunakan sebagai nama bidang. Untuk mereferensikan ID dokumen, Anda dapat menggunakan nama tersebut `_id`.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## Batas

### Deskripsi

Tidak ada dokumentasi tindakan yang tersedia.

### Daftar Isi

#### MaximumPartitionCount

Jenis: Integer

Diperlukan: Ya

#### MaximumReplicationCount

Jenis: Integer

Diperlukan: Ya

## ListDomainNamesResult

### Deskripsi

Hasil dari `ListDomainNames` permintaan. Berisi daftar domain yang dimiliki oleh akun.

### Daftar Isi

#### DomainNames

Nama-nama domain pencarian yang dimiliki oleh akun.

Tipe: Peta String ke String

Diperlukan: Tidak

## LiteralArrayOptions

### Deskripsi

Pilihan untuk bidang yang berisi array string literal. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `literal-array`. Semua opsi diaktifkan secara default.

### Daftar Isi

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

#### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

#### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

#### SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

## SourceFields

Daftar bidang sumber untuk dipetakan ke bidang.

Tipe: String

Wajib: Tidak

## LiteralOptions

### Deskripsi

Opsi untuk bidang literal. Hadir jika `IndexFieldType` menentukan bidang adalah tipe `literal`. Semua opsi diaktifkan secara default.

### Daftar Isi

### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

### FacetEnabled

Apakah informasi facet dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SearchEnabled

Apakah isi bidang dapat dicari.

Tipe: Boolean

Wajib: Tidak

## SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SourceField

String yang mewakili nama bidang indeks. CloudSearch mendukung bidang indeks reguler serta bidang dinamis. Nama bidang dinamis mendefinisikan pola yang dimulai atau diakhiri dengan wildcard. Bidang dokumen apa pun yang tidak dipetakan ke bidang indeks biasa tetapi cocok dengan pola bidang dinamis dikonfigurasi dengan opsi pengindeksan bidang dinamis.

Nama bidang reguler dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan \_ (garis bawah). Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (\*). Wildcard juga bisa menjadi satu-satunya karakter dalam nama bidang dinamis. Beberapa wildcard, dan wildcard yang disematkan dalam string tidak didukung.

Nama `score` dicadangkan dan tidak dapat digunakan sebagai nama bidang. Untuk merujuk ke ID dokumen, Anda dapat menggunakan nama `tersebut_id`.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## OptionStatus

### Deskripsi

Status opsi konfigurasi domain.

## Daftar Isi

### CreationDate

Stempel waktu untuk kapan opsi ini dibuat.

Jenis: DateTime

Diperlukan: Ya

### PendingDeletion

Menunjukkan bahwa opsi akan dihapus setelah pemrosesan selesai.

Tipe: Boolean

Wajib: Tidak

### Status

Keadaan memproses perubahan ke opsi. Kemungkinan nilai:

- `RequiresIndexDocuments`: nilai terbaru opsi tidak akan diterapkan sampai [IndexDocuments](#) dipanggil dan pengindeksan selesai.
- `Processing`: nilai terbaru opsi sedang dalam proses diaktifkan.
- `Active`: nilai terbaru opsi sepenuhnya diterapkan.
- `FailedToValidate`: nilai opsi tidak kompatibel dengan data domain dan tidak dapat digunakan untuk mengindeks data. Anda harus mengubah nilai opsi atau memperbarui atau menghapus dokumen yang tidak kompatibel.

Jenis: String

Nilai yang Valid: `RequiresIndexDocuments` | `Processing` | `Active` | `FailedToValidate`

Diperlukan: Ya

### UpdateDate

Stempel waktu kapan opsi ini terakhir diperbarui.

Jenis: DateTime

Diperlukan: Ya



## UpdateVersion

Sebuah integer unik yang menunjukkan kapan opsi ini terakhir diperbarui.

Tipe: Integer

Wajib: Tidak

## ScalingParameters

### Deskripsi

Jenis instance yang diinginkan dan jumlah replika yang diinginkan dari setiap partisi indeks.

### Daftar Isi

#### DesiredInstanceType

Jenis instans yang ingin Anda konfigurasi untuk domain Anda. Sebagai contoh, `search.medium`.

Jenis: String

Nilai yang Valid: `search.small` | `search.medium` | `search.large` | `search.xlarge`  
| `search.2xlarge`

#### Note

Untuk domain yang lebih lama, nilai yang valid mungkin juga mencakup `search.m1.small`, `search.m1.large`, `search.m2.xlarge`, `search.m2.2xlarge`, `search.m3.large`, `search.m3.xlarge`, dan `search.m3.2xlarge`.

Diperlukan: Tidak

#### DesiredPartitionCount

Jumlah partisi yang ingin Anda konfigurasi untuk domain Anda. Hanya berlaku ketika Anda memilih `search.2xlarge` sebagai jenis instance.

Tipe: Integer

Wajib: Tidak

## DesiredReplicationCount

Jumlah replika yang ingin Anda konfigurasi untuk setiap partisi indeks.

Tipe: Integer

Wajib: Tidak

## ScalingParametersStatus

### Deskripsi

Status dan konfigurasi parameter penskalaan domain pencarian.

### Daftar Isi

#### Pilihan

Jenis instance yang diinginkan dan jumlah replika yang diinginkan dari setiap partisi indeks.

Tipe: [ScalingParameters](#)

Diperlukan: Ya

#### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## ServiceEndpoint

### Deskripsi

Titik akhir di mana permintaan layanan dapat dikirimkan.

### Daftar Isi

#### Titik akhir

Titik akhir di mana permintaan layanan dapat dikirimkan. Misalnya, `search-imdb-movies-oopcnpjfn6ugofer3zx5iadxxca.eu-west-1.cloudsearch.amazonaws.com`

atau `doc-imdb-movies-oopcnjfn6ugofer3zx5iadxxca.eu-west-1.cloudsearch.amazonaws.com`.

Tipe: String

Wajib: Tidak

## Suggester

### Deskripsi

Informasi konfigurasi untuk penelusuran penelusuran. Setiap suggester memiliki nama unik dan menentukan bidang teks yang ingin Anda gunakan untuk saran. Opsi berikut dapat dikonfigurasi untuk suggester: `FuzzyMatching`, `SortExpression`.

### Daftar Isi

#### DocumentSuggesterOptions

Opsi untuk penelusuran penelusuran.

Tipe: [DocumentSuggesterOptions](#)

Diperlukan: Ya

#### SuggesterName

Nama harus dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan `_` (garis bawah).

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Ya

## SuggesterStatus

### Deskripsi

Nilai a Suggester dan statusnya saat ini.

## Daftar Isi

### Pilihan

Informasi konfigurasi untuk penelusuran penelusuran. Setiap sugester memiliki nama unik dan menentukan bidang teks yang ingin Anda gunakan untuk saran. Opsi berikut dapat dikonfigurasi untuk sugester: `FuzzyMatching`, `SortExpression`.

Tipe: [Suggester](#)

Diperlukan: Ya

### Status

Status opsi konfigurasi domain.

Tipe: [OptionStatus](#)

Diperlukan: Ya

## TextArrayOptions

### Deskripsi

Pilihan untuk bidang yang berisi array string teks. Hadir jika `IndexFieldType` menentukan bidang adalah `text-array`. `text-array` bidang selalu dapat dicari. Semua opsi diaktifkan secara default.

### Daftar Isi

#### AnalysisScheme

Nama skema analisis untuk suatu `text-array` bidang.

Tipe: String

Wajib: Tidak

#### DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

### HighlightEnabled

Apakah sorotan dapat dikembalikan untuk lapangan.

Tipe: Boolean

Wajib: Tidak

### ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

### SourceFields

Daftar bidang sumber untuk dipetakan ke bidang.

Tipe: String

Wajib: Tidak

## TextOptions

### Deskripsi

Opsi untuk bidang teks. Hadir jika `IndexFieldType` menentukan bidang adalah `text`. `textBidang` selalu dapat dicari. Semua opsi diaktifkan secara default.

### Daftar Isi

### AnalysisScheme

Nama skema analisis untuk suatu `text` bidang.

Tipe: String

Wajib: Tidak

## DefaultValue

Nilai yang akan digunakan untuk bidang jika bidang tidak ditentukan untuk dokumen.

Jenis: String

Kendala panjang: Panjang minimum 0. Panjang maksimum 1024.

Diperlukan: Tidak

## HighlightEnabled

Apakah sorotan dapat dikembalikan untuk bidang tersebut.

Tipe: Boolean

Wajib: Tidak

## ReturnEnabled

Apakah isi bidang dapat dikembalikan dalam hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SortEnabled

Apakah bidang dapat digunakan untuk mengurutkan hasil pencarian.

Tipe: Boolean

Wajib: Tidak

## SourceField

String yang mewakili nama bidang indeks. CloudSearch mendukung bidang indeks reguler serta bidang dinamis. Nama bidang dinamis mendefinisikan pola yang dimulai atau diakhiri dengan wildcard. Bidang dokumen apa pun yang tidak dipetakan ke bidang indeks biasa tetapi cocok dengan pola bidang dinamis dikonfigurasi dengan opsi pengindeksan bidang dinamis.

Nama bidang reguler dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan \_ (garis bawah). Nama bidang dinamis harus dimulai atau diakhiri dengan wildcard (\*). Wildcard juga bisa menjadi satu-satunya karakter dalam nama bidang dinamis. Beberapa wildcard, dan wildcard yang disematkan dalam string tidak didukung.

Nama `score` dicadangkan dan tidak dapat digunakan sebagai nama bidang. Untuk mereferensikan ID dokumen, Anda dapat menggunakan nama tersebut `_id`.

Jenis: String

Panjang batasan: Panjang minimum 1. Panjang maksimum adalah 64.

Diperlukan: Tidak

## UpdateAvailabilityOptionsResult

### Deskripsi

Hasil dari `UpdateAvailabilityOptions` permintaan. Berisi status opsi ketersediaan domain.

### Daftar Isi

#### AvailabilityOptions

Opsi ketersediaan yang baru dikonfigurasi. Menunjukkan apakah Multi-AZ diaktifkan untuk domain.

Tipe: [AvailabilityOptionsStatus](#)

Diperlukan: Tidak

## UpdateScalingParametersResult

### Deskripsi

Hasil `UpdateScalingParameters` permintaan. Berisi status parameter penskalaan yang baru dikonfigurasi.

### Daftar Isi

#### ScalingParameters

Status dan konfigurasi parameter penskalaan domain pencarian.

Tipe: [ScalingParametersStatus](#)

Diperlukan: Ya

## UpdateServiceAccessPoliciesResult

### Deskripsi

Hasil dari UpdateServiceAccessPolicies permintaan. Berisi kebijakan akses baru.

### Daftar Isi

#### AccessPolicies

Aturan akses yang dikonfigurasi untuk domain.

Tipe: [AccessPoliciesStatus](#)

Diperlukan: Ya

## Parameter Umum

Bagian ini mencantumkan parameter permintaan yang digunakan semua tindakan. Parameter spesifik tindakan apa pun tercantum dalam topik untuk tindakan tersebut.

### Aksi

Tindakan yang harus dilakukan.

Default: Tidak Ada

Tipe string

Diperlukan: Ya

### AuthParams

Parameter yang diperlukan untuk mengautentikasi permintaan Bersyarat. Berisi:

- AWSSignatureVersion
- SignatureVersion
- Stempel Waktu
- Tanda tangan

Default: Tidak Ada



Diperlukan: Kondisional

### AWSAccessKeyId

ID kunci akses yang sesuai dengan kunci akses rahasia yang Anda gunakan untuk menandatangani permintaan.

Default: Tidak Ada

Tipe string

Diperlukan: Ya

### Kedaluwarsa

Tanggal dan waktu ketika tanda tangan permintaan kedaluwarsa, dinyatakan dalam format YYYY-MM-DDTHH:mm:ssZ, sebagaimana ditentukan dalam standar ISO 8601.

Kondisi: Permintaan harus menyertakan Timestamp atau Kedaluwarsa, tetapi tidak keduanya.

Default: Tidak Ada

Tipe: string

Wajib: Bersyarat

### SecurityToken

Token keamanan sementara yang diperoleh melalui panggilan ke AWS Security Token Service. Untuk daftar layanan yang mendukung AWS Security Token Service, buka [Using Temporary Security Credentials to Access AWS di Menggunakan Temporary Security Credentials](#).

Default: Tidak Ada

Jenis: string

Wajib: Tidak

### Tanda tangan

Tanda tangan digital yang Anda buat untuk permintaan tersebut. Untuk informasi tentang membuat tanda tangan, buka dokumentasi pengembang layanan.

Default: Tidak Ada

### Tipe string

Diperlukan: Ya

### SignatureMethod

Algoritme hash yang Anda gunakan untuk membuat tanda tangan permintaan.

Default: Tidak Ada

Tipe: string

Nilai yang Valid: HmacSHA256 | HmacSHA1

Diperlukan: Ya

### SignatureVersion

Versi tanda tangan yang Anda gunakan untuk menandatangani permintaan. Tetapkan ini ke nilai yang direkomendasikan untuk layanan Anda.

Default: Tidak Ada

Tipe string

Diperlukan: Ya

### Stempel waktu

Tanggal dan waktu ketika permintaan ditandatangani, dinyatakan dalam format YYYY-MM-DDTHH:MM:SSZ, sebagaimana ditentukan dalam standar ISO 8601.

Kondisi: Permintaan harus menyertakan Timestamp atau Kedaluwarsa, tetapi tidak keduanya.

Default: Tidak Ada

Tipe: string

Wajib: Bersyarat

### Versi

Versi API yang ditulis dalam permintaan, dinyatakan dalam format HH-BB-TTTT.

Default: Tidak Ada

Tipe string

Diperlukan: Ya

## Kesalahan Umum

Bagian ini mencantumkan kesalahan umum yang dikembalikan semua tindakan. Setiap kesalahan khusus tindakan tercantum dalam topik untuk tindakan tersebut.

### IncompleteSignature

Tanda tangan yang diminta tidak sesuai dengan standar AWS.

Kode Status HTTP: 400

### InternalFailure

Pemrosesan permintaan telah gagal karena kesalahan yang tidak diketahui, pengecualian atau kegagalan.

Kode Status HTTP: 500

### InvalidAction

Tindakan atau operasi yang diminta tidak valid. Verifikasi bahwa tindakan diketik dengan benar.

Kode Status HTTP: 400

### InvalidClientId

Sertifikat X.509 atau access key ID AWS yang disediakan tidak ada dalam catatan kami.

Kode Status HTTP: 403

### InvalidParameterCombination

Parameter yang tidak boleh digunakan secara bersamaan digunakan secara bersamaan.

Kode Status HTTP: 400

### InvalidParameterValue

out-of-range Nilai atau tidak valid diberikan untuk parameter input.

Kode Status HTTP: 400

### InvalidQueryParameter

String kueri AWS salah bentuk atau tidak mematuhi standar AWS.

Kode Status HTTP: 400

### MalformedQueryString

String kueri berisi kesalahan sintaks.

Kode Status HTTP: 404

### MissingAction

Permintaan tidak memiliki tindakan atau parameter yang diperlukan.

Kode Status HTTP: 400

### MissingAuthenticationToken

Permintaan harus berisi access key ID AWS yang valid (terdaftar) atau sertifikat X.509.

Kode Status HTTP: 403

### MissingParameter

Parameter yang diperlukan untuk tindakan tertentu tidak disediakan.

Kode Status HTTP: 400

### OptInRequired

Access key ID AWS membutuhkan langganan untuk layanan ini.

Kode Status HTTP: 403

### RequestExpired

Permintaan menjangkau layanan lebih dari 15 menit setelah stempel tanggal pada permintaan atau lebih dari 15 menit setelah tanggal kedaluwarsa permintaan (seperti untuk URL pre-signed), atau stempel tanggal pada permintaan lebih dari 15 menit di masa mendatang.

Kode Status HTTP: 400

### ServiceUnavailable

Permintaan telah gagal karena kegagalan sementara server.

Kode Status HTTP: 503

## Pelambatan

Permintaan ditolak karena throttling permintaan.

Kode Status HTTP: 400

## ValidationError

Input gagal memenuhi batasan yang ditentukan oleh layanan AWS.

Kode Status HTTP: 400

## Referensi API Layanan Dokumen untuk Amazon CloudSearch

Anda menggunakan API layanan dokumen untuk menambah, mengganti, atau menghapus dokumen di CloudSearch domain Amazon Anda. Untuk informasi selengkapnya mengelola dokumen di domain penelusuran Anda, lihat [upload documents](#).

API lain yang Anda gunakan untuk berinteraksi dengan Amazon CloudSearch adalah:

- [Referensi API Konfigurasi untuk Amazon CloudSearch](#)—Siapkan dan kelola domain pencarian Anda.
- [Search API](#)—Cari domain Anda.

## dokumen/batch

Bagian ini menjelaskan permintaan HTTP dan pesan respons untuk `documents/batch` sumber daya.

Anda membuat kumpulan dokumen untuk mendeskripsikan data yang ingin Anda unggah ke CloudSearch domain Amazon. Kumpulan dokumen adalah kumpulan operasi tambah dan hapus yang mewakili dokumen yang ingin Anda tambahkan, perbarui, atau hapus dari domain Anda. Batch dapat dijelaskan dalam JSON atau XHTML. Batch menyediakan semua informasi yang CloudSearch dibutuhkan Amazon untuk pengindeksan. Setiap item yang ingin Anda kembalikan sebagai hasil penelusuran (seperti produk) direpresentasikan sebagai dokumen—kumpulan hanyalah kumpulan permintaan tambah dan hapus untuk dokumen individual. Setiap dokumen memiliki ID unik dan satu atau beberapa bidang yang berisi data yang ingin Anda cari dan kembalikan hasilnya.

Untuk memperbarui dokumen, Anda menentukan permintaan tambah dengan ID dokumen dokumen yang ingin Anda perbarui. Untuk informasi selengkapnya, lihat [Menambahkan dan Memperbarui](#)

[Dokumen di Amazon CloudSearch](#). Demikian pula, untuk menghapus dokumen, Anda mengirimkan permintaan hapus dengan ID dokumen dari dokumen yang ingin Anda hapus. Untuk informasi tentang menghapus dokumen, lihat [Menghapus Dokumen di Amazon CloudSearch](#).

Untuk informasi selengkapnya tentang mengirimkan data untuk pengindeksan, lihat [upload documents](#)

## dokumen/batch JSON API

### Dokumen JSON/Permintaan batch

Badan `documents/batch` permintaan menggunakan JSON atau XML untuk menentukan operasi dokumen yang ingin Anda lakukan. Representasi JSON dari batch adalah kumpulan objek yang mendefinisikan operasi tambah dan hapus individu. `type` Properti mengidentifikasi apakah suatu objek mewakili operasi tambah atau hapus. Misalnya, batch JSON berikut menambahkan satu dokumen dan menghapus satu dokumen:

```
[
  { "type": "add",
    "id": "tt0484562",
    "fields": {
      "title": "The Seeker: The Dark Is Rising",
      "directors": ["Cunningham, David L."],
      "genres": ["Adventure", "Drama", "Fantasy", "Thriller"],
      "actors": ["McShane, Ian", "Eccleston, Christopher", "Conroy, Frances",
        "Crewson, Wendy", "Ludwig, Alexander", "Cosmo, James",
        "Warner, Amelia", "Hickey, John Benjamin", "Piddock, Jim",
        "Lockhart, Emma"]
    }
  },
  { "type": "delete",
    "id": "tt0484575"
  }
]
```

#### Note

Saat menentukan kumpulan dokumen di JSON, nilai untuk bidang tidak bisa `null`

Representasi [skema JSON](#) dari batch ditunjukkan di bawah ini:

```

{
  "type": "array",
  "minItems": 1,
  "items": {
    "type": "object",
    "properties": {
      "type": {
        "type": "string",
        "enum": ["add", "delete"],
        "required": true
      },
      "id": {
        "type": "string",
        "pattern": "[a-z0-9][a-z0-9_]{0,127}",
        "minLength": 1,
        "maxLength": 128,
        "required": true
      },
      "fields": {
        "type": "object",
        "patternProperties": {
          "[a-zA-Z0-9][a-zA-Z0-9_]{0,63}": {
            "type": "string",
          }
        }
      }
    }
  }
}

```

### dokumen/Properti Permintaan batch (JSON)

Properti	Deskripsi	Diperlukan
type	Jenis operasi, add atau delete.	Ya
id	String alfanumerik. Karakter yang diizinkan adalah: A-Z (huruf besar), -a-z (huruf kecil), 0-9, _ (garis bawah), - (tanda hubung),/(garis miring), # (tanda hash),: (titik dua). Panjang maksimal adalah 128 karakter.	Ya

Properti	Deskripsi	Diperlukan
ladang	<p>Kumpulan satu atau lebih properti field_name yang menentukan bidang yang berisi dokumen.</p> <p>Kondisi: Diperlukan untuk menambah operasi. Harus berisi setidaknya satu properti field_name.</p>	Bersyarat
field_name	<p>Menentukan bidang dalam dokumen yang ditambahkan. Nama bidang harus dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan _ (garis bawah). Nama bidang harus minimal 3 dan tidak lebih dari 64 karakter. Skor nama dicadangkan dan tidak dapat digunakan sebagai nama bidang.</p> <p>Untuk menentukan beberapa nilai untuk bidang, Anda menentukan larik nilai, bukan nilai tunggal. Sebagai contoh:</p> <pre>"genre": ["Adventure", "Drama", "Fantasy", "Thriller"]</pre> <p>Kondisi: Setidaknya satu bidang harus ditentukan dalam objek bidang.</p>	Bersyarat

## dokumen/Batch Response (JSON)

Badan respons mencantumkan jumlah penambahan dan penghapusan yang dilakukan dan kesalahan atau peringatan apa pun yang dihasilkan.

Representasi skema JSON dari respons API layanan dokumen ditunjukkan di bawah ini:

```
{
  "type": "object",
  "properties": {
    "status": {
```



```
    "type": "text",
    "enum": ["success", "error"],
    "required": true
  },
  "adds": {
    "type": "integer",
    "minimum": 0,
    "required": true
  },
  "deletes": {
    "type": "integer",
    "minimum": 0,
    "required": true
  },
  "errors": {
    "type": "array",
    "required": false,
    "items": {
      "type": "object",
      "properties": {
        "message": {
          "type": "string",
          "required": true
        }
      }
    }
  },
  "warnings": {
    "type": "array",
    "required": false,
    "items": {
      "type": "object",
      "properties": {
        "message": {
          "type": "string",
          "required": true
        }
      }
    }
  }
}
```

## dokumen/properti respons batch (JSON)

Properti	Deskripsi
status	Status hasil, yaitu salah satu <code>success</code> atau <code>error</code> .
menambahkan	Jumlah operasi tambah dokumen yang dilakukan. Selalu nol saat statusnya <code>error</code> .
menghapus	Jumlah operasi hapus dokumen yang dilakukan. Selalu nol saat statusnya <code>error</code> . Untuk informasi tentang menghapus dokumen secara permanen, lihat <a href="#">the section called "Menghapus Dokumen"</a> .
kesalahan	Memberikan informasi tentang kesalahan penguraian atau validasi. Ditentukan hanya jika statusnya <code>error</code> .
memperingati	Memberikan informasi tentang peringatan yang dihasilkan selama penguraian atau validasi.

## dokumen/batch XMLAPI

## Dokumen XML/Permintaan batch

Badan `documents/batch` permintaan menentukan operasi dokumen yang ingin Anda lakukan dalam XHTML. Sebagai contoh:

```
<batch>
  <add id="tt0484562">
    <field name="title">The Seeker: The Dark Is Rising</field>
    <field name="director">Cunningham, David L.</field>
    <field name="genre">Adventure</field>
    <field name="genre">Drama</field>
    <field name="genre">Fantasy</field>
    <field name="genre">Thriller</field>
    <field name="actor">McShane, Ian</field>
    <field name="actor">Eccleston, Christopher</field>
    <field name="actor">Conroy, Frances</field>
    <field name="actor">Ludwig, Alexander</field>
    <field name="actor">Crewson, Wendy</field>
    <field name="actor">Warner, Amelia</field>
  </add>
</batch>
```

```

<field name="actor">Cosmo, James</field>
<field name="actor">Hickey, John Benjamin</field>
<field name="actor">Piddock, Jim</field>
<field name="actor">Lockhart, Emma</field>
</add>
<delete id="tt0301199" />
</batch>

```

## dokumen/batch Permintaan Elemen (XHTML)

Elemen	Deskripsi	Diperlukan
batch	Kumpulan operasi tambah atau hapus yang ingin Anda kirimkan ke domain pencarian Anda. Batch harus berisi setidaknya satu elemen tambah atau hapus.	Ya
tambahkan	Menentukan dokumen yang ingin Anda tambahkan ke domain pencarian Anda. Atribut id diperlukan dan elemen add harus berisi setidaknya satu bidang.  Atribut: <ul style="list-style-type: none"> <li>id—Sebuah string alfanumerik. Karakter apa pun selain A-Z (huruf besar atau kecil) dan 0-9 adalah ilegal. Panjang maksimal adalah 128 karakter.</li> </ul>	Tidak
bidang	Menentukan bidang dalam dokumen yang ditambahkan. Atribut nama dan nilai bidang diperlukan. Nama bidang harus dimulai dengan huruf dan dapat berisi karakter berikut: a-z (huruf kecil), 0-9, dan _ (garis bawah). Skor nama dicadangkan dan tidak dapat digunakan sebagai nama bidang. Nilai bidang dapat berupa teks atau CDATA.	Bersyarat

Elemen	Deskripsi	Diperlukan
	<p>Untuk menentukan beberapa nilai untuk bidang, Anda menyertakan beberapa elemen bidang dengan nama yang sama. Sebagai contoh:</p> <pre data-bbox="474 424 1149 667">&lt;field name="genre"&gt;Adventure&lt;/field&gt; &lt;field name="genre"&gt;Drama&lt;/field&gt; &lt;field name="genre"&gt;Fantasy&lt;/field&gt; &lt;field name="genre"&gt;Thriller&lt;/field&gt;</pre> <p><b>Batasan:</b></p> <ul data-bbox="474 856 1123 1033" style="list-style-type: none"><li>• name—String alfanumerik yang dimulai dengan huruf. Dapat berisi a-z (huruf kecil), 0-9, _ (garis bawah), - (tanda hubung), dan . (periode).</li></ul> <p><b>Kondisi:</b> Setidaknya satu bidang harus ditentukan dalam elemen add.</p>	

Elemen	Deskripsi	Diperlukan
hapus	<p>Menentukan dokumen yang ingin Anda hapus dari domain pencarian Anda. Atribut id diperlukan. Elemen hapus harus kosong. Untuk informasi tentang menghapus dokumen secara permanen, lihat <a href="#">the section called “Menghapus Dokumen”</a>.</p> <p>Batasan:</p> <ul style="list-style-type: none"> <li>• <code>id</code>—Sebuah string alfanumerik. Karakter apa pun selain A-Z (huruf besar atau kecil) dan 0-9 adalah ilegal.</li> </ul>	Tidak

#### dokumen/Batch Response (XHTML)

Badan respons mencantumkan jumlah penambahan dan penghapusan yang dilakukan dan kesalahan atau peringatan apa pun yang dihasilkan.

Skema RelaxNG dari respons API layanan dokumen adalah:

```

start = response

response = element response {
  attribute status { "success" | "error" },
  attribute adds { xsd:integer },
  attribute deletes { xsd:integer },
  element errors {
    element error {
      text
    }+
  }? &
  element warnings {
    element warning {
      text
    }+
  }?
}

```

}

## dokumen/batch Response Elements (XHTML)

Elemen	Deskripsi
result	<p>Berisi elemen yang mencantumkan kesalahan dan peringatan yang dihasilkan saat mengurai dan memvalidasi permintaan.</p> <p>Atribut:</p> <ul style="list-style-type: none"> <li>• <code>status</code>—Status hasil, yaitu salah satu <code>success</code> atau <code>error</code>.</li> <li>• <code>addsJumlah</code> dokumen yang ditambahkan. Jika statusnya <code>error</code>, ini selalu nol.</li> <li>• <code>deletesJumlah</code> dokumen yang dihapus. Jika statusnya <code>error</code>, ini selalu nol.</li> </ul> <p>Kendala: Jika statusnya <code>error</code>, elemen hasil berisi daftar kesalahan. Jika statusnya <code>success</code>, elemen hasil dapat berisi daftar peringatan, tetapi tidak ada kesalahan.</p>
kesalahan	Berisi kumpulan elemen kesalahan yang mengidentifikasi kesalahan yang terjadi saat mengurai dan memvalidasi permintaan.
kesalahan	Memberikan informasi tentang kesalahan penguraian atau validasi. Nilai memberikan deskripsi kesalahan.
peringatan	Berisi kumpulan elemen peringatan yang mengidentifikasi peringatan yang dihasilkan saat mengurai dan memvalidasi permintaan.
memperingati	Memberikan informasi tentang peringatan parsing atau validasi. Nilai memberikan deskripsi kesalahan.

## dokumen/kode status batch

Permintaan layanan dokumen dapat mengembalikan tiga jenis kode status:

- Kode status 5xx menunjukkan bahwa ada kesalahan server internal. Kami merekomendasikan untuk menangkap dan mencoba kembali semua kode kesalahan 5xx karena biasanya mewakili kondisi kesalahan sementara.
- Kode status 4xx menunjukkan bahwa permintaan itu salah bentuk.
- Kode status 2xx menunjukkan bahwa permintaan berhasil diproses.

Kesalahan	Deskripsi	Kode Status HTTP
Tidak Ada Tipe Konten	Header Content-Type tidak ada.	400
Tanpa Konten-Panjang	Header Content-Length tidak ada.	411
Jalan yang salah	Jalur URL tidak cocok dengan "/yyyy-mm-dd/dokumen/batch".	404
Metode HTTP Tidak Valid	Metode HTTP bukan POST. Permintaan harus diposting ke dokumen/batch.	405
Jenis Terima Tidak Valid	Accept header menentukan jenis konten selain "application/xml" atau "application/json". Tanggapan dapat dikirim hanya sebagai XML/JSON.	406
Permintaan Terlalu Besar	Panjang badan permintaan lebih besar dari nilai maksimum yang diizinkan.	413
Jenis Konten Tidak Valid	Jenis konten adalah sesuatu selain "application/json" atau "application/xml".	415
Set Karakter Tidak Valid	Set karakter adalah sesuatu selain "ASCII", "ISO-8859-1", atau "UTF-8".	415

## Header Permintaan Umum

Nama	Deskripsi	Diperlukan
Content-Type	<p>Tipe MIME standar yang menjelaskan format data objek. Untuk informasi lebih lanjut, lihat <a href="#">W3C RFC 2616</a> Bagian 14.</p> <p>Default: aplikasi/json</p> <p>Kendala: aplikasi/json atau aplikasi/xmlsaja</p>	Diperlukan
Content-Length	Panjang dalam byte dari badan permintaan.	Ya
Menerima	<p>Tipe MIME standar yang menjelaskan format data respons. Untuk informasi lebih lanjut, lihat <a href="#">W3C RFC 2616</a> Bagian 14.</p> <p>Default: tipe konten permintaan</p> <p>Kendala: aplikasi/json atau aplikasi/xmlsaja</p>	Tidak

## Header Respons Umum

Nama	Penjelasan
Content-Type	<p>Tipe MIME standar yang menjelaskan format data objek. Untuk informasi lebih lanjut, lihat <a href="#">W3C RFC 2616</a> Bagian 14.</p> <p>Default: nilai header Terima dalam permintaan, atau Tipe Konten permintaan jika header Terima hilang atau tidak menentukan aplikasi/xml/aplikasi/json.</p> <p>Kendala: aplikasi/xml atau aplikasi/json saja</p>
Content-Length	Panjang dalam byte tubuh dalam respon.



# Cari Referensi API untuk Amazon CloudSearch

## Topik

- [Pencarian](#)
- [Mengirimkan Permintaan Saran di Amazon CloudSearch](#)
- [Sarankan](#)
- [Kesalahan Layanan Pencarian](#)

Anda menggunakan API Penelusuran untuk mengirimkan permintaan pencarian atau saran ke CloudSearch domain Amazon Anda. Untuk informasi lebih lanjut tentang pencarian, lihat [Mencari Data Anda dengan Amazon CloudSearch](#). Untuk informasi lebih lanjut tentang saran, lihat [Mendapatkan Saran Pelengkapan Otomatis di Amazon CloudSearch](#).

API lain yang Anda gunakan untuk berinteraksi dengan Amazon CloudSearch adalah:

- [API Konfigurasi](#) —Siapkan dan kelola domain penelusuran Anda.
- [Document Service API](#) —Kirim data yang ingin Anda cari.

## Pencarian

Bagian ini menjelaskan permintaan HTTP dan pesan respons untuk sumber pencarian.

### Sintaks Pencarian

```
GET /2013-01-01/search
```

### Header Permintaan Pencarian

#### HOST

Titik akhir permintaan pencarian untuk domain yang Anda kueri. Anda dapat menggunakan [DescribeDomains](#) untuk mengambil titik akhir permintaan pencarian domain Anda.

Diperlukan: Ya

## Parameter Permintaan Pencarian

### cursor

Mengambil nilai cursor yang dapat Anda gunakan untuk halaman melalui set hasil besar. Gunakan `size` parameter untuk mengontrol jumlah klik yang ingin Anda sertakan dalam setiap respons. Anda dapat menentukan `start` parameter `cursor` atau dalam permintaan, keduanya saling eksklusif. Untuk informasi selengkapnya, lihat [Paginate the results](#).

Untuk mendapatkan cursor pertama, tentukan `cursor=initial` dalam permintaan awal Anda. Dalam permintaan berikutnya, tentukan nilai cursor yang dikembalikan di bagian `hits` respons.

Misalnya, permintaan berikut menetapkan nilai cursor ke `initial` dan `size` parameter ke 100 untuk mendapatkan set hit pertama. Cursor untuk set hit berikutnya disertakan dalam respons.

```
search?q=john&cursor=initial&size=100&return=_no_fields
{
  "status": {
    "rid": "+/Xu5s0oHwojC6o=",
    "time-ms": 15
  },
  "hits": {
    "found": 503,
    "start": 0,
    "cursor": "VegKzpYYQW9JSVFFRU1UeWwwZERBd09EUTNPRGM9ZA",
    "hit": [
      {"id": "tt0120601"},
      {"id": "tt1801552"},
      ...
    ]
  }
}
```

Untuk mendapatkan set hit berikutnya, Anda menentukan nilai cursor dan jumlah klik yang akan diambil.

```
search?q=john&cursor=VegKzpYYQW9JSVFFRU1UeWwwZERBd09EUTNPRGM9ZA&size=100
```

Tipe: String

Wajib: Tidak

## expr.NAME

Mendefinisikan ekspresi yang dapat digunakan untuk mengurutkan hasil. Anda juga dapat menentukan ekspresi sebagai bidang kembali. Untuk informasi selengkapnya tentang mendefinisikan dan menggunakan ekspresi, lihat [Mengkonfigurasi Ekspresi](#).

Anda dapat menentukan dan menggunakan beberapa ekspresi dalam permintaan pencarian. Misalnya, permintaan berikut membuat dua ekspresi yang digunakan untuk mengurutkan hasil dan menyertakannya dalam hasil pencarian:

```
search?q=(and (term field=genres 'Sci-Fi')(term field=genres 'Comedy'))&q.parser=structured
&expr.expression1=_score*rating
&expr.expression2=(1/rank)*year
&sort=expression1 desc,expression2 desc
&return=title,rating,rank,year,_score,expression1,expression2
```

Tipe: String

Wajib: Tidak

## facet.FIELD

Menentukan bidang yang Anda ingin mendapatkan informasi facet untuk- FIELD adalah nama bidang. Bidang yang ditentukan harus facet diaktifkan dalam konfigurasi domain. Opsi facet ditentukan sebagai objek JSON. Jika objek JSON kosong, `facet.FIELD={}`, jumlah faset dihitung untuk semua nilai bidang, faset diurutkan berdasarkan jumlah faset, dan 10 aspek teratas dikembalikan dalam hasil.

Anda dapat menentukan tiga opsi di objek JSON:

- `sort` menentukan bagaimana Anda ingin mengurutkan aspek dalam hasil: `bucket` atau `count`. Tentukan `bucket` untuk mengurutkan menurut abjad atau numerik berdasarkan nilai faset (dalam urutan menaik). Tentukan `count` untuk mengurutkan berdasarkan jumlah faset yang dihitung untuk setiap nilai faset (dalam urutan menurun). Untuk mengambil jumlah faset untuk nilai atau rentang nilai tertentu, gunakan `buckets` opsi alih-alih `sort`.
- `buckets` menentukan array dari nilai faset atau rentang yang ingin Anda hitung. Ember dikembalikan sesuai urutan yang ditentukan dalam permintaan. Untuk menentukan rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Braket persegi, [atau], menunjukkan bahwa ikatan termasuk dalam rentang, penjepit keriting, {atau}, tidak termasuk batas. Anda

dapat menghilangkan batas atas atau bawah untuk menentukan rentang terbuka. Saat menghilangkan ikatan, Anda harus menggunakan penjepit keriting. `sizeOpsi sort` dan tidak valid jika Anda menentukan `buckets`.

- `size` menentukan jumlah maksimum aspek untuk dimasukkan dalam hasil. Secara default, Amazon CloudSearch mengembalikan jumlah untuk 10 besar. `sizeParameter` hanya valid ketika Anda menentukan `sort` opsi; itu tidak dapat digunakan bersama dengan `buckets`.

Misalnya, permintaan berikut mendapatkan jumlah faset untuk `year` bidang, mengurutkan jumlah faset berdasarkan nilai dan mengembalikan jumlah untuk tiga teratas:

```
facet.year={sort:"bucket", size:3}
```

Untuk menentukan nilai atau rentang nilai yang ingin Anda hitung jumlah faset, gunakan opsi `buckets`. Misalnya, permintaan berikut menghitung dan mengembalikan jumlah faset menurut dekade:

```
facet.year={buckets:["[1970,1979]", "[1980,1989]",  
                    "[1990,1999]", "[2000,2009]",  
                    "[2010,}"]}
```

Anda juga dapat menentukan nilai individual sebagai bucket:

```
facet.genres={buckets:["Action", "Adventure", "Sci-Fi"]}
```

Perhatikan bahwa nilai faset peka huruf besar—dengan sampel data film IMDb, jika Anda menentukan, `["action", "adventure", "sci-fi"]` bukan, semua jumlah `["Action", "Adventure", "Sci-Fi"]` faset adalah nol.

Tipe: String

Wajib: Tidak

format

Menentukan jenis konten respon.

Jenis: String

Nilai yang Valid: json|xml|

Default: json

Diperlukan: Tidak

fq

Menentukan query terstruktur yang menyaring hasil pencarian tanpa mempengaruhi bagaimana hasil dinilai dan diurutkan. Anda menggunakan fq bersama dengan q parameter untuk memfilter dokumen yang cocok dengan batasan yang ditentukan dalam parameter. q Menentukan filter hanya mengontrol dokumen pencocokan mana yang disertakan dalam hasil, itu tidak berpengaruh pada bagaimana mereka dinilai dan diurutkan. fqParameter mendukung sintaks kueri terstruktur penuh. Untuk informasi selengkapnya tentang menggunakan filter, lihat[Memfilter Dokumen yang Cocokan](#). Untuk informasi selengkapnya tentang kueri terstruktur, lihat[Sintaks Pencarian Terstruktur](#).

Tipe: String

Wajib: Tidak

highlight.FIELD

Mengambil sorotan untuk kecocokan di `text-array` bidang `text` atau yang ditentukan. Opsi sorotan ditentukan sebagai objek JSON. Jika objek JSON kosong, teks bidang yang dikembalikan diperlakukan sebagai HTML dan kecocokan pertama disorot dengan tag penekanan:`<em>search-term</em>`.

Anda dapat menentukan empat opsi di objek JSON:

- `format`—menentukan format data di bidang teks: `text` atau `html`. Ketika data dikembalikan sebagai HTML, semua karakter non-alfanumerik dikodekan. Default-nya adalah `html`.
- `max_phrases`—menentukan jumlah maksimum kemunculan istilah pencarian yang ingin Anda sorot. Secara default, kejadian pertama disorot.
- `pre_tag`—menentukan string untuk ditambahkan ke kemunculan istilah pencarian. Default untuk sorotan HTML adalah `<em>`. Default untuk sorotan teks adalah `*`.
- `post_tag`—menentukan string untuk ditambahkan ke kemunculan istilah pencarian. Default untuk sorotan HTML adalah `</em>`. Default untuk sorotan teks adalah `*`.

```
Contoh:highlight.plot={},
highlight.plot={format:'text',max_phrases:2,pre_tag:'<b>',post_tag:'</b>'}
```

Tipe: String

## Wajib: Tidak

### partial

Mengontrol apakah hasil sebagian dikembalikan jika satu atau lebih partisi indeks tidak tersedia. Ketika indeks pencarian Anda dipartisi di beberapa instance penelusuran, secara default Amazon CloudSearch hanya mengembalikan hasil jika setiap partisi dapat ditanyakan. Ini berarti bahwa kegagalan satu contoh pencarian dapat mengakibatkan kesalahan 5xx (server internal). Saat Anda menentukan `partial=true`, Amazon CloudSearch mengembalikan hasil apa pun yang tersedia dan menyertakan persentase dokumen yang dicari di hasil penelusuran (`percent-searched`). Hal ini memungkinkan Anda untuk lebih anggun menurunkan pengalaman pencarian pengguna Anda. Misalnya, daripada tidak menampilkan hasil, Anda dapat menampilkan sebagian hasil dan pesan yang menunjukkan bahwa hasilnya mungkin tidak lengkap karena pemadaman sistem sementara.

Jenis: Boolean

Default: Salah

Diperlukan: Tidak

### pretty

Format output JSON sehingga lebih mudah dibaca.

Jenis: Boolean

Default: Salah

Diperlukan: Tidak

### q

Kriteria pencarian untuk permintaan tersebut. Cara Anda menentukan kriteria pencarian tergantung pada parser kueri yang digunakan untuk permintaan dan opsi parser yang ditentukan dalam parameter `q.options`. Secara default, parser `simple` kueri digunakan untuk memproses permintaan. Untuk menggunakan `structured`, `lucene`, atau `dismax` query parser, Anda juga harus menentukan `q.parser` parameter. Untuk informasi selengkapnya tentang menentukan kriteria pencarian, lihat [Mencari Data Anda dengan Amazon CloudSearch](#).

Tipe: String

Diperlukan: Ya

## q.options

Konfigurasi opsi untuk parser kueri yang ditentukan dalam `q.parser` parameter. Opsi ditentukan sebagai objek JSON, misalnya: `q.options={defaultOperator: 'or', fields: ['title^5', 'description']}`.

Opsi yang dapat Anda konfigurasi bervariasi sesuai dengan parser mana yang Anda gunakan:

- `defaultOperator`—Operator default digunakan untuk menggabungkan istilah individual dalam string pencarian. Misalnya: `defaultOperator: 'or'`. Untuk `dismax` parser, Anda menentukan persentase yang mewakili persentase istilah dalam string pencarian (dibulatkan ke bawah) yang harus cocok, bukan operator default. Nilai `0%` setara dengan OR, dan nilai `100%` setara dengan AND. Persentase harus ditentukan sebagai nilai dalam kisaran 0-100 diikuti oleh simbol persen (%). Misalnya, `defaultOperator: 50%`. Nilai yang valid: `and,or`, persentase dalam kisaran `0%-100%` (). `dismax` Default: `and (simplestructured,,lucene)` atau `100 (dismax)`. Berlaku untuk: `simple,structured,lucene, dandismax`.
- `fields`—Sebuah array bidang untuk mencari ketika tidak ada bidang yang ditentukan dalam pencarian. Jika tidak ada bidang yang ditentukan dalam pencarian dan opsi ini tidak ditentukan, semua dikonfigurasi secara statis `text` dan `text-array` bidang dicari. Anda dapat menentukan bobot untuk setiap bidang untuk mengontrol kepentingan relatif setiap bidang saat Amazon CloudSearch menghitung skor relevansi. Untuk menentukan berat bidang, tambahkan simbol tanda sisipan (^) dan bobot ke nama bidang. Misalnya, untuk meningkatkan pentingnya bidang di atas `title` bidang, `description` Anda dapat menentukan: `fields: ['title^5', 'description']`. Nilai yang valid: Nama bidang yang dikonfigurasi dan nilai numerik opsional lebih besar dari nol. Default: Semua dikonfigurasi secara statis `text` dan `text-array` bidang. Bidang dan `literal` bidang dinamis tidak dicari secara default. Berlaku untuk: `simple,structured,lucene, dandismax`.
- `operators`—Array operator atau karakter khusus yang ingin Anda nonaktifkan untuk parser kueri sederhana. Jika Anda menonaktifkan `and,or`, atau `not` operator, operator yang sesuai (`+`, `|`, `-`) tidak memiliki arti khusus dan dijatuhkan dari string pencarian. Demikian pula, menonaktifkan `prefix` menonaktifkan operator wildcard (`*`) dan `phrase` menonaktifkan kemampuan untuk mencari frasa dengan melampirkan frasa dalam tanda kutip ganda. Menonaktifkan `priority` menonaktifkan kemampuan untuk mengontrol urutan prioritas menggunakan tanda kurung. Menonaktifkan `near` menonaktifkan kemampuan untuk menggunakan operator `~` untuk melakukan pencarian frasa yang ceroboh. Menonaktifkan `fuzzy` operator menonaktifkan kemampuan untuk menggunakan operator `~` untuk melakukan pencarian fuzzy. `escape` menonaktifkan kemampuan untuk menggunakan garis miring terbalik (`\`) untuk melarikan diri dari karakter khusus dalam string pencarian.

Menonaktifkan spasi putih adalah opsi lanjutan yang mencegah parser melakukan tokenisasi pada spasi putih, yang dapat berguna untuk orang Vietnam. (Ini mencegah kata-kata Vietnam dipecah secara tidak benar.) Misalnya, Anda dapat menonaktifkan semua operator selain operator frasa untuk mendukung hanya kueri istilah dan frasa sederhana: `operators:['and', 'not', 'or', 'prefix']`. Nilai yang valid: `andescape,fuzzy,,near,not,or,phrase,precedence,prefix,whitespace`. Default: Semua operator dan karakter khusus diaktifkan. Berlaku untuk: `simple`.

- `phraseFields`—Sebuah array dari `text` atau `text-array` bidang yang ingin Anda gunakan untuk pencarian frase. Ketika istilah dalam string pencarian muncul dalam jarak dekat dalam bidang, skor bidang lebih tinggi. Anda dapat menentukan bobot untuk setiap bidang untuk meningkatkan skor itu. `phraseSlop` Opsi ini mengontrol seberapa banyak kecocokan dapat menyimpang dari string pencarian dan masih ditingkatkan. Untuk menentukan berat bidang, tambahkan simbol tanda sisipan (^) dan bobot ke nama bidang. Misalnya, untuk meningkatkan kecocokan frase di `title` bidang di atas `abstract` bidang, Anda dapat menentukan: Nilai yang `phraseFields:['title^3', 'abstract']` valid: Nama `text-array` bidang `text` atau apa pun dan nilai numerik opsional lebih besar dari nol. Default: Tidak ada bidang. Jika Anda tidak menentukan bidang apa pun `phraseFields`, skor kedekatan dinonaktifkan meskipun `phraseSlop` ditentukan. Berlaku untuk: `dismax`.
- `phraseSlop`—Nilai integer yang menentukan berapa banyak kecocokan dapat menyimpang dari frase pencarian dan masih ditingkatkan sesuai dengan bobot yang ditentukan dalam opsi. `phraseFields` Misalnya, `phraseSlop: 2`. Anda juga harus menentukan `phraseFields` untuk mengaktifkan penilaian kedekatan. Nilai yang valid: bilangan bulat positif. Default: 0. Berlaku untuk: `dismax`.
- `explicitPhraseSlop`—Nilai integer yang menentukan seberapa banyak kecocokan dapat menyimpang dari frase pencarian ketika frase diapit tanda kutip ganda dalam string pencarian. (Frasa yang melebihi jarak kedekatan ini tidak dianggap cocok.) `explicitPhraseSlop: 5`. Nilai yang valid: bilangan bulat positif. Default: 0. Berlaku untuk: `dismax`.
- `tieBreaker`—Ketika sebuah istilah dalam string pencarian ditemukan di bidang dokumen, skor dihitung untuk bidang tersebut berdasarkan seberapa umum kata tersebut di bidang itu dibandingkan dengan dokumen lain. Jika istilah muncul di beberapa bidang dalam dokumen, secara default hanya bidang penilaian tertinggi yang berkontribusi pada skor keseluruhan dokumen. Anda dapat menentukan `tieBreaker` nilai untuk mengaktifkan kecocokan di bidang dengan skor lebih rendah untuk berkontribusi pada skor dokumen. Dengan begitu, jika dua dokumen memiliki skor bidang maks yang sama untuk istilah tertentu, skor untuk dokumen yang memiliki kecocokan di lebih banyak bidang akan lebih tinggi. Rumus untuk menghitung skor dengan `TieBreaker` adalah:



```
(max field score) + (tieBreaker) * (sum of the scores for the rest of the matching fields)
```

Misalnya, kueri berikut mencari istilah dog di review bidangtitle,description, dan disetel tieBreaker ke 0.1:

```
q=dog&q.parser=dismax&q.options={fields:['title', 'description', 'review'], tieBreaker: 0.1}
```

Jika dog muncul di ketiga bidang dokumen dan skor untuk setiap bidang adalah title=1, description=3, dan review=1, skor keseluruhan untuk istilah dog adalah:

$$3 + 0.1 * (1+1) = 3.2$$

Setel tieBreaker ke 0 untuk mengabaikan semua kecuali bidang skor tertinggi (maks murni). Setel ke 1 untuk menjumlahkan skor dari semua bidang (jumlah murni). Nilai yang valid: 0,0 hingga 1,0. Default: 0.0. Berlaku untuk:dismax.

Jenis: objek JSON

Default: Lihat deskripsi opsi individual.

Diperlukan: Tidak

q.parser

Menentukan parser query yang akan digunakan untuk memproses permintaan:simple,, structuredlucene, dan. dismax Jika tidak q.parser ditentukan, Amazon CloudSearch menggunakan parser simple kueri.

- **simple**—melakukan pencarian sederhana text dan text-array bidang. Secara default, parser simple kueri mencari semua bidang dan dikonfigurasi text secara statis. text-array Anda dapat menentukan bidang mana yang akan dicari dengan q.options parameter. Jika Anda mengawali istilah pencarian dengan tanda tambah (+) dokumen harus berisi istilah yang akan dianggap cocok. (Ini adalah default, kecuali jika Anda mengkonfigurasi operator default dengan q.options parameter.) Anda dapat menggunakan operator - (NOT), | (OR), dan \* (wildcard) untuk mengecualikan istilah tertentu, menemukan hasil yang cocok dengan salah satu istilah yang ditentukan, atau mencari awalan. Untuk mencari frasa daripada istilah

individual, lampirkan frasa dalam tanda kutip ganda. Untuk informasi selengkapnya, lihat [Mencari Data Anda dengan Amazon CloudSearch](#).

- `structured`—melakukan pencarian lanjutan dengan menggabungkan beberapa ekspresi untuk menentukan kriteria pencarian. Anda juga dapat mencari dalam bidang tertentu, mencari nilai dan rentang nilai, dan menggunakan opsi lanjutan seperti peningkatan istilah, `matchAll`, dan `near`. Untuk informasi selengkapnya, lihat [Membangun Kueri Senyawa](#).
- `lucene`—cari menggunakan sintaks parser kueri Apache Lucene. Untuk informasi selengkapnya, lihat [Apache Lucene Query Parser Syntax](#).
- `dismax`—search menggunakan subset sederhana dari sintaks parser kueri Apache Lucene yang ditentukan oleh parser kueri. DisMax Untuk informasi selengkapnya, lihat [Sintaks Pengurai DisMax Kueri](#).

Jenis: String

Default: `simple`

Diperlukan: Tidak

return

Nilai bidang dan ekspresi untuk disertakan dalam respons, ditentukan sebagai daftar yang dipisahkan koma. Secara default, respons penelusuran mencakup semua bidang yang diaktifkan kembali (`return=_all_fields`). Untuk mengembalikan hanya ID dokumen untuk dokumen yang cocok, tentukan `return=_no_fields`. Untuk mengambil skor relevansi yang dihitung untuk setiap dokumen, tentukan `return=_score` Anda menentukan beberapa bidang kembali sebagai daftar dipisahkan koma. Misalnya, `return=title,_score` mengembalikan hanya judul dan skor relevansi dari setiap dokumen yang cocok.

Tipe: String

Wajib: Tidak

size

Jumlah maksimum klik pencarian untuk kembali.

Tipe: Bilangan bulat positif

Default: 10

Diperlukan: Tidak

## sort

Daftar kolom atau ekspresi kustom yang dipisahkan koma untuk digunakan untuk mengurutkan hasil pencarian. Anda harus menentukan arah pengurutan (`asc` atau `desc`) untuk setiap bidang. Misalnya, `sort=year desc,title asc`. Anda dapat menentukan maksimal 10 bidang dan ekspresi. Untuk menggunakan bidang untuk mengurutkan hasil, itu harus diurutkan diaktifkan dalam konfigurasi domain. Bidang tipe array tidak dapat digunakan untuk menyortir. Jika tidak ada `sort` parameter yang ditentukan, hasil diurutkan berdasarkan skor relevansi defaultnya dalam urutan menurun: `sort=_score desc`. Anda juga dapat mengurutkan berdasarkan dokumen ID (`sort=_id`) dan versi (`sort=_version`).

Tipe: String

Wajib: Tidak

## start

Offset dari hit pencarian pertama yang ingin Anda kembalikan. Anda dapat menentukan `cursor` parameter `start` atau dalam permintaan, keduanya saling eksklusif. Untuk informasi selengkapnya, lihat [Paginate the results](#).

Tipe: Bilangan bulat positif

Default: 0 (hit pertama)

Diperlukan: Tidak

## Sintaks Pencarian Terstruktur

Anda menggunakan sintaks penelusuran CloudSearch terstruktur Amazon untuk menentukan kriteria penelusuran saat menggunakan parser `structured` kueri, dan untuk menentukan kriteria filter dengan parameter. `fq`

Saat menggunakan operator kueri terstruktur, Anda menentukan nama operator, opsi untuk operator, dan kemudian istilah yang dioperasikan, (`OPERATOR OPTIONS STRING|EXPRESSION`). Opsi apa pun harus ditentukan sebelum string atau ekspresi. Misalnya, `(and (not field=genres 'Sci-Fi'))(or (term field=title boost=2 'star')(term field=plot 'star'))`.

### Important

Anda harus mengkodekan URL karakter khusus dalam string kueri. Misalnya, Anda harus menyandikan `=` operator dalam kueri terstruktur sebagai `%3D:(term+field%3Dtitle`

+ 'star'). Amazon CloudSearch mengembalikan `InvalidQueryString` kesalahan jika karakter khusus tidak dikodekan URL. [Untuk referensi lengkap pengkodean URL, lihat Referensi Pengkodean URL HTML W3C.](#)

Jika Anda tidak menentukan bidang yang ingin Anda cari saat menggunakan parser kueri terstruktur, semua `text-array` bidang `text` dan dikonfigurasi secara statis akan dicari. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari secara default dengan `q.options` parameter.

Tanda kurung mengontrol urutan evaluasi ekspresi dalam kueri majemuk. Ketika ekspresi diapit dalam tanda kurung, ekspresi itu dievaluasi terlebih dahulu, dan kemudian nilai yang dihasilkan digunakan dalam evaluasi sisa kueri. Ekspresi dapat berisi salah satu operator kueri terstruktur.

Anda juga dapat menggunakan parser kueri terstruktur untuk mencari string teks sederhana —cukup lampirkan string yang ingin Anda cari dalam tanda kutip tunggal: `q='black swan'&q.parser="structured"`

Untuk informasi selengkapnya tentang membangun kueri majemuk dengan operator kueri terstruktur, lihat [Membangun Kueri Senyawa](#)

## FIELD

Sintaks: `FIELD: 'STRING' |value`

Mencari bidang tertentu untuk string, nilai numerik, tanggal, atau rentang nilai numerik atau tanggal.

String harus dilampirkan dalam tanda kutip tunggal. Setiap tanda kutip tunggal atau garis miring terbalik dalam string harus lolos dengan garis miring terbalik. Untuk menentukan rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Braket persegi, [atau], menunjukkan bahwa ikatan termasuk dalam rentang, penjepit keriting, {atau}, tidak termasuk batas. Anda dapat menghilangkan batas atas atau bawah untuk menentukan rentang terbuka. Saat menghilangkan ikatan, Anda harus menggunakan penjepit keriting.

Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut [IETF RFC3339](#): `yyyy-mm-ddTHH:mm:ss.SSSZ` Di UTC, misalnya, 17:00 23 Agustus 1970 adalah: `1970-08-23T17:00:00Z` Perhatikan bahwa Anda juga dapat menentukan detik pecahan saat menentukan waktu di UTC. Misalnya, `1967-01-31T23:20:50.650Z`.

**Contoh:**

```
title:'star'  
year:2000  
year:[1998,2000]  
year:{,2011}  
release_date:['2013-01-01T00:00:00Z',]
```

**and**

Sintaks: (and boost=N EXPRESSION EXPRESSION ... EXPRESSIONn)

Termasuk dokumen hanya jika cocok dengan semua ekspresi yang ditentukan. (AND Operator Boolean.) Ekspresi dapat berisi salah satu operator kueri terstruktur, atau string pencarian sederhana. String pencarian harus dilampirkan dalam tanda kutip tunggal. Perhatikan bahwa untuk mencocokkan dokumen yang berisi istilah tertentu di salah satu bidang yang dicari, Anda menentukan setiap istilah sebagai ekspresi terpisah: (and 'star' 'wars'). Jika Anda menentukan (and 'star wars'), bintang dan perang harus terjadi dalam bidang yang sama untuk dianggap cocok.

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

**Contoh:**

```
(and title:'star' actors:'Harrison Ford' year:{,2000])
```

**matchall**

Sintaks: matchall

Cocokkan setiap dokumen di domain. Secara default, mengembalikan 10 pertama. Gunakan start parameter size dan ke halaman melalui hasil.

**near**

Sintaks: (near field=FIELD distance=N boost=N 'STRING')

Mencari text atau text-array bidang untuk string multi-term yang ditentukan dan mencocokkan dokumen yang berisi istilah dalam jarak yang ditentukan satu sama lain. (Ini kadang-kadang disebut pencarian frase ceroboh.) Jika Anda menghilangkan field opsi, Amazon

CloudSearch mencari semua yang dikonfigurasi secara statis `text` dan `text-array` bidang secara default. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari secara default dengan menentukan `q.options.fields` opsi.

Nilai jarak harus berupa bilangan bulat positif. Misalnya, untuk menemukan semua dokumen di mana remaja terjadi dalam 10 kata vampir di plot lapangan, Anda menentukan nilai jarak 10:.  
(`near field=plot distance=10 'teenage vampire'`)

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

Contoh:

```
(near field=plot distance=10 'teenage vampire')
```

not

Sintaks: (`not boost=N EXPRESSION`)

Mengecualikan dokumen jika cocok dengan ekspresi yang ditentukan. (NOT Operator Boolean.) Ekspresi dapat berisi salah satu operator kueri terstruktur, atau string pencarian sederhana. String pencarian harus dilampirkan dalam tanda kutip tunggal.

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

Contoh:

```
(not (or actors:'Harrison Ford' year:{,2010}))
```

or

Sintaks: (`or boost=N EXPRESSION1 EXPRESSION2 ... EXPRESSIONn`)

Termasuk dokumen jika cocok dengan salah satu ekspresi yang ditentukan. (OR Operator Boolean.) Ekspresi dapat berisi salah satu operator kueri terstruktur, atau string pencarian sederhana. String pencarian harus dilampirkan dalam tanda kutip tunggal.

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

**Contoh:**

```
(or actors:'Alec Guinness' actors:'Harrison Ford' actors:'James Earl Jones')
```

**phrase**

Sintaks: `(phrase field=FIELD boost=N 'STRING')`

Mencari `text` atau `text-array` bidang untuk frasa yang ditentukan. Jika Anda menghilangkan `field` opsi, Amazon CloudSearch mencari semua yang dikonfigurasi secara statis `text` dan `text-array` bidang secara default. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari secara default dengan menentukan `q.options fields` opsi.

Gunakan `phrase` operator untuk menggabungkan pencarian frase dengan kriteria penelusuran lainnya dalam kueri terstruktur. Misalnya `q=(and (term field=title 'star') (range field=year {,2000}))` cocok dengan semua dokumen yang berisi bintang di bidang judul dan memiliki nilai tahun kurang dari atau sama dengan 2000.

Nilai `boost` adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

**Contoh:**

```
(phrase field=plot 'teenage girl')
```

**prefix**


Sintaks: `(prefix field=FIELD boost=N 'STRING')`

Mencari `text`, `text-array`, `literal`, atau `literal-array` bidang untuk awalan yang ditentukan diikuti oleh nol atau lebih karakter. Jika Anda menghilangkan `field` opsi, Amazon CloudSearch mencari semua yang dikonfigurasi secara statis `text` dan `text-array` bidang secara default. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari secara default dengan menentukan `q.options fields` opsi.

Gunakan `prefix` operator untuk menggabungkan penelusuran awalan dengan kriteria penelusuran lainnya dalam kueri terstruktur. Misalnya, `q=(and (prefix field=title`

'sta') (range field=year {,2000})) mencocokkan semua dokumen yang berisi awalan sta di bidang judul dan memiliki nilai tahun kurang dari atau sama dengan 2000.

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

 Note

Untuk menerapkan saran penelusuran, Anda harus mengonfigurasi dan meminta sugester, daripada melakukan pencarian awalan. Untuk informasi selengkapnya, lihat [Permintaan Saran](#).

Contoh:

```
(prefix field=title 'star')
```

range

Sintaks: (range field=FIELD boost=N RANGE)

Mencari bidang numerik (double, double-array, int, int-array) atau bidang tanggal (tanggal, tanggal-array) untuk nilai dalam rentang yang ditentukan. Cocokkan dokumen yang memiliki setidaknya satu nilai di bidang dalam rentang yang ditentukan. fieldOpsi harus ditentukan.

Gunakan range operator untuk menggabungkan pencarian rentang dengan kriteria penelusuran lainnya dalam kueri terstruktur. Misalnya q=(and (term field=title 'star') (range field=year {,2000})) cocok dengan semua dokumen yang berisi bintang di bidang judul dan memiliki nilai tahun kurang dari atau sama dengan 2000.

Untuk menentukan rentang nilai, gunakan koma (,) untuk memisahkan batas atas dan bawah dan lampirkan rentang menggunakan tanda kurung atau tanda kurung. Braket persegi, [atau], menunjukkan bahwa ikatan termasuk dalam rentang, penjepit keriting, {atau}, tidak termasuk batas. Anda dapat menghilangkan batas atas atau bawah untuk menentukan rentang terbuka. Saat menghilangkan ikatan, Anda harus menggunakan penjepit keriting.

Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut [IETF RFC3339](#): yyyy-mm-ddTHH:mm:ss.SSSZ Di UTC, misalnya, 17:00 23 Agustus 1970 adalah: 1970-08-23T17:00:00Z Perhatikan bahwa Anda juga dapat menentukan detik pecahan saat menentukan waktu di UTC. Misalnya, 1967-01-31T23:20:50.650Z.



Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

Contoh:

```
(range field=year [1990,2000])  
(range field=year {,2000})  
(range field=year [1990,])
```

## term

Sintaks: `(term field=FIELD boost=N 'STRING' |VALUE)`

Mencari bidang tertentu untuk string, nilai numerik, atau tanggal. `field` opsi harus ditentukan saat mencari nilai. Jika Anda menghilangkan `field` opsi, Amazon CloudSearch mencari semua yang dikonfigurasi secara statis `text` dan `text-array` bidang secara default. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menentukan bidang mana yang ingin Anda cari secara default dengan menentukan `q.options fields` opsi.

Gunakan `term` operator untuk menggabungkan pencarian istilah dengan kriteria penelusuran lainnya dalam kueri terstruktur. Misalnya, `q=(and (term field=title 'star') (range field=year {,2000}))` cocok dengan semua dokumen yang berisi bintang di bidang judul dan memiliki nilai tahun kurang dari atau sama dengan 2000.

String dan tanggal harus dilampirkan dalam tanda kutip tunggal. Setiap tanda kutip tunggal atau garis miring terbalik dalam string harus lolos dengan garis miring terbalik.

Tanggal dan waktu ditentukan dalam UTC (Coordinated Universal Time) menurut [IETF RFC3339](#): `yyyy-mm-ddTHH:mm:ss.SSSZ` Di UTC, misalnya, 17:00 23 Agustus 1970 adalah: `1970-08-23T17:00:00Z` Perhatikan bahwa Anda juga dapat menentukan detik pecahan saat menentukan waktu di UTC. Misalnya, `1967-01-31T23:20:50.650Z`.

Nilai boost adalah nilai numerik positif yang meningkatkan pentingnya bagian kueri penelusuran ini relatif terhadap bagian lain.

Contoh:

```
(term field=title 'star')  
(term field=year 2000)
```

## Sintaks Pencarian Sederhana

Anda menggunakan sintaks penelusuran CloudSearch sederhana Amazon untuk menentukan kriteria penelusuran saat menggunakan parser `simple` kueri. Parser query sederhana digunakan secara default jika Anda tidak menentukan `q.parser` parameter.

Anda menggunakan parser kueri sederhana untuk mencari istilah atau frasa individual. Secara default, semua dikonfigurasi secara `static text` dan `text-array` bidang dicari. Bidang dan `literal` bidang dinamis tidak dicari secara default. Anda dapat menggunakan `q.options` parameter untuk menentukan bidang mana yang ingin Anda cari, mengubah operator default yang digunakan untuk menggabungkan istilah individual dalam string pencarian, atau menonaktifkan salah satu operator parser sederhana (`and`, `escape`, `fuzzy`, `near`, `not`, `or`, `phrase`, `precedenceprefix`, `whitespace`).

Untuk informasi selengkapnya tentang menggunakan parser kueri sederhana, lihat [text](#).

### + (and)

Sintaks: `+TERM`

Membutuhkan istilah yang ditentukan. Untuk mencocokkan, dokumen harus berisi istilah yang ditentukan.

Contoh: `+bintang`

### \ (escape)

Sintaks: `\CHAR`

Melarikan diri dari karakter khusus yang ingin Anda cari. Anda harus melarikan diri dari karakter berikut jika Anda ingin mereka menjadi bagian dari kueri: `+ - & |! () {} [] ^ "~ *? : \ /`.

Contoh: `M\*A\*S\*H`

### ~ (fuzzy)

Sintaks: `TERM~N`

Melakukan pencarian fuzzy. Tambahkan operator `~` dan nilai ke istilah untuk menunjukkan berapa banyak istilah yang dapat berbeda dan masih dianggap cocok.

Contoh: `stor~1`

## ~ (near)

Sintaks: "PHRASE"~N

Melakukan pencarian frase yang ceroboh. Tambahkan operator ~ dan nilai ke frasa untuk menunjukkan seberapa jauh jarak istilah tersebut dan masih dianggap cocok untuk frasa tersebut.

Contoh: "star wars"~4

## - (not)

Sintaks: -TERM

Melarang istilah yang ditentukan. Untuk mencocokkan, dokumen tidak boleh mengandung istilah.

Contoh: star -wars

## | (or)

Sintaks: |TERM

Membuat istilah yang ditentukan opsional.

Contoh: star |wars

## "..." (phrase)

Sintaks: "PHRASE"

Melakukan pencarian untuk seluruh frasa. Dapat dikombinasikan dengan ~ operator untuk melakukan pencarian frase ceroboh.

Contoh: "perang bintang"

## (...) (precedence)

Sintaks: ( . . . )

Mengontrol urutan di mana kendala kueri dievaluasi. Isi tanda kurung paling dalam dievaluasi terlebih dahulu.

Contoh: +(war|trek)+star

## \* (prefix)

Sintaks: CHARS\*

Cocokkan dokumen yang berisi istilah yang memiliki awalan yang ditentukan.

Contoh: sta\*

## Respon Pencarian

Ketika permintaan berhasil diselesaikan, badan respons berisi hasil pencarian. Secara default, hasil pencarian dikembalikan dalam JSON. Jika format parameter diatur ke xml, hasil pencarian dikembalikan dalam XHTML.

Kecuali Anda secara eksplisit menentukan `return` parameter, ID dokumen dan semua bidang yang dapat dikembalikan disertakan untuk setiap dokumen yang cocok (hit). Respons juga menunjukkan jumlah total hit found (`found`) dan indeks dokumen pertama yang terdaftar (`start`). Secara default, respons berisi 10 klik pertama. Anda menentukan `size` parameter dalam permintaan Anda untuk mengontrol berapa banyak klik yang disertakan dalam setiap respons. Untuk halaman melalui hits, Anda dapat menggunakan `cursor` parameter `start` or. Untuk informasi selengkapnya, lihat [Paginate the results](#).

Contoh berikut menunjukkan respons JSON yang khas.

```
{
  "status": {
    "rid": "rtKz7rkoeAojlvk=",
    "time-ms": 10
  },
  "hits": {
    "found": 3,
    "start": 0,
    "hit": [
      {
        "id": "tt1142977",
        "fields": {
          "rating": "6.9",
          "genres": [
            "Animation",
            "Comedy",
            "Family",
            "Horror",
            "Sci-Fi"
          ],
          "plot": "Young Victor conducts a science experiment to
            bring his beloved dog Sparky back to life, only
            to face unintended, sometimes monstrous,
```

```

        consequences.",
        "release_date": "2012-09-20T00:00:00Z",
        "title": "Frankenweenie",
        "rank": "1462",
        "running_time_secs": "5220",
        "directors": [
            "Tim Burton"
        ],
        "image_url": "http://ia.media-imdb.com/images/M/MV5BMjIx
            ODY3MjEwNV5BMl5BanBnXkFtZTcwOTMzNjc4Nw@@._
            V1_SX400_.jpg",
        "year": "2012",
        "actors": [
            "Winona Ryder",
            "Catherine O'Hara",
            "Martin Short"
        ]
    },
    .
    .
    .
    ]
}
}

```

Contoh berikut menunjukkan respon XMLnya setara.

```

<results>
  <status rid="itzL7rkoeQojlvk=" time-ms="34"/>
  <hits found="3" start="0">
    <hit id="tt1142977">
      <field name="rating">6.9</field>
      <field name="genres">Animation</field>
      <field name="genres">Comedy</field>
      <field name="genres">Family</field>
      <field name="genres">Horror</field>
      <field name="genres">Sci-Fi</field>
      <field name="plot">Young Victor conducts a science experiment to
        bring his beloved dog Sparky back to life, only
        to face unintended, sometimes monstrous,
        consequences.
      </field>
    </hit>
  </hits>
</results>

```

```
<field name="release_date">2012-09-20T00:00:00Z</field>
<field name="title">Frankenweenie</field>
<field name="rank">1462</field>
<field name="running_time_secs">5220</field>
<field name="directors">Tim Burton</field>
<field name="image_url">http://ia.media-imdb.com/images/M/MV5BMjIxODY3MjEwNV5BMl5BanBnXkFtZTcwOTMzNjc4Nw@@.
_V1_SX400_.jpg
</field>
<field name="year">2012</field>
<field name="actors">Winona Ryder</field>
<field name="actors">Catherine O'Hara</field>
<field name="actors">Martin Short</field>
</hit>
.
.
.
</hits>
</results>
```

Menyetel format respons hanya memengaruhi respons terhadap permintaan yang berhasil. Format respons kesalahan tergantung pada asal kesalahan. Kesalahan yang dikembalikan oleh layanan pencarian selalu dikembalikan di JSON. Kesalahan 5xx karena batas waktu server dan masalah perutean permintaan lainnya dikembalikan dalam XHTML. Ketika permintaan mengembalikan kode kesalahan, isi respons berisi informasi tentang kesalahan yang terjadi. Jika terjadi kesalahan saat badan permintaan diurai dan divalidasi, kode kesalahan disetel ke 400 dan badan respons menyertakan daftar kesalahan dan di mana mereka terjadi.

## Cari Header Respons

### Content-Type

Tipe MIME standar yang menjelaskan format data objek. Untuk informasi lebih lanjut, lihat [W3C RFC 2616](#) Bagian 14.

Nilai yang valid: `aplikasi/json` atau `aplikasi/xml`

Default: `aplikasi/json`

### Content-Length

Panjang dalam byte tubuh dalam respon.

## Cari Properti Respon (JSON)

### status

Berisi id sumber daya (`rid`) dan waktu yang dibutuhkan untuk memproses permintaan (`time-ms`).

#### rid

ID Sumber Daya terenkripsi.

#### time-ms

Berapa lama waktu yang dibutuhkan untuk memproses permintaan pencarian dalam milidetik.

### hits

Berisi jumlah dokumen yang cocok (`found`), indeks dokumen pertama yang disertakan dalam response (`start`), dan array (`hit`) yang mencantumkan ID dokumen dan data untuk setiap klik.

#### found

Jumlah total klik yang cocok dengan permintaan pencarian setelah Amazon CloudSearch selesai memproses permintaan.

#### start

Indeks hit pertama kembali dalam respons ini.

#### hit

Array yang mencantumkan ID dokumen dan data untuk setiap hit.

#### id

Pengidentifikasi unik untuk dokumen.

#### fields

Daftar bidang yang dikembalikan.

#### facets

Berisi informasi faset dan jumlah faset.

#### FACETFIELD

Bidang yang aspeknya dihitung.

#### buckets

Array dari nilai dan hitungan faset yang dihitung.

**value**

Nilai faset yang dihitung.

**count**

Jumlah hit yang berisi nilai faset diFACETFIELD.

**Elemen Respon Pencarian (XHTML)****results**

Berisi hasil pencarian. Setiap kesalahan yang terjadi saat memproses permintaan dikembalikan sebagai pesan dalam elemen info.

**status**

Berisi id sumber daya (`rid`) dan waktu yang dibutuhkan untuk memproses request (`time-ms`).

**hits**

Berisi statistik hit dan kumpulan elemen hit. Atribut yang ditemukan adalah jumlah total klik yang cocok dengan permintaan pencarian setelah Amazon CloudSearch selesai memproses hasilnya. Elemen hit yang terkandung diurutkan sesuai dengan skor relevansinya atau `sort` opsi yang ditentukan dalam permintaan pencarian.

**hit**

Dokumen yang cocok dengan permintaan pencarian. Atribut `id` adalah id unik dokumen. Berisi elemen `d` (data) untuk setiap bidang yang dikembalikan.

**field**

Sebuah lapangan kembali dari pukulan. Elemen hit berisi elemen `d` (data) untuk setiap bidang yang dikembalikan.

**facets**

Berisi elemen faset untuk setiap aspek yang diminta dalam permintaan pencarian.

**facet**

Berisi elemen bucket untuk setiap nilai bidang yang dihitung hitungannya. Opsi `facet.FIELD` ukuran dapat digunakan untuk menentukan berapa banyak kendala yang akan dikembalikan. Secara default, jumlah faset dikembalikan untuk



10 batasan teratas. Opsi facet .FIELD bucket dapat digunakan untuk secara eksplisit menentukan nilai mana yang akan dihitung.

bucket

Nilai bidang facet dan jumlah kemunculan (hitungan) dari nilai tersebut dalam hit pencarian.

## Mengirimkan Permintaan Saran di Amazon CloudSearch

Anda mengirimkan permintaan saran melalui HTTP GET ke titik akhir pencarian domain Anda di `2013-01-01/suggest`. Untuk informasi tentang mengontrol akses ke layanan saran, lihat [configure access policies](#).

Anda harus menentukan versi API di semua permintaan saran dan versi tersebut harus cocok dengan versi API yang ditentukan saat domain dibuat.

Misalnya, permintaan berikut mendapat saran dari `search-movies-rr2f34ofg56xneuemujamut52i.us-east-1.cloudsearch.amazonaws.com` domain untuk string kueri `oce` menggunakan sugester yang dipanggil `title`.

```
http://search-imdb-hd6ebyouhw21czkueyuqksnuzu.us-west-2.cloudsearch.amazonaws.com/2013-01-01/suggest -d"q=oce&sugester=suggest_title"
```

Anda dapat menggunakan metode apa pun yang ingin Anda kirim permintaan GET ke titik akhir pencarian domain Anda—Anda dapat memasukkan URL permintaan secara langsung di browser Web, menggunakan cURL untuk mengirimkan permintaan, atau membuat panggilan HTTP menggunakan pustaka HTTP favorit Anda. Anda juga dapat menggunakan Search Tester di CloudSearch konsol Amazon untuk mendapatkan saran. Untuk informasi selengkapnya, lihat [Mencari dengan Search Tester](#).

### Important

Dokumen domain dan titik akhir pencarian tetap sama untuk masa pakai domain. Anda harus men-cache titik akhir daripada mengambilnya sebelum setiap permintaan unggahan atau pencarian. Menanyakan layanan CloudSearch konfigurasi Amazon dengan menelepon `aws cloudsearch describe-domains` atau `DescribeDomains` sebelum setiap permintaan kemungkinan akan mengakibatkan permintaan Anda dibatasi.

Secara default, Amazon CloudSearch mengembalikan respons di JSON. Anda bisa mendapatkan hasil yang diformat dalam XHTML dengan menentukan parameter, `format`. `format=xml` Menyetel format respons hanya memengaruhi respons terhadap permintaan yang berhasil. Format respons kesalahan tergantung pada asal kesalahan. Kesalahan yang dikembalikan oleh layanan pencarian selalu dikembalikan di JSON. Kesalahan 5xx karena batas waktu server dan masalah perutean permintaan lainnya dikembalikan dalam XHTML.

## Sarankan

### Permintaan Saran

#### Sarankan Sintaks di Amazon CloudSearch

```
GET /2013-01-01/suggest
```

#### Sarankan Header Permintaan di Amazon CloudSearch

##### HOST

Titik akhir permintaan pencarian untuk domain yang Anda kueri. Anda dapat menggunakan [DescribeDomains](#) untuk mengambil titik akhir permintaan pencarian domain Anda.

Diperlukan: Ya

#### Sarankan Parameter Permintaan di Amazon CloudSearch

##### q

String untuk mendapatkan saran untuk.

Tipe: String

Diperlukan: Ya

##### suggester

Nama suggester untuk digunakan untuk menemukan kecocokan yang disarankan.

Tipe: String

Diperlukan: Ya

## size

Jumlah maksimum saran untuk dikembalikan.

Tipe: Bilangan bulat positif

Default: 10

Diperlukan: Tidak

## format

Menentukan jenis konten respon.

Jenis: String

Nilai yang Valid: json|xml

Default: json

Diperlukan: Tidak

## Sarankan Tanggapan

Ketika permintaan berhasil diselesaikan, badan respons berisi saran. Secara default, saran dikembalikan dalam JSON. Atur format parameter `xml` untuk mendapatkan hasil dalam XHTML.

Menyetel format respons hanya memengaruhi respons terhadap permintaan yang berhasil. Format respons kesalahan tergantung pada asal kesalahan. Kesalahan yang dikembalikan oleh layanan pencarian selalu dikembalikan di JSON. Kesalahan 5xx karena batas waktu server dan masalah perutean permintaan lainnya dikembalikan dalam XHTML. Ketika permintaan mengembalikan kode kesalahan, isi respons berisi informasi tentang kesalahan yang terjadi. Jika terjadi kesalahan saat badan permintaan diurai dan divalidasi, kode kesalahan disetel ke 400 dan badan respons menyertakan daftar kesalahan dan di mana mereka terjadi.

Contoh berikut menunjukkan respons JSON terhadap permintaan saran:

```
{
  "status": {
    "rid": "q0SM5s0oCwr8pVk=",
    "time-ms": 2
  },
  "suggest": {
```

```
"query": "oce",
"found": 3,
"suggestions": [
  {
    "suggestion": "Ocean's Eleven",
    "score": 0,
    "id": "tt0054135"
  },
  {
    "suggestion": "Ocean's Thirteen",
    "score": 0,
    "id": "tt0496806"
  },
  {
    "suggestion": "Ocean's Twelve",
    "score": 0,
    "id": "tt0349903"
  }
]
}
```

Contoh berikut menunjukkan respon XMLnya setara:

```
<results>
  <status rid="/pSz580oDQr8pVk=" time-ms="2"/>
  <suggest query="oce" found="3">
    <suggestions>
      <item suggestion="Ocean's Eleven" score="0" id="tt0054135"/>
      <item suggestion="Ocean's Thirteen" score="0" id="tt0496806"/>
      <item suggestion="Ocean's Twelve" score="0" id="tt0349903"/>
    </suggestions>
  </suggest>
</results>
```

## Kesalahan Layanan Pencarian

Permintaan pencarian atau saran dapat mengembalikan tiga jenis kode status:

- Kode status 5xx menunjukkan bahwa ada kesalahan server internal. Anda harus menangkap dan mencoba kembali semua kode kesalahan 5xx karena biasanya mewakili kondisi kesalahan sementara. Untuk informasi selengkapnya, lihat [Penanganan Kesalahan](#).

- Kode status 4xx menunjukkan bahwa permintaan itu salah bentuk. Perbaiki kesalahan sebelum mengirimkan kembali permintaan Anda.
- Kode status 2xx menunjukkan bahwa permintaan berhasil diproses.

Format respons kesalahan tergantung pada asal kesalahan. Kesalahan yang dikembalikan oleh layanan pencarian selalu dikembalikan di JSON. Kesalahan 5xx karena batas waktu server dan masalah perutean permintaan lainnya dikembalikan dalam XHTML.

Kesalahan yang dikembalikan oleh layanan pencarian berisi informasi berikut:

`error`

    Berisi pesan kesalahan yang dikembalikan oleh layanan pencarian. `msgProperti code` dan disertakan untuk setiap kesalahan.

`code`

    Kode kesalahan.

`msg`

    Deskripsi kesalahan yang dikembalikan oleh layanan pencarian.

# Memecahkan Masalah Amazon CloudSearch

Topik berikut menjelaskan solusi untuk masalah yang mungkin Anda temui saat menggunakan Amazon CloudSearch.

## Topik

- [Mengunggah Dokumen](#)
- [Menghapus Semua Dokumen di Domain Amazon CloudSearch](#)
- [CloudSearch Domain Amazon Tidak Menurun Setelah Menghapus Dokumen](#)
- [Latensi Pembaruan Dokumen](#)
- [Sejumlah Besar Kesalahan 5xx Saat Mengunggah Dokumen ke Domain Amazon CloudSearch](#)
- [Cari Latensi dan Timeout di Amazon CloudSearch](#)
- [Cari Latensi untuk Query Faceted di Amazon CloudSearch](#)
- [Peningkatan Tiba-tiba dalam Kesalahan 5xx Saat Mencari Domain Amazon CloudSearch](#)
- [Kegagalan Pengindeksan setelah Memperbarui Opsi Pengindeksan di Amazon CloudSearch](#)
- [Domain Tidak Ditemukan Saat Mengirimkan Permintaan Amazon CloudSearch](#)
- [Jumlah Dokumen yang Dapat D dicari yang Tidak Dikembalikan dengan Informasi Domain](#)
- [Kebijakan Akses Layanan Konfigurasi Tidak Berfungsi di Amazon CloudSearch](#)
- [Kebijakan Akses Layanan Pencarian dan Dokumen Tidak Berfungsi di Amazon CloudSearch](#)
- [Kesalahan Izin CloudSearch Konsol Amazon](#)
- [Menggunakan Wildcard untuk Mencari Bidang Teks Tidak Menghasilkan Hasil yang Diharapkan](#)
- [Hasil yang Tidak Konsisten Saat Menggunakan Cursor untuk Deep Paging](#)
- [Kesalahan Sertifikat Saat Menggunakan SDK](#)

## Mengunggah Dokumen

Jika data dokumen Anda tidak diformat dengan benar atau berisi nilai yang tidak valid, Anda akan mendapatkan kesalahan saat mencoba mengunggahnya atau menggunakannya untuk mengonfigurasi bidang untuk domain Anda. Berikut adalah beberapa masalah umum dan solusinya:

- JSON tidak valid —jika Anda menggunakan JSON, hal pertama yang harus dilakukan adalah memastikan tidak ada kesalahan sintaks JSON dalam kumpulan dokumen Anda. Untuk melakukan

itu, jalankan melalui alat validasi seperti [JSON Validator](#). Ini akan mengidentifikasi masalah mendasar apa pun dengan data.

- Batch XML-dokumen yang tidak valid harus XML-dibentuk dengan baik. Anda kemungkinan besar akan mengalami masalah jika bidang Anda berisi data XML—data harus dikodekan XML atau terlampir di bagian CDATA. Untuk mengidentifikasi masalah apa pun, jalankan batch dokumen Anda melalui alat validasi seperti Layanan Validasi [Markup W3C](#).
- Tidak Diakui sebagai Batch Dokumen —jika Amazon CloudSearch tidak mengenali data Anda sebagai kumpulan dokumen yang valid saat Anda mengunggah data menggunakan konsol, Amazon akan CloudSearch menghasilkan kumpulan valid yang berisi satu bidang konten dan bidang metadata umum seperti `content_encoding`, dan `content_type` `resourcename`. Karena ini biasanya bukan bidang yang dikonfigurasi untuk domain, Anda mendapatkan kesalahan yang menyatakan bahwa bidang tersebut tidak ada. Demikian pula, jika Anda mencoba mengonfigurasi domain dari kumpulan yang tidak valid, Amazon CloudSearch merespons dengan bidang konten dan meta-data, bukan bidang dalam batch.

Pertama, pastikan bahwa batch adalah XHTML atau JSON yang valid. Jika ya, periksa ID dokumen yang tidak valid dan pastikan Anda telah menentukan jenis operasi untuk setiap dokumen. Untuk menambahkan operasi, pastikan bahwa jenis, ID, dan setidaknya satu bidang ditentukan untuk setiap dokumen. Hapus operasi hanya perlu menentukan jenis dan ID. Untuk informasi selengkapnya tentang memformat data Anda, lihat [Creating Document Batches](#).

- ID dokumen dengan nilai buruk —ID dokumen dapat berisi huruf atau angka dan karakter berikut: `_ - = #;:/? @ &`. ID dokumen harus minimal 1 dan panjangnya tidak lebih dari 128 karakter.
- Bidang multi-nilai tanpa nilai —saat menentukan data dokumen di JSON, Anda tidak dapat menentukan array kosong sebagai nilai bidang. Bidang multi-nilai harus berisi setidaknya satu nilai.
- Karakter buruk — satu masalah yang bisa sulit dideteksi jika Anda tidak memfilter data Anda saat membuat batch dokumen Anda adalah yang dapat berisi karakter yang tidak valid dalam XML.2. Batch JSON dan XHTML hanya dapat berisi karakter UTF-8 yang valid dalam XML.2. Anda dapat menggunakan alat validasi seperti [JSON Validator atau W3C Markup Validation Service](#) untuk mengidentifikasi karakter yang tidak valid.

## Menghapus Semua Dokumen di Domain Amazon CloudSearch

Amazon CloudSearch saat ini tidak menyediakan mekanisme untuk menghapus semua dokumen dalam domain.

# CloudSearch Domain Amazon Tidak Menurun Setelah Menghapus Dokumen

Jika domain Anda telah ditingkatkan untuk mengakomodasi ukuran indeks Anda dan Anda menghapus sejumlah besar dokumen, skala domain akan turun saat berikutnya indeks penuh dibangun kembali. Meskipun indeks secara otomatis dibangun kembali secara berkala, untuk menurunkan skala secepat mungkin Anda dapat secara eksplisit [menjalankan pengindeksan](#) ketika Anda selesai menghapus dokumen.

## Latensi Pembaruan Dokumen

Mengirim sejumlah besar batch dokumen tunggal dapat meningkatkan jumlah waktu yang dibutuhkan setiap dokumen untuk dapat dicari. Jika Anda memiliki lalu lintas pembaruan dalam jumlah besar, Anda perlu mengumpulkan pembaruan Anda. Sebaiknya gunakan ukuran batch yang mendekati batas 5 MB. Untuk informasi selengkapnya, lihat [Creating Document Batches](#).

Anda dapat memuat hingga 10.000 batch dokumen per hari (setiap 24 jam), dengan setiap ukuran batch hingga 5 MB. Memuat lebih banyak data per hari secara signifikan meningkatkan latensi pembaruan dokumen. Untuk mengurangi risiko ini, Anda dapat meningkatkan kapasitas pembaruan dengan memilih jenis instans yang lebih besar. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

## Sejumlah Besar Kesalahan 5xx Saat Mengunggah Dokumen ke Domain Amazon CloudSearch

Jika Anda memparalelkan unggahan dan domain Anda berada di instance search.small, Anda mungkin mengalami tingkat kesalahan 504 atau 507 yang sangat tinggi. Menyetel jenis instans yang diinginkan ke jenis instans yang lebih besar akan meningkatkan kapasitas pembaruan Anda dan mengurangi tingkat kesalahan. Untuk informasi selengkapnya tentang penanganan kesalahan 5xx, lihat [Penanganan Kesalahan](#). Untuk informasi tentang pra-skala domain Anda untuk meningkatkan kapasitas upload, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#)

## Cari Latensi dan Timeout di Amazon CloudSearch

Jika Anda mengalami waktu respons yang lambat, sering mengalami kesalahan server internal (biasanya kesalahan 507 atau 509), atau melihat jumlah jam instans yang dikonsumsi domain



pencarian Anda meningkat tanpa peningkatan substansial dalam volume data yang Anda cari, menyempurnakan permintaan pencarian Anda untuk mengurangi overhead pemrosesan dapat membantu. Untuk informasi selengkapnya, lihat [Menyetel Kinerja Permintaan Pencarian di Amazon CloudSearch](#). Meningkatkan jumlah replikasi yang diinginkan juga dapat mempercepat pemrosesan permintaan pencarian. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

Kesalahan 507 dan 509 biasanya menunjukkan bahwa layanan pencarian Anda kelebihan beban. Hal ini dapat disebabkan oleh volume atau kompleksitas permintaan pencarian yang Anda kirimkan. Amazon CloudSearch biasanya menskalakan secara otomatis untuk menangani beban. Karena membutuhkan waktu untuk menerapkan instance penelusuran tambahan, sebaiknya gunakan kebijakan coba ulang backoff eksponensial untuk mengurangi tingkat permintaan sementara dan meminimalkan kegagalan permintaan. Untuk informasi selengkapnya, lihat [Error Retries and Exponential Backoff](#).

Pola penggunaan tertentu, seperti mengirimkan kueri penelusuran yang kompleks ke satu contoh penelusuran kecil, terkadang dapat menghasilkan batas waktu tanpa memicu penskalaan otomatis. Jika Anda berulang kali mengalami tingkat kesalahan yang tinggi, Anda dapat secara eksplisit meminta kapasitas tambahan melalui formulir [Permintaan Batas CloudSearch Layanan](#) Amazon.

## Cari Latensi untuk Query Faceted di Amazon CloudSearch

Jika Anda mengumpulkan informasi facet dengan buckets opsi dan mengalami kinerja kueri yang lambat, coba atur metode bucket ke `interval`. Untuk informasi selengkapnya, lihat [Informasi Aspek Bucketing](#).

## Peningkatan Tiba-tiba dalam Kesalahan 5xx Saat Mencari Domain Amazon CloudSearch

Jika domain penelusuran Anda mengalami lonjakan lalu lintas yang tiba-tiba, Amazon CloudSearch merespons dengan menambahkan instance penelusuran ke domain Anda untuk menangani peningkatan beban. Namun, dibutuhkan beberapa menit untuk menyiapkan instance baru. Anda mungkin akan melihat peningkatan sementara dalam kesalahan 5xx hingga instance baru siap untuk mulai memproses permintaan. Untuk informasi selengkapnya tentang penanganan kesalahan 5xx, lihat [Penanganan Kesalahan](#). Untuk informasi tentang pra-penskalaan domain Anda guna menangani lonjakan permintaan penelusuran yang diharapkan, lihat [Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch](#).

# Kegagalan Pengindeksan setelah Memperbarui Opsi Pengindeksan di Amazon CloudSearch

Jika Anda membuat perubahan pada konfigurasi indeks domain, dalam kasus tertentu Anda mungkin mengalami kesalahan Gagal Memvalidasi saat menjalankan pengindeksan. Ini berarti bahwa opsi bidang indeks yang Anda tetapkan tidak kompatibel dengan dokumen yang sudah ada dalam indeks Anda. Secara khusus, Anda mengubah jenis bidang indeks, dan ada dokumen dalam indeks Anda yang berisi data yang tidak kompatibel dengan jenis itu. Misalnya, ini mungkin terjadi jika Anda mengubah `literal` bidang ke `int` bidang, dan beberapa dokumen Anda berisi data alfanumerik di bidang itu. Ketika ini terjadi, Amazon CloudSearch menetapkan status SEMUA bidang yang sedang diproses ke `FailedToValidate` status. Mengembalikan perubahan konfigurasi yang tidak kompatibel akan memungkinkan Anda untuk berhasil membangun kembali indeks Anda. Jika perubahan diperlukan, Anda harus memperbarui atau menghapus dokumen yang tidak kompatibel dari indeks Anda untuk menggunakan konfigurasi baru. Jika Anda tidak dapat mengidentifikasi perubahan yang menyebabkan kesalahan atau memerlukan bantuan untuk mengidentifikasi dokumen yang tidak kompatibel, hubungi dukungan.

## Domain Tidak Ditemukan Saat Mengirimkan Permintaan Amazon CloudSearch

Anda tidak dapat mengakses domain 2013-01-01 dengan alat baris perintah 2011-02-01 atau SDK. Demikian pula, Anda tidak dapat mengakses domain 2011-02-01 dengan alat baris perintah 2013-01-01 atau SDK. Pastikan Anda menentukan versi API yang benar dalam permintaan Anda dan menggunakan alat baris perintah atau SDK yang sesuai.

## Jumlah Dokumen yang Dapat Dicari yang Tidak Dikembalikan dengan Informasi Domain

Itu `aws cloudsearch describe-domains` dan `DescribeDomains` tidak mengembalikan jumlah dokumen yang dapat dicari di domain. Untuk mendapatkan jumlah dokumen yang dapat dicari, gunakan konsol atau kirimkan `matchall` permintaan ke titik akhir penelusuran domain Anda.

```
q=matchall&q.parser=structured&size=0
```

## Kebijakan Akses Layanan Konfigurasi Tidak Berfungsi di Amazon CloudSearch

Jika Anda memiliki domain 2011 dan 2013, telah mengonfigurasi kebijakan IAM untuk mengakses layanan konfigurasi, dan mendapatkan kesalahan yang tidak diotorisasi, perhatikan bahwa Amazon CloudSearch ARN berbeda untuk 2011-02-01 API dan 2013-01-01 API. Untuk memungkinkan pengguna mengakses domain 2011 dan 2013, Anda harus mengizinkan akses ke kedua ARN dalam kebijakan IAM. Sebagai contoh:

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudsearch:*",
      ],
      "Resource": "arn:aws:cloudsearch:*",
      "Resource": "arn:aws:cs:*"
    }
  ]
}
```

Jika kebijakan 2011 Anda memberikan akses ke domain atau tindakan tertentu, Anda harus menyertakan pembatasan tersebut dalam kebijakan Anda. Perhatikan bahwa satu-satunya tindakan yang didukung untuk domain 2011 adalah `cloudsearch:*` dan Anda mungkin mengalami kesalahan lain saat mencoba mengonfigurasi izin tingkat sumber daya untuk domain yang dibuat dengan API 2011-01-01.

## Kebijakan Akses Layanan Pencarian dan Dokumen Tidak Berfungsi di Amazon CloudSearch

Jika Anda telah mengonfigurasi kebijakan akses untuk titik akhir penelusuran dan layanan dokumen domain Anda, tetapi mendapatkan kesalahan 403 Permintaan yang dilarang oleh aturan administratif, kemungkinan karena salah satu masalah berikut.

- Pastikan versi API dan nama sumber daya ditentukan dalam permintaan Anda. Misalnya, untuk mengunggah dokumen dengan API 2013-01-01, Anda harus menambahkan `/2013-01-01/documents/batch` ke endpoint layanan dokumen domain Anda:

```
doc-movies-123456789012.us-east-1.cloudsearch.amazonaws.com/2013-01-01/documents/  
batch
```

Untuk mengirimkan permintaan penelusuran menggunakan API 2013-01-01, Anda harus menambahkan `/2013-01-01/search` ke titik akhir penelusuran domain Anda:

```
search-movies-123456789012.us-east-1.cloudsearch.amazonaws.com/2013-01-01/search?  
q=star+wars&return=title
```

Untuk mendapatkan saran menggunakan API 2013-01-01, Anda harus menambahkan ke titik akhir penelusuran domain `/2013-01-01/suggest` Anda:

```
search-movies-123456789012.us-east-1.cloudsearch.amazonaws.com/2013-01-01/suggest?  
q=kat&suggester=mysuggester
```

- Jika Anda terhubung dari instans EC2, pastikan kebijakan akses menentukan alamat IP publik instans EC2 Anda.
- Jika mesin yang Anda sambungkan berada di belakang router, pastikan kebijakan akses menentukan alamat IP yang menghadap publik Anda.

Untuk informasi selengkapnya, lihat [configure access policies](#).

## Kesalahan Izin CloudSearch Konsol Amazon

Untuk mengakses ke konsol, Anda harus memiliki izin untuk `DescribeDomains` tindakan tersebut. Akses ke domain dan tindakan tertentu mungkin dibatasi oleh kebijakan akses IAM yang dikonfigurasi. Selain itu, mengunggah data dari bucket Amazon S3 atau tabel DynamoDB memerlukan akses ke layanan dan sumber daya tersebut. Untuk informasi selengkapnya tentang kebijakan CloudSearch akses Amazon, lihat [configure access policies](#).

## Menggunakan Wildcard untuk Mencari Bidang Teks Tidak Menghasilkan Hasil yang Diharapkan

Saat Anda mengirimkan permintaan pencarian, teks yang Anda cari mengalami pemrosesan teks yang sama sehingga dapat dicocokkan dengan istilah yang muncul di indeks. Namun, tidak ada analisis teks yang dilakukan pada istilah pencarian saat Anda melakukan pencarian awalan. Ini

berarti bahwa pencarian awalan yang diakhiri s biasanya tidak akan cocok dengan versi tunggal istilah saat stemming diaktifkan. Ini bisa terjadi untuk istilah apa pun yang berakhir, bukan hanya bentuk jamak. Misalnya, jika Anda mencari `actor` bidang dalam data film sampel `Anders`, ada tiga film yang cocok. Jika Anda mencari `Ande*`, Anda mendapatkan film-film itu serta beberapa lainnya. Namun, jika Anda mencari tidak `Anders*` ada kecocokan. Ini karena istilah disimpan dalam indeks sebagai `ander`, `anders` tidak muncul dalam indeks.

Jika stemming mencegah pencarian wildcard mengembalikan semua kecocokan yang relevan, Anda dapat menekan stemming untuk bidang teks dengan menyetel `AlgorithmicStemming` opsi ke `none`, atau Anda dapat memetakan data ke bidang, bukan bidang. `literal text`

Untuk informasi selengkapnya tentang cara Amazon CloudSearch memproses teks, lihat [Pemrosesan Teks di Amazon CloudSearch](#).

## Hasil yang Tidak Konsisten Saat Menggunakan Cursor untuk Deep Paging

Saat Anda menggunakan cursor ke halaman melalui kumpulan hasil yang diurutkan berdasarkan skor dokumen (`_score`), Anda bisa mendapatkan hasil yang tidak konsisten jika indeks diperbarui di antara permintaan. Ini juga dapat terjadi jika jumlah replikasi domain Anda lebih besar dari satu, karena pembaruan diterapkan secara konsisten di seluruh instance dalam domain. Jika ini menjadi masalah, hindari menyortir hasil berdasarkan skor. Anda dapat menggunakan `sort` opsi untuk mengurutkan berdasarkan bidang tertentu, atau menggunakan `fq` alih-alih `q` untuk menentukan kriteria pencarian Anda. (Skor dokumen tidak dihitung untuk kueri filter.)

## Kesalahan Sertifikat Saat Menggunakan SDK

Karena AWS SDK menggunakan sertifikat CA dari komputer Anda, perubahan pada sertifikat di server AWS dapat menyebabkan kegagalan koneksi saat Anda mencoba menggunakan SDK. Pesan kesalahan bervariasi, tetapi biasanya berisi teks berikut:

```
SSL3_GET_SERVER_CERTIFICATE:certificate verify failed
```

Anda dapat mencegah kegagalan ini dengan menyimpan sertifikat CA dan sistem operasi komputer Anda up-to-date. Jika Anda mengalami masalah ini di lingkungan perusahaan dan tidak mengelola komputer Anda sendiri, Anda mungkin perlu meminta administrator untuk membantu proses pembaruan.


Daftar berikut menunjukkan sistem operasi minimum dan versi Java:

- Versi Microsoft Windows yang memiliki pembaruan mulai Januari 2005 atau yang lebih baru yang sudah terinstal memuat setidaknya satu dari CA yang diperlukan dalam daftar kepercayaan mereka.
- Mac OS X 10.4 dengan Java untuk Mac OS X 10.4 Release 5 (Februari 2007), Mac OS X 10.5 (Oktober 2007), dan versi lebih baru memuat setidaknya satu dari CA yang diperlukan dalam daftar kepercayaan mereka.
- Red Hat Enterprise Linux 5 (Maret 2007), 6, dan 7 serta CentOS 5, 6, dan 7 semuanya berisi setidaknya satu dari CA yang diperlukan dalam daftar default CA tepercaya mereka.
- Java 1.4.2\_12 (Mei 2006), 5 Pembaruan 2 (Maret 2005), dan semua versi setelahnya, termasuk Java 6 (Desember 2006), 7, dan 8, memuat setidaknya satu dari CA yang diperlukan dalam daftar default CA tepercaya mereka.

Ketiga otoritas sertifikasi (CA) adalah:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certification Authority

Sertifikat root dari dua otoritas pertama tersedia dari [Amazon Trust Services](#), tetapi menjaga komputer Anda up-to-date adalah solusi yang lebih mudah. Untuk mempelajari lebih lanjut tentang sertifikat yang disediakan ACM, lihat [AWS Certificate Manager FAQ](#).

 Note

Sertifikat ini belum diperlukan, tetapi dijadwalkan untuk diterapkan ke server AWS pada bulan November 2017.

# Memahami CloudSearch Batas Amazon

Tabel ini menunjukkan batasan penamaan dan ukuran di Amazon CloudSearch. Anda dapat [mengirimkan permintaan](#) jika Anda perlu meningkatkan jumlah partisi maksimum untuk domain pencarian. Untuk informasi tentang meningkatkan batas lain seperti jumlah maksimum domain pencarian, hubungi Amazon CloudSearch.


CloudSearch Batas Amazon saat ini dirangkum dalam tabel berikut.

Item	Kuota
Ukuran batch	Ukuran batch maksimum adalah 5 MB.
Volume pemuatan data	<p>Anda dapat memuat satu batch dokumen setiap 10 detik (sekitar 10.000 batch setiap 24 jam), dengan setiap ukuran batch hingga 5 MB.</p> <p>Melebihi batas ini secara signifikan meningkatkan latensi pembaruan dokumen dan dapat mengakibatkan pelambatan. Untuk mengurangi risiko ini, Anda dapat meningkatkan kapasitas pembaruan dengan memilih jenis instans yang lebih besar. Untuk informasi selengkapnya, lihat <a href="#">Creating Document Batches</a>.</p> <div data-bbox="695 1276 1507 1789" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px;"><p><b>⚠ Important</b></p><p>Apa pun jenis instans yang Anda pilih, Amazon CloudSearch tidak menjamin pemesanan dokumen yang diterima di detik yang sama. Misalnya, jika Anda mengirim tiga pembaruan dengan sepersepuluh detik di antara mereka, pembaruan terakhir mungkin bukan yang terakhir diterapkan. Mempertahankan urutan pembaruan adalah alasan lain untuk mematuhi batas ini.</p></div>
Ukuran dokumen	Ukuran dokumen maksimum adalah 1 MB.

Item	Kuota
Bidang dokumen	Dokumen dapat memiliki tidak lebih dari 200 bidang.
Ekspresi	<ul style="list-style-type: none"><li>• Hingga 50 ekspresi dapat dikonfigurasi untuk domain.</li><li>• Ukuran maksimum ekspresi adalah 10240 byte.</li><li>• Nilai maksimum yang dapat dikembalikan oleh ekspresi adalah <code>max (int64_t)</code>.</li></ul>
Menyoroti	<ul style="list-style-type: none"><li>• Jumlah maksimum kemunculan istilah pencarian yang dapat disorot adalah 5.</li><li>• Sorotan hanya dikembalikan untuk 10 KB data pertama di bidang teks.</li></ul>
Bidang indeks	<ul style="list-style-type: none"><li>• Hingga 200 bidang indeks dapat dikonfigurasi untuk domain. Bidang dinamis dihitung sebagai satu bidang indeks, tetapi biasanya cocok dengan beberapa bidang dokumen. Bidang dinamis dapat menyebabkan jumlah total bidang dalam indeks Anda melebihi 200. Jika Anda menggunakan bidang dinamis, pertahankan jumlah kolom indeks di bawah 1.000 untuk menghindari masalah kinerja.</li><li>• Hingga 1000 nilai dapat ditentukan dalam bidang.</li><li>• Hingga 20 sumber dapat ditentukan untuk bidang tipe array.</li><li>• Ukuran maksimum bidang literal adalah 4096 poin kode UTF-8.</li><li>• Ukuran maksimum nilai default untuk bidang adalah 1 KB.</li><li>• Bidang int dapat berisi nilai dari -9.223.372.036,854.775.808 hingga 9.223.372.036,854.775.807 (inklusif).</li><li>• Istilah individual dalam bidang teks atau teks-array diperlakukan sebagai stopword jika melebihi 256 karakter.</li></ul>



Item	Kuota
Konvensi penamaan	<ul style="list-style-type: none"><li>• Nama Domain: Karakter yang diizinkan adalah a-z (huruf kecil), 0-9, dan tanda hubung (-). Nama domain harus dimulai dengan huruf atau angka dan setidaknya 3 dan tidak lebih dari 28 karakter.</li><li>• Nama Bidang: Karakter yang diizinkan adalah a-z (huruf kecil), 0-9, dan _ (garis bawah). Nama bidang harus dimulai dengan huruf dan setidaknya 1 dan tidak lebih dari 64 karakter. Skor nama dicadangkan dan tidak dapat digunakan sebagai nama bidang.</li><li>• Nama Ekspresi: Karakter yang diizinkan adalah a-z (huruf kecil), 0-9, dan _ (garis bawah). Nama ekspresi harus dimulai dengan huruf dan setidaknya 3 dan tidak lebih dari 64 karakter. Skor nama dicadangkan dan tidak dapat digunakan sebagai nama ekspresi.</li><li>• ID dokumen: ID dokumen (_id) dapat berisi huruf atau angka dan karakter berikut: _ - = # ; : / ? @ &amp; . ID dokumen harus minimal 1 dan panjangnya tidak lebih dari 128 karakter.</li></ul>
Ukuran dokumen kebijakan	Ukuran maksimum dokumen CloudSearch kebijakan Amazon adalah 100 KB.
Pembatasan wilayah	Wilayah ap-northeast-2 hanya mendukung tipe instans m4.
_skor	Skor relevansi teks dokumen adalah nilai floating point positif.
Cari domain	Setiap akun AWS dapat membuat hingga 100 domain pencarian.

Item	Kuota
Cari partisi	<p>Indeks pencarian dapat dibagi menjadi maksimal 10 partisi. Anda dapat <a href="#">mengajukan permintaan</a> jika Anda perlu meningkatkan batas ini.</p> <p>Untuk menghindari kegagalan kueri penelusuran, CloudSearch domain Amazon dapat tumbuh melampaui batas partisi maksimum ini, tetapi penambahan dokumen baru ditolak. Jika Anda menemukan skenario ini, hapus dokumen dan picu IndexDocuments API. Bergantian, minta kenaikan batas.</p> <p>Anda dapat memantau Amazon CloudWatch <code>IndexUtilization</code> dan <code>Partitions</code> metrik untuk mengambil tindakan sebelum melebihi batas partisi maksimum.</p>
Cari replika	<p>Setiap partisi pencarian dapat memiliki hingga 5 replika.</p> <div data-bbox="695 1020 1510 1239"><p> <b>Note</b></p><p>Mengaktifkan Multi-AZ menggandakan jumlah replika.</p></div>

Item	Kuota
Permintaan pencarian	<ul style="list-style-type: none"><li>• kueri majemuk: Dapat berisi maksimum 1024 klausa.</li><li>• DAPATKAN permintaan: Ukuran maksimum permintaan pencarian yang dikirimkan sebagai permintaan HTTP GET adalah 8190 byte.</li><li>• parameter faset: Jumlah maksimum nilai faset yang dapat Anda kembalikan adalah 10.000.</li><li>• parameter ukuran: Dapat berisi nilai dalam kisaran 0 - 10000. Jumlah ukuran dan parameter awal tidak boleh melebihi 10.000. Jika Anda perlu halaman melalui lebih dari 10.000 klik, gunakan kursor.</li><li>• parameter sort: Dapat berisi hingga 10 bidang int dan ekspresi.</li><li>• parameter mulai: Dapat berisi nilai dalam kisaran 0 - 10000. Jumlah ukuran dan parameter awal tidak boleh melebihi 10.000. Jika Anda perlu halaman melalui lebih dari 10.000 klik, gunakan kursor.</li></ul>
Penyarankan	<ul style="list-style-type: none"><li>• Anda dapat menentukan maksimal 10 sugester untuk domain.</li><li>• Hanya 512 byte pertama dari bidang teks yang digunakan untuk menghasilkan saran.</li><li>• Skor yang dihitung dari sugester's <code>SortExpression</code> dibulatkan ke bilangan bulat terdekat, dengan lantai 0 dan langit-langit <math>2^{31}-1</math>.</li></ul>
Ukuran kamus sinonim	Ukuran maksimum kamus CloudSearch sinonim Amazon adalah 100 KB.

# CloudSearch Sumber Daya Amazon

Tabel berikut mencantumkan sumber daya yang mungkin berguna bagi Anda saat bekerja dengan Amazon CloudSearch.

Sumber Daya	Deskripsi
<a href="#">AWS SDK</a>	Sebagian besar <a href="#">AWS SDK</a> mendukung Amazon CloudSearch, termasuk SDK Java, .NET, Node.js, PHP, Python, dan Ruby.
<a href="#">Data CloudSearch Sampel Amazon</a>	Unduh <a href="#">Data Sampel IMDb</a> untuk mengaktifkan domain pencarian dan berjalan cepat dengan alat baris perintah atau API Layanan Konfigurasi dan lihat cara memformat data Anda sendiri untuk Amazon CloudSearch.
<a href="#">Forum CloudSearch Diskusi Amazon</a>	Forum tempat CloudSearch pengguna Amazon dapat memposting pertanyaan dan mendiskusikan berbagai CloudSearch topik Amazon.
<a href="#">CloudSearch Harga Amazon</a>	Informasi harga untuk Amazon CloudSearch.
<a href="#">Permintaan untuk Meningkatkan Batas</a>	Formulir untuk meminta peningkatan jumlah maksimum instance pencarian atau partisi untuk domain pencarian.
<a href="#">Amazon CloudSearch 2011-02-01 Panduan Pengembang</a>	<a href="#">Panduan Pengembang CloudSearch Amazon 2011-02-01 hanya tersedia dalam PDF: Unduh PDF.</a>

# Riwayat Dokumen untuk Amazon CloudSearch

Topik ini menjelaskan perubahan penting pada Amazon CloudSearch.

Tanggal yang relevan untuk Riwayat ini:

- Versi produk saat ini- 2013-01-01
- Rilis produk terbaru— 6 Januari 2021
- Pembaruan dokumentasi terbaru—6 Januari 2021

Perubahan	Deskripsi	Tanggal Rilis
Jenis instance baru	Amazon CloudSearch sekarang menggunakan jenis instance yang lebih baru untuk domain baru. Jenis instans ini memberikan perkembangan penskalaan yang lebih intuitif dan kinerja yang lebih baik dengan harga yang sama.	6 Januari 2021
Menegakkan HTTPS	Anda sekarang dapat meminta agar semua permintaan ke CloudSearch domain Amazon Anda tiba melalui HTTPS. Untuk mempelajari selengkapnya, lihat <a href="#">the section called “Mengkonfigurasi Opsi Endpoint Domain”</a> .	13 November 2019
Support untuk penandaan sumber daya	Amazon CloudSearch menambahkan dukungan untuk penandaan sumber daya. Untuk informasi selengkapnya, lihat <a href="#">Menandai CloudSearch Domain Amazon</a> di panduan layanan ini.	10 Februari 2016
Dukungan AP (Seoul)	Amazon CloudSearch menambahkan dukungan untuk ap-northeast-2 wilayah AP (Seoul). Untuk daftar wilayah yang didukung oleh Amazon CloudSearch, lihat <a href="#">Wilayah AWS dan Titik Akhir</a> di Referensi Umum AWS	28 Januari 2016
Integrasi dengan Amazon CloudWatch dan dukungan	Anda sekarang dapat menggunakan Amazon CloudWatch untuk memantau CloudSearch domain Amazon Anda. CloudWatch adalah layanan pemantauan untuk sumber	5 Maret 2015


Perubahan	Deskripsi	Tanggal Rilis
untuk statistik bidang indeks	<p>daya cloud AWS dan aplikasi yang Anda jalankan di AWS. Amazon CloudSearch secara otomatis mengirimkan metrik CloudWatch sehingga Anda dapat mengumpulkan dan menganalisis statistik kinerja. Anda dapat memantau metrik ini dengan menggunakan CloudSearch konsol Amazon, atau dengan menggunakan konsol, AWS CLI, atau AWS SDK. CloudWatch Tidak ada biaya untuk CloudSearch metrik Amazon yang dilaporkan melalui CloudWatch. Untuk informasi selengkapnya tentang penggunaan CloudWatch untuk memantau domain penelusuran, lihat <a href="#">Memantau Domain dengan Amazon CloudWatch</a>.</p> <p>Anda juga sekarang dapat mengambil statistik terhadap bidang numerik berkemampuan aspek. Amazon CloudSearch dapat mengembalikan statistik berikut terhadap bidang numerik yang diindeks dalam dokumen: count, min, max, mean, missing, stddev, sum, dan. sumOfSquares Untuk mempelajari lebih lanjut tentang statistik bidang indeks, lihat <a href="#">Meminta Informasi Lebih Lanjut</a>.</p>	
Support untuk tipe instans M3	<p>Anda sekarang dapat menggunakan instans M3 untuk memberi daya pada domain Amazon CloudSearch Anda. Amazon CloudSearch sekarang mendukung jenis instance berikut untuk domain yang baru dibuat: <code>m1.small</code>, <code>m3.medium</code>, <code>m3.large</code>, <code>m3.xlarge</code>, dan <code>m3.2xlarge</code>. Untuk informasi selengkapnya tentang jenis instans yang baru tersedia dan cara memodifikasi domain yang ada, lihat <a href="#">Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch</a>.</p>	10 Februari 2015


Perubahan	Deskripsi	Tanggal Rilis
Support untuk Dynamic Fields	Bidang dinamis menyediakan cara untuk mengindeks dokumen tanpa mengetahui terlebih dahulu bidang apa yang dikandungnya. Nama bidang dinamis mendefinisikan pola yang berisi wildcard (*) sebagai karakter pertama, terakhir, atau satu-satunya. Setiap bidang dokumen yang tidak dikenal yang cocok dengan pola dikonfigurasi dengan opsi pengindeksan bidang dinamis. Untuk informasi selengkapnya, lihat <a href="#">Menggunakan Bidang Dinamis di Amazon CloudSearch</a> .	11 Desember 2014

Perubahan	Deskripsi	Tanggal Rilis
Pemrosesan dan CloudTrail Dukungan Bahasa Jepang yang Ditingkatkan	<p>Sekarang Anda dapat mengontrol cara Amazon CloudSearch memberi token bahasa Jepang dengan menambahkan kamus tokenisasi Jepang kustom ke skema analisis yang Anda gunakan untuk bidang yang berisi bahasa Jepang. Mengkonfigurasi kamus tokenisasi kustom dapat meningkatkan akurasi hasil pencarian dengan memfasilitasi pengindeksan dan pengambilan frase khusus domain. Untuk mempelajari selengkapnya tentang menggunakan kamus kustom, lihat <a href="#">Menyesuaikan Tokenisasi Jepang</a></p> <p>Anda juga dapat mengindeks bigram untuk bahasa Mandarin, Jepang, dan Korea. Untuk informasi selengkapnya, lihat <a href="#">Pengindeksan Bigram untuk China, Jepang, dan Korea</a>.</p> <p>Anda juga sekarang dapat menggunakan AWS CloudTrail untuk mendapatkan riwayat panggilan API CloudSearch konfigurasi Amazon dan peristiwa terkait untuk akun Anda. CloudTrail adalah layanan web yang merekam panggilan API akun Anda dan mengirimkan file log yang dihasilkan ke bucket Amazon S3 Anda. Anda juga dapat menggunakan CloudTrail untuk melacak perubahan yang dibuat pada sumber daya AWS Anda. Misalnya, Anda dapat menggunakan riwayat panggilan API untuk melakukan analisis keamanan atau memecahkan masalah operasional. CloudTrail juga memudahkan Anda untuk menunjukkan kepatuhan terhadap kebijakan internal atau standar peraturan. Untuk informasi selengkapnya, lihat <a href="#">Security at Scale: Logging in AWS</a> whitepaper. Untuk informasi selengkapnya tentang penggunaan CloudTrail untuk mencatat CloudSearch panggilan Amazon, lihat <a href="#">Mencatat Panggilan API CloudSearch Konfigurasi Amazon dengan AWS CloudTrail</a>.</p>	15 Oktober 2014



Perubahan	Deskripsi	Tanggal Rilis
Pembaruan Dokumentasi	Pembaruan ini menjelaskan bahwa Anda harus menyandikan URL string kueri penelusuran dan memberikan informasi tambahan tentang mendapatkan informasi aspek untuk bucket yang dipilih. Untuk informasi lebih lanjut tentang aspek ember, lihat. <a href="#">Mendapatkan Informasi Facet</a>	19 September 2014

Perubahan	Deskripsi	Tanggal Rilis
Integrasi IAM yang Ditingkatkan	<p>Sekarang Anda dapat menggunakan IAM untuk mengontrol akses ke setiap dokumen domain, mencari, dan menyarankan layanan serta menggunakan AWS Signature Version 4 untuk menandatangani semua CloudSearch permintaan Amazon. Permintaan ditandatangani secara otomatis saat Anda menggunakan AWS SDK terbaru dan AWS CLI. Untuk informasi selengkapnya, lihat <a href="#">configure access policies</a>.</p> <p>Sehubungan dengan rilis ini, ada pembaruan alat baris CloudSearch perintah Amazon. CLT yang diperbarui sekarang secara otomatis menandatangani permintaan unggahan dokumen yang dikirimkan melalui <code>cs-import-documents</code> perintah. Anda dapat mengunduh bundel CLT baru dari <a href="#">halaman alat CloudSearch pengembang Amazon</a>.</p> <div data-bbox="444 1052 1289 1604" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px;"><p> <b>Important</b></p><p>Pembaruan CLT ini hanya berisi dua perintah: <code>cs-import-documents</code> dan <code>cs-configure-from-batches</code>. Semua tindakan konfigurasi harus dilakukan dengan menggunakan file AWS CLI. Ini AWS CLI juga mendukung mengunggah dokumen dan mengirimkan pencarian dan menyarankan permintaan. Untuk informasi selengkapnya, lihat <a href="#">Panduan Pengguna Antarmuka Baris Perintah AWS</a>.</p></div>	14 Agustus 2014

Perubahan	Deskripsi	Tanggal Rilis
CloudSearch Dukungan Amazon yang Ditingkatkan di AWS SDK dan AWS CLI	<p>AWS SDK dan AWS CLI sekarang menyediakan dukungan penuh untuk semua operasi API Amazon CloudSearch 2013-01-01, termasuk membuat, mengonfigurasi, dan mengelola domain penelusuran, mengunggah dokumen, dan mengirimkan permintaan pencarian. Untuk informasi tentang menginstal dan menggunakan AWS CLI, lihat <a href="#">Panduan Pengguna AWS Command Line Interface</a>.</p> <div data-bbox="444 638 1289 997"><p> <b>Note</b></p><p>Untuk menghasilkan batch dokumen dan secara otomatis mengonfigurasi opsi pengindeksan berdasarkan konten batch, Anda masih perlu menggunakan alat baris perintah CloudSearch Amazon mandiri.</p></div>	26 Juni 2014

Perubahan	Deskripsi	Tanggal Rilis
Dukungan Bahasa Ibrani dan Opsi Penskalaan Hitungan Partisi yang Diinginkan	<p>Amazon CloudSearch sekarang mendukung bahasa Ibrani selain 33 lainnya <a href="#">Bahasa yang Didukung</a>. Pembaruan ini juga menambahkan opsi penskalaan baru, jumlah partisi yang diinginkan. Anda dapat menggunakan opsi ini untuk mengkonfigurasi jumlah partisi indeks untuk domain yang menggunakan jenis instance pencarian m2.2xlarge. Jika Anda memiliki sejumlah besar data pencarian, pra-konfigurasi domain untuk menggunakan lebih banyak partisi dapat memungkinkan Anda untuk memuat data lebih cepat. Anda juga dapat mengonfigurasi domain dengan partisi tambahan untuk menjatuhkan jumlah dokumen per partisi dan mempercepat kueri kompleks. Amazon masih CloudSearch akan menskalakan domain naik atau turun berdasarkan volume data atau lalu lintas, tetapi jumlah partisi tidak akan pernah turun di bawah jumlah partisi yang Anda inginkan. Untuk informasi selengkapnya, lihat <a href="#">Mengkonfigurasi Opsi Penskalaan di Amazon CloudSearch</a>.</p>	24 Maret 2014
Amazon CloudSearch 2013-01-01 API	<p>Amazon CloudSearch memiliki versi API baru dengan banyak peningkatan dan fitur baru. API baru tidak kompatibel dengan API 2011-02-01. Untuk menggunakan fitur baru, Anda harus membuat domain pencarian baru dengan API 2013-01-01. Dalam hubungannya dengan rilis ini, ada juga satu set baru alat baris perintah. Perhatikan bahwa alat baru memerlukan JRE yang kompatibel dengan Java 7, jadi Anda mungkin perlu memperbarui Java untuk menggunakan alat.</p>	24 Maret 2014

# Glosarium AWS

Lihat terminologi AWS terbaru di [AWS glosarium](#) dalam Referensi Glosarium AWS.

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