ユーザーガイド

Amazon Q Developer

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

Amazon Q Developer: ユーザーガイド

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

Amazon の商標とトレードドレスは、Amazon 以外の製品またはサービスとの関連において、顧客に混乱を招いたり、Amazon の名誉または信用を毀損するような方法で使用することはできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Amazon Q Developer とは? 1
はじめに 1
Amazon Q Developer の料金 3
機能 4
分析 4
データの要約 4
マネジメントとガバナンス 4
ノードの探索 4
調査中 4
インベントリの確認 5
AWS Console Mobile Applicationで Amazon Q を使用する 6
コンソールエラーの診断 6
コンピューティング 6
Amazon EC2 インスタンスの選択 6
データベース 7
自然言語を使用したデータベースクエリの作成 7
ネットワーキングとコンテンツ配信 7
ネットワーク到達可能性の分析 7
セキュリティ、アイデンティティ、コンプライアンス 8
ネットワークセキュリティ設定の分析 (プレビュー) 8
デベロッパーツール 8
コード機能の開発 8
インラインコードの提案の取得 8
IDEs のコードに関するチャット 8
セキュリティの脆弱性と品質の問題に関するコードの確認 9
コードの変換 9
ユニットテストの生成 9
Amazon CodeCatalyst でのソフトウェア開発 10
Amazon SageMaker AI Studio のコードに関するチャット 10
コマンドラインと の操作 AWS CloudShell 11
アプリケーション統合 11
AWS のサービスを自動化するためのスクリプトの作成 11
ETL スクリプトの記述とデータの統合 12
サードパーティ統合 12

Amazon Q での GitLab デュオ の使用 12
での Amazon Q Developer 機能の使用 GitHub 13
クラウド財務管理 13
コストを理解する 13
カスタマーサポート 13
Amazon Q から直接カスタマーサポートを受ける 13
サポートチケットの作成 13
チャットアプリケーションの Amazon Q 13
入門 15
サービスの階層 15
空き 15
Pro (プロ) 16
個人アカウントの使用を開始する 17
Builder IDs の制限 18
ステップ 1: サインアップする 18
ステップ 2: Amazon Q をインストールする 18
ステップ 3: (オプション) Pro 階層へのアップグレード 19
IAM Identity Center の使用を開始する 19
ステップ 1: デプロイオプションを選択する 19
ステップ 2: ユーザーをサブスクライブする 25
Pro 階層サブスクリプション 38
サポート対象のリージョン 39
サブスクリプションの請求 42
サブスクリプションステータス 43
開始 URL の検索 45
暗号化方法の管理 45
Q Developer プロファイル 47
サブスクリプションのトラブルシューティング 51
サブスクリプションの集計リストの表示 55
サブスクリプション解除 57
Pro 階層へのアップグレード 60
オン AWS 64
Amazon Q Developer Pro サブスクリプションへの認証 64
に関するチャット AWS 65
アクセス許可を追加する 66
会話を開始する 66

コンソールで会話を管理する 66
チャット設定の設定 67
質問例 68
リソースに関するチャット 68
リソースのトラブルシューティングを Amazon Q に依頼する 71
コストに関するチャット 75
ネットワークセキュリティに関するチャット 76
テレメトリとオペレーションに関するチャット 77
プラグインの使用 78
CloudZero 79
Datadog 86
Wiz 94
Console-to-Code 101
Console-to-Code 101
Console-to-Code を使用できる場所 102
アクセス許可の付与 103
使用 103
コンソールエラーの診断 105
アクセス許可を追加する 106
コンソールでの一般的なエラーの診断 106
とのチャット サポート 106
前提条件 107
適切なサービスを指定する 107
サポートケースの作成 108
フィードバックを残す 110
IDE 内 111
サポートされている IDE 111
Amazon Q のインストール 112
サポートされている IDE バージョン 113
Eclipse IDE 内 114
JetBrains IDE 内 115
Visual Studio Code 内 116
Visual Studio の場合 118
AWS コンソールの IAM プリンシパル 120
コードに関するチャット 120
エージェントコーディング 121

他の自然言語の使用 121
IDE で Amazon Q の操作 121
トピックおよび質問の例 123
応答に関する問題の報告 123
コードの説明と更新 124
インラインチャット 124
チャットへのコンテキストの追加 127
チャット履歴の圧縮 135
会話の管理をする 137
チャットでのショートカットキーの使用 139
モデルの選択 140
インライン提案の生成 141
提案の一時停止 142
Amazon Q のコード補完の仕組み 146
コーディング環境で AWS の提案 152
ショートカットキーの使用 160
コードリファレンスの使用 166
コードの例 177
コードの変換 (/transform) 200
Java アプリケーションの変換 200
.NET アプリケーションの変換 244
機能の開発 (/dev) 251
/dev で機能を開発する 252
ベストプラクティス 254
タスクの例 254
クォータ 255
「/dev」提案のテスト 256
トラブルシューティング 264
ユニットテストの生成 (/test) 265
前提条件 266
/test を使用してユニットテストを生成する 266
使用法と考慮事項 268
特殊なケースの処理 268
コードの確認 (/レビュー) 269
コードの問題のタイプ 270
クォータ 255

レビューの開始 ... 271
コードの問題を理解する ... 275
コードの問題に対処する ... 277
コードの問題のフィルタリング ... 281
コードの問題の重要度 ... 283
ドキュメントの生成 (/doc) ... 285
ユースケース ... 285
[サポートされているファイルの種類] ... 286
クォータ ... 255
README の生成 ... 288
ベストプラクティス ... 289
サポートされている言語 ... 289
インライン提案 ... 290
変換 ... 291
ユニットテスト生成 (/test) ... 291
コードレビュー (/レビュー) ... 292
カスタマイズ ... 292
Q CLI を使用する ... 294
サポートされている環境 ... 295
macOS ... 295
リナックス ... 295
自然言語 ... 296
インストール ... 296
macOS ... 296
Linux AppImage ... 298
Ubuntu ... 298
Homebrew ... 299
zip ファイルを使用する ... 299
リモート使用の場合 ... 303
アンインストール ... 305
デバッグ ... 305
チャットの使用 ... 306
チャットセッションの開始 ... 307
コマンド ... 307
アクセス許可 ... 308
モデルの選択 ... 312

Conversations ... 313
イメージ ... 316
セキュリティに関する考慮事項 ... 317
カスタムエージェント ... 320
概要 ... 320
カスタムエージェントの定義 ... 322
カスタムエージェントの管理 ... 325
設定リファレンス ... 327
例 ... 332
トラブルシューティング ... 340
エディタコマンドの使用 ... 345
基本的な使用法 ... 346
任意のエディタの設定 ... 346
仕組み ... 347
エディタでのコードの使用 ... 347
を他のコマンドと組み合わせる ... 348
コマンドの組み合わせのベストプラクティス ... 351
効果的な使用のためのヒント ... 351
トラブルシューティング ... 351
Git 対応ファイル選択の使用 ... 352
Git 対応の選択の仕組み ... 352
Git 対応ファイル選択の使用 ... 352
Git 対応の選択に関するヒント ... 353
コンテキスト管理 ... 353
コンテキストの管理 ... 353
一般的なユースケース ... 355
コンテキストフック ... 357
コンテキストフックのタイプ ... 357
コンテキストフックの管理 ... 358
ユースケースの例 ... 360
動作と制限 ... 360
セキュリティに関する考慮事項 ... 361
トラブルシューティング ... 361
設定 ... 362
設定へのアクセス ... 362
コマンドライン設定の管理 ... 362

その他の Amazon Q CLI コマンド 363
コマンドリファレンス 364
グローバル引数 364
コマンド 364
ログファイル 376
コマンドラインのサポート 377
自動入力 377
インライン 378
サポートされているツール 379
貢献 379
GitHub リポジトリ 380
レポートに関する問題 380
フィードバックの提供 381
RFCs 381
テレメトリデータ 381
MCP の使用 383
MCP の概要 383
CLI で 384
設定コマンド 384
MCP サーバーのロード 385
サーバーステータスの確認 385
サーバー初期化の設定 385
ツールとプロンプト 385
MCP ツールについて 385
使用可能なツールの検出 386
ツールの使用方法 386
プロンプトの使用 386
IDE を使用する 387
IDE を使用する 387
MCP サーバーの追加 388
MCP 設定のトラブルシューティング 389
MCP サーバーの有効化 389
MCP サーバーの無効化 389
現在有効になっている MCP サーバーの削除 390
現在無効になっている MCP サーバーの削除 390
ツールのアクセス許可の確認と調整 390

主な利点 390
MCP アーキテクチャ 391
コア MCP の概念 392
ツール 392
プロンプト 392
リソース 392
MCP セキュリティ 393
セキュリティモデル 393
セキュリティに関する考慮事項 393
サードパーティー統合 394
GitLab Duo Amazon Q Developer を使用する 394
Amazon Q Developer for GitHub (プレビュー) 394
Amazon Q Developer のプロジェクトルール 395
GitLab Duo 395
GitLab Duo の概念 396
入門 401
CI/CD パイプラインのカスタマイズ 402
トラブルシューティング 405
GitHub (プレビュー) 406
Amazon Q Developer アプリのインストールとアクセスの承認 408
Amazon Q Developer エージェント 409
アプリのインストールの登録 410
でのブラウザ拡張機能の使用 GitHub 410
GitHub 問題とプルリクエストでのスラッシュコマンドの使用 410
クイックスタート 411
機能の開発と反復処理 414
コードの確認 417
コードの変換 419
ワークフローのカスタマイズ 421
使用制限と設定を増やす 427
設定 429
トラブルシューティング 431
プロジェクトルールの作成 433
チャットアプリケーション内 435
チャネルで Amazon Q チャットを有効にする 435
チャネルで Amazon Q に質問する 436

カスタマイズ 438
準備中 439
認可 439
データの準備 439
作成 440
データソースへの接続 441
カスタマイズとデータ 443
トラブルシューティング 443
削除 444
最適化 444
最適化 445
トラブルシューティング 445
ログ配信の設定 445
Amazon CloudWatch Logs を理解する 446
コンソールのエラーメッセージを理解する 448
アクティブ化 450
バージョンのアクティブ化 450
[更新中] 451
新しいバージョンを作成する 451
ユーザーとグループの追加 452
使用 453
セキュリティ 456
データ保護 457
データストレージ 458
データ暗号化 459
サービスの改善 461
IDE とコマンドラインでデータ共有をオプトアウトする 462
クロスリージョン処理 470
Identity and Access Management 472
対象者 473
アイデンティティを使用した認証 473
ポリシーを使用したアクセスの管理 477
Amazon Q が IAM で機能する仕組み 479
Amazon Q へのアクセスを管理する 486
統合のために Amazon Q Developer へのアクセスを管理する 526
Amazon Q アクセス許可リファレンス 528

AWS Amazon Q の マネージドポリシー 531
サービスにリンクされたロールの使用 537
コンプライアンス検証 545
耐障害性 546
インフラストラクチャセキュリティ 546
ファイアウォール、プロキシ、データ境界 547
許可リストの一般的な URLs 547
許可リストの Amazon S3 バケット URLs と ARNs 548
Amazon Q での企業プロキシの設定 551
VPC エンドポイントAWS PrivateLink 557
Amazon Q の VPC エンドポイントに関する考慮事項 558
前提条件 558
Amazon Q 用のインターフェイス VPC エンドポイントの作成 558
オンプレミスコンピュータを使用して Amazon Q エンドポイントに接続する 559
コンソール内のコーディング環境を使用して Amazon Q エンドポイントに接続する 559
Amazon EC2 インスタンスのサードパーティー IDE AWS PrivateLink から 経由で Amazon Q に接続する Amazon EC2 560
モニタリングと追跡 561
で AWS CloudTrail 562
CloudTrail での Amazon Q Developer の情報 562
Amazon Q Developer ログファイルエントリの概要 563
CloudWatch の使用 567
特定のユーザーによるアクションの特定 569
カスタマイズ関連のログへのアクセス 586
使用状況メトリクスの表示 (ダッシュボード) 587
ダッシュボードメトリクス 589
ダッシュボードの無効化 592
ダッシュボードのトラブルシューティング 592
ユーザーごとのアクティビティの表示 593
ユーザーアクティビティレポートのメトリクス 596
ユーザーのプロンプトのログ記録 603
プロンプトログの例 606
サポート対象のリージョン 616
サポートされているリージョン (デフォルトで有効) 616
サポートされているオプトインリージョン 617
トラブルシューティング 619

ログアクセスと分析 619
ログアクセスの概要 620
IDE 拡張機能ログ 620
Amazon Q CLI ログ 621
一般的なログパターンとソリューション 625
ログ分析のヘルプ 626
Amazon Q Developer のサービス名称変更 627
ドキュメント履歴 628
..... dclvii

Amazon Q Developer とは?

> **Note**
>
> Amazon Bedrock を利用: Amazon Q Developer は Amazon Bedrock 上に構築されており、Amazon Bedrock に実装された自動不正検出が含まれており、安全性、セキュリティ、AI の責任ある使用を強化します。

Amazon Q Developer は、生成人工知能 (AI) を活用した会話型アシスタントで、 AWS アプリケーションの理解、構築、拡張、運用に役立ちます。 AWS アーキテクチャ、 AWS リソース、ベストプラクティス、ドキュメント、サポートなどについて質問できます。Amazon Q は、コンテキストへの関連性が最も高く、実用的な回答が得られるように、機能を常に更新しています。

統合開発環境 (IDE) で使用すると、Amazon Q はソフトウェア開発を支援します。Amazon Q では、コードに関するチャット、インラインコードの完了、新しいコードの生成、セキュリティ上の脆弱性のスキャン、言語の更新、デバッグ、最適化などのコードのアップグレードと改善を行うことができます。

Amazon Q は、API を介して基盤モデル (FM) を利用可能にするフルマネージドサービスである Amazon Bedrock が基盤となっています。Amazon Q を強化するモデルは、 でのビルドを加速するために、より完全で実用的な、参照された回答を得るために、高品質の AWS コンテンツで強化されています AWS。

> **Note**
>
> これは、Amazon Q Developer に関するドキュメントです。Amazon Q Business のドキュメントをお探しの場合は、「Amazon Q Business ユーザーガイド」を参照してください。

Amazon Q Developer の使用を開始する

Amazon Q の使用をすばやく開始するには、次の方法でアクセスできます。

AWS アプリとウェブサイト

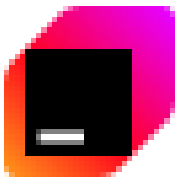
IAM ID に必要なアクセス許可を追加し、Amazon Q アイコンを選択して AWS Management Console、、 AWS Documentation ウェブサイト、 AWS ウェブサイト、または でチャットを開

始します AWS Console Mobile Application。詳細については、「AWS アプリケーションとウェブサイトでの Amazon Q Developer の使用」を参照してください。

IDE

Amazon Q 拡張機能をダウンロードし AWS ビルダー ID 、 (AWS アカウント不要) を使用して無料でサインインします。

Visual Studio Code で Amazon Q をダウンロードする



JetBrains IDE で Amazon Q をダウンロードする

AWS Toolkit for Visual Studio で Amazon Q をダウンロードする

IDEs Eclipse で Amazon Q をダウンロードする (プレビュー)

IDE から Amazon Q アイコンを選択してチャットを開始するか、開発ワークフローを開始します。詳細については、「IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする」を参照してください。

[Command line]

のコマンドライン用に Amazon Q をダウンロードする macOS

のコマンドライン用に Amazon Q をダウンロードする Linux AppImage

のコマンドライン用に Amazon Q をダウンロードする Ubuntu

詳細については、「コマンドラインで Amazon Q Developer を使用する」を参照してください。

Amazon Q Developer in chat applications

AmazonQDeveloperAccess 管理ポリシーを Microsoft Teamsまたは Slackアプリケーションの IAM ID とチャネルガードレールに追加します。詳細については、「チャットアプリケーションでの Amazon Q Developer とのチャット」を参照してください。

Amazon Q Developer の料金

Amazon Q Developer は、無料利用枠と Amazon Q Developer Pro サブスクリプションを通じて利用できます。詳細については、「Amazon Q Developer の料金」を参照してください。

Amazon Q Developer 機能

Amazon Q Developer は、 AWS 環境やサービス全体で利用でき、サードパーティー IDEs のコーディングアシスタントとしても利用できます。

Amazon Q Developer の機能の多くはチャットインターフェイスに存在し、自然言語を使用して質問AWSしたり、コードに関するヘルプを得たり、リソースを調べたり、トラブルシューティングしたりできます。Amazon Q とチャットすると、Amazon Q は現在の会話のコンテキストを基に回答を返します。フォローアップの質問をしたり、新しい質問をする際にその回答を参照したりできます。

その他の Amazon Q Developer 機能は、 AWS サービスコンソールおよびサポートされている IDEs のワークフローの一部として使用できます。以下のセクションでは、 AWS エクスペリエンス全体で発生する可能性のある Amazon Q Developer のさまざまな機能について説明します。

分析

データの要約

Amazon Q QuickSight を使用すると、Generative BI オーサリングエクスペリエンスを活用し、データのエグゼクティブサマリーを作成し、データの質問をして回答し、データストーリーを生成できます。

詳細については、QuickSight ユーザーガイドの「Amazon Q QuickSight での Generative BI の使用」を参照してください。 QuickSight

マネジメントとガバナンス

テキストプロンプトを使用したノードの探索

AWS Systems Manager と Amazon Q を使用すると、マネージドノードまたはインスタンスについて自然言語の質問をすることができます。次に、Amazon Q は Systems Manager `ListNodes`アクションを使用し、テキスト入力に基づいてフィルターを作成して結果を取得します。

詳細については、「 AWS Systems Manager ユーザーガイド」の「Amazon Q のテキストプロンプトを使用したノードの探索」を参照してください。

運用上の問題の調査

Amazon CloudWatch の調査は、 AWS 環境全体のリソース、イベント、アクティビティを調査および分析する機能を強化します。自然言語処理を活用することで、Amazon Q は 内の複雑なシナリオと関係を理解するプロセスを簡素化します AWS アカウント。

Amazon Q Developer は、環境全体で CloudWatch 調査を高速化する AWS のに役立ちます。Q はテレメトリで異常を探し、調査すべき関連シグナルを表示し、潜在的な根本原因仮説を特定し、問題をより迅速に修正するための次のステップを提案します。

Amazon Q を調査ワークフローに統合することで、問題解決を加速し、 AWS 環境の理解を深め、インフラストラクチャとアプリケーションについてより多くの情報に基づいた意思決定を行うことができます。

Amazon CloudWatch 調査のコンテキストで Amazon Q に尋ねる質問の例については、テレメトリとオペレーションに関するチャット「」を参照してください。

CloudWatch 調査全般の詳細については、「Amazon CloudWatch ユーザーガイド」の「CloudWatch 調査」を参照してください。 Amazon CloudWatch

AWS リソースのインベントリの取得

特定の AWS アカウントリソースについては、 AWS マネジメントコンソールのどこからでも Amazon Q にお問い合わせいただけます。リソースに関する関連情報の場所がわからない場合や、あるサービスのコンソールを利用中に、ワークフローを中断することなく別のサービスのリソースに関する情報にアクセスしたい場合があります。

Amazon Q Developer は、リソースに関する自然言語の質問に回答し、そうしたリソースへの詳細なリンクを提供するため、リソースをすばやく見つけることができます。Amazon Q に、アカウント内のリソースのタイプを一覧表示したり、特定のリソースの詳細を取得したり、リージョンや州などの基準に基づいてリソースを一覧表示したりするようにリクエストできます。

例えば、現在実行している Amazon EC2 インスタンスの数を知りたいとします。その場合、Amazon Q に自然言語で質問すると、特定のリソースに基づいた回答が得られます。

詳細については、「Amazon Q Developer とのリソースに関するチャット」を参照してください。

各タイプに適用される特定の制限と、そうした制限が特定のサブスクリプションパッケージの料金にどのように関係するかについては、「Amazon Q Developer の料金」を参照してください。

AWS Console Mobile Applicationで Amazon Q を使用する

Amazon Q は と統合され AWS Console Mobile Application 、 に関する質問に回答します AWS。AWS Management Consoleで Amazon Q にアクセスするのと同じ方法でアクセスを設定します。詳細については、「Amazon Q Developer の使用開始」を参照してください。

コンソールエラーの診断

AWS マネジメントコンソールでは、Amazon Q Developer は、アクセス許可の不足、誤った設定、AWS サービス制限の超過など、 サービスの使用中に発生する一般的なエラーを診断できます。

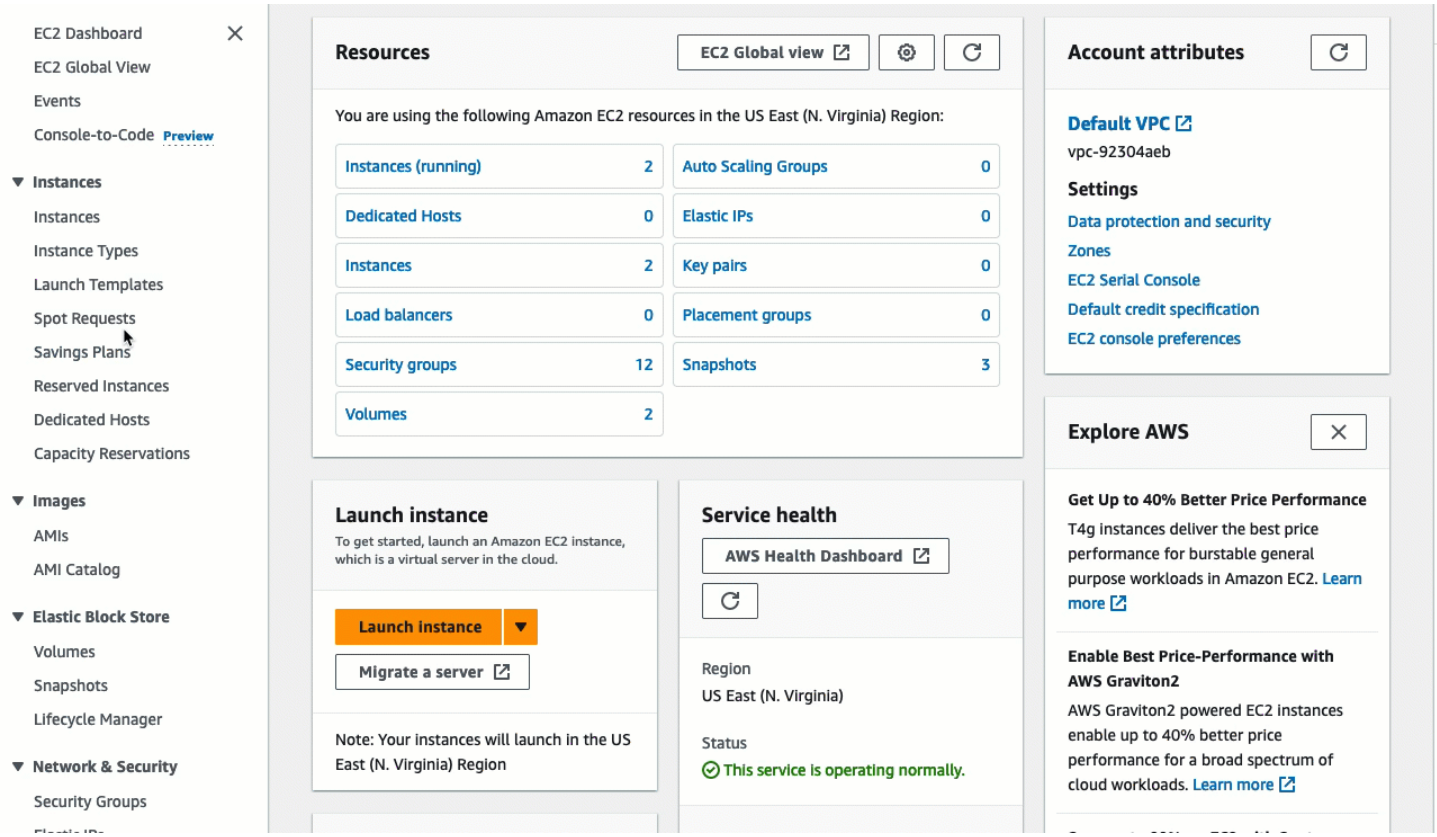
詳細については、「Amazon Q Developer でコンソールでの一般的なエラーを診断する」を参照してください。

コンピューティング

Amazon Elastic Compute Cloud インスタンスの選択

利用可能な Amazon EC2 インスタンスタイプの数が多いため、ワークロードに適したインスタンスタイプを見つけるには時間と手間がかかることがあります。Amazon Q のインスタンスタイプセレクターでは、ユースケース、ワークロードタイプ、CPU メーカーの優先設定、価格とパフォーマンスの優先度のほかに、ユーザーが指定できる追加のパラメータも考慮されます。そして、このデータを使用して、新しいワークロードに最適な Amazon EC2 インスタンスタイプの推奨とガイダンスを提供します。

詳細については、「Amazon Elastic Compute Cloud ユーザーガイド」の「EC2 インスタンスタイプファインダーからレコメンデーションを取得する」を参照してください。



データベース

自然言語を使用したデータベースクエリの作成

Amazon Q 生成 SQL は生成 AI を使用してユーザーインテント、クエリパターン、スキーマメタデータを分析し、Amazon Redshift 内で直接一般的な SQL クエリパターンを特定することで、ユーザーのクエリ作成プロセスを高速化し、実用的なデータインサイトを得るのに必要な時間を短縮します。

詳細については、「Amazon Redshift 管理ガイド」の「Amazon Q 生成 SQL の操作」を参照してください。

ネットワークとコンテンツ配信

ネットワークのトラブルシューティングの分析

Amazon Q を使用すると、Amazon VPCs で実行されるアプリケーションのネットワーク接続の問題の診断に役立ちます。Amazon Q ネットワークのトラブルシューティングでは、自然言語クエリを理

解し、Reachability Analyzer と連携して関連するレスポンスを提供できます。Amazon Q では、ネットワーク到達可能性に関する質問を会話形式で行うことができます。

詳細については、「Amazon VPC Reachability Analyzer ガイド」の「Reachability Analyzer の Amazon Q ネットワークのトラブルシューティング」を参照してください。

セキュリティ、アイデンティティ、コンプライアンス

ネットワークセキュリティ設定の分析 (プレビュー)

ネットワークセキュリティ設定に関する質問に対する回答は、 AWS Shield ネットワークセキュリティディレクターから自然言語で簡単に取得できます。Amazon Q は、ネットワークセキュリティの検出結果を分析し、コンソールとチャットアプリケーションで推奨される修復手順を提供します。詳細については、「 ネットワークセキュリティディレクターデベロッパーガイド」の「Amazon Q Developer を使用したAWS Shield ネットワークセキュリティの分析」を参照してください。

デベロッパーツール

と でのビルドについて Amazon Q Developer に質問 AWS し、ソフトウェア開発の支援を受けてください。Amazon Q は、コーディングの概念とコードスニペットについて説明できるほか、コードやユニットテストを生成し、デバッグやリファクタリングなどでコードを改善できます。

コード機能の開発

開発する機能を自然言語で説明した後、Amazon Q は現在のプロジェクトのコンテキストを使用して実装計画と付随するコードを生成できます。Amazon Q は、 AWS プロジェクトまたは独自のアプリケーションの構築に役立ちます。詳細については、「Amazon Q Developer を使用した機能の開発」を参照してください。

インラインコードの提案の取得

Amazon Q は、リアルタイムでコードを提案します コードの記述中、Amazon Q は既存のコードとコメントに基づいて、自動的に提案コードを生成します。詳細については、「Amazon Q Developer によるインライン提案の生成」を参照してください。

IDEs のコードに関するチャット

統合開発環境 (IDE) 内では、Amazon Q はプログラミングに関する概念的な質問、特定のコードの仕組み、コードの改善、コード生成など、ソフトウェア開発プロセスに関連する質問に答えることが

できます。また、チャットパネルからコードスニペットを更新および改善するように Amazon Q にリクエストすることもできます。多言語サポートでは、英語、北京語、フランス語、ドイツ語、イタリア語、日本語、スペイン語、韓国語、ヒンディー語、ポルトガル語など、サポートされている任意の自然言語で Amazon Q とチャットできます。詳細については、「コードに関する Amazon Q Developer とのチャット」を参照してください。

Amazon Q Developer でコードを記述し、最もフル機能の環境で開発支援を受けるには、「IDE での Amazon Q Developer の使用」を参照してください。

全体の他のインターフェイスで基本的なコード補完機能を有効にするには AWS、AWS コーディング環境でインライン提案を生成する「」を参照してください。

セキュリティの脆弱性と品質の問題に関するコードの確認

IDEs 内で、Amazon Q はコードのセキュリティの脆弱性とコード品質の問題を確認します。Amazon Q は、開発中のアプリケーションのセキュリティと品質をモニタリングするために、プロジェクト全体をコーディングまたはレビューしながら を確認できます。詳細については、「Amazon Q Developer でコードを確認する」を参照してください。

コードの変換

Amazon Q は、アプリケーションの言語およびオペレーティングシステム (OS) レベルの自動アップグレードを実行できます。詳細については、「Amazon Q Developer を使用した IDE でのコードの変換 」を参照してください。

ユニットテストの生成

Amazon Q Developer は、AI を活用したユニットテスト生成機能を提供し、開発チームがソフトウェア開発ライフサイクルを通じてコードカバレッジを改善できるようにします。ユニットテスト生成用の Amazon Q Developer エージェントは、次の環境で使用できます。

- Amazon Q Developer IDE 拡張機能。詳細については、「Amazon Q を使用したユニットテストの生成」を参照してください。
- GitLab の一部として GitLab 。詳細については、「the section called “GitLab クイックアクション”」を参照してください。

> **Note**
>
> ユニットテスト生成機能は、Amazon Q Developer がサポートするすべてのリージョンで使用できます。

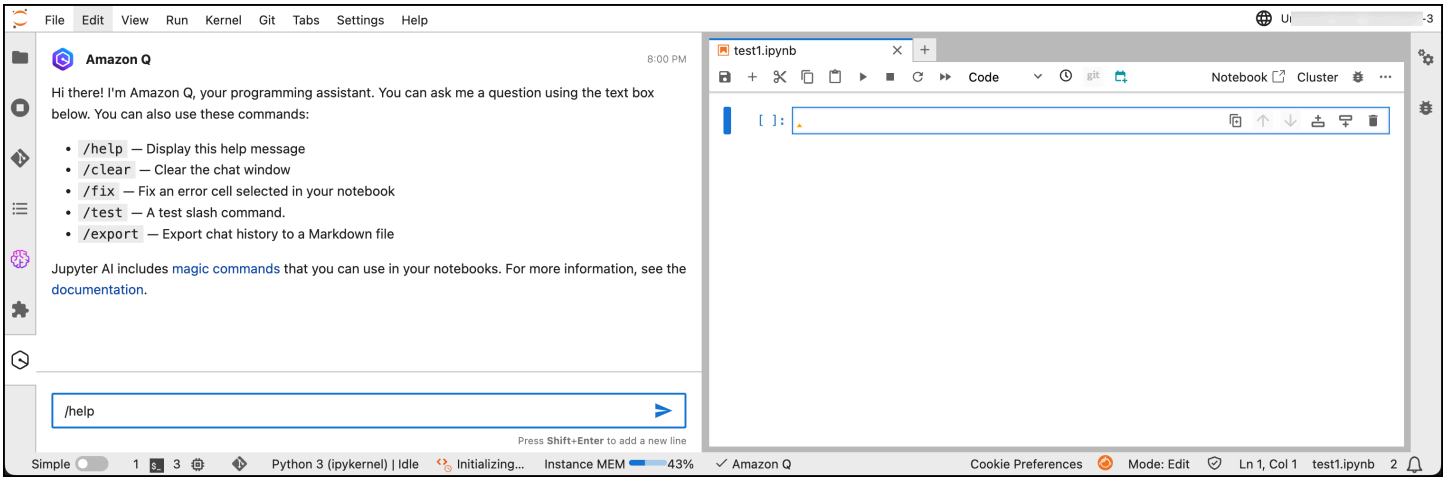
Amazon CodeCatalyst でのソフトウェア開発

CodeCatalyst の Amazon Q Developer には、スペース内のプロジェクトのユーザーがソフトウェアをより迅速に開発するのに役立つ生成 AI 機能が備わっています。Amazon Q に問題を割り当てたり、Amazon Q にタスクを推奨したりすることができます。また、Amazon Q に説明を記述したり、コンテンツを要約したりするようにリクエストすることもできます。

詳細については、「Amazon CodeCatalyst 管理者ガイド」の「Managing generative AI features in Amazon CodeCatalyst」を参照してください。

Amazon SageMaker AI Studio のコードに関するチャット

Amazon SageMaker AI Studio は、ML ワークフローを実行するためのウェブベースのエクスペリエンスです。Studio 内で Amazon Q Developer とチャットして、SageMaker AI 機能のガイダンスの取得、JupyterLab エラーのトラブルシューティング、サンプルコードの取得を行うことができます。



詳細については、SageMaker AI デベロッパーガイドの「Amazon Q を使用してMachine Learning ワークフローを高速化する」を参照してください。

コマンドラインと の操作 AWS CloudShell

コマンドラインインターフェイス (CLI)

コマンドラインに Amazon Q をインストールしたら、コンテキストに関連するサブコマンド、オプション、引数を入力するときに、それを使用して CLI コマンドを完了できます。コマンドラインに入力すると、AI によって補完が行われます。さらに、Amazon Q を使用して、自然言語指示を記述し、実行可能なシェルコードスニペットに即座に変換させることもできます。Amazon Q の複雑な質問をすることもできます。また、会話に基づいたフィードバックや指示、会話外のコンテキストや情報を提供します。その後、Amazon Q がユーザーに代わってアクションを実行できるように、Amazon Q にアクセス許可を付与できます。多言語サポートでは、英語、北京語、フランス語、ドイツ語、イタリア語、日本語、スペイン語、韓国語、ヒンディー語、ポルトガル語など、サポートされている任意の自然言語で Amazon Q とチャットできます。詳細については、「コマンドラインで Amazon Q Developer を使用する」を参照してください。

AWS CloudShell

で Amazon Q CLI を使用して AWS CloudShell 、自然言語の会話をしたり、質問をしたり、ターミナルで Amazon Q からレスポンスを受け取ったりすることもできます。関連するシェルコマンドを取得することで、構文を検索したり記憶したりする必要性を軽減できます。Amazon Q では、ターミナルで入力しながら、コマンドの提案を受け取ることができます。詳細については、「 AWS CLI での Amazon Q の使用」を参照してください AWS CloudShell。

アプリケーション統合

AWS のサービスを自動化するためのスクリプトの作成

AWS リソースの操作を正確に把握し、同じアクションを繰り返し実行している場合があります。Amazon Q に反復タスクを自動化するコードを記述するようにリクエストできます。

例えば、Amazon VPC、Amazon EC2 インスタンス、Amazon RDS データベースを使用するプロジェクトで作業しているとします。テストを進めるうちに、Amazon VPC の作成、サーバーの立ち上げ、データベースのデプロイを行う際、設定は毎回同じであることがわかりました。必ず同じインスタンスとデータベースタイプを選択し、同じオプションを選んで、同じセキュリティグループを使用しています。サブネットの NACL 設定も同じです。テスト条件を再現する際に、毎回同じ手動プロセスを繰り返すのは避けたいと考えています。

Amazon Q の Console-to-Code 機能を使用すると、ワークフローを毎回手動で実行する代わりに、ワークフローを自動化できます。まず、Amazon EC2 コンソールで Console-to-Code を有効にしま

す。次に、Amazon Q が、インスタンスの設定と起動のプロセスを実行するユーザーのアクションを記録します。最後に、Amazon Q が、先ほど実行したプロセスを自動化するコードを任意の言語で提供します。

詳細については、「Amazon Q Developer Console-to-Code による AWS サービスの自動化」を参照してください。

ETL スクリプトの記述とデータの統合

AWS Glue は、分析ユーザーが複数のソースからのデータを簡単に検出、準備、移動、統合できるようにするサーバーレスデータ統合サービスです。

の Amazon Q データ統合 AWS Glue には、次の機能が含まれています。

- チャット – の Amazon Q データ統合 AWS Glue は、 および送信 AWS Glue 元 AWS Glue と送信先のコネクタ、 AWS Glue ETL ジョブ、データカタログ、クローラと AWS Lake Formation、その他の機能ドキュメント、ベストプラクティスなどのデータ統合ドメインに関する自然言語の質問に回答できます。の Amazon Q データ統合はstep-by-stepの手順で AWS Glue 応答し、その情報源への参照が含まれています。
- データ統合コード生成 – の Amazon Q データ統合 AWS Glue は、ETL AWS Glue スクリプトに関する質問に回答し、自然言語の質問が英語で与えられた新しいコードを生成できます。
- トラブルシューティング — の AWS Glue Amazon Q データ統合は、 AWS Glue ジョブのエラーを理解し、根本原因を突き止めて問題を解決するためのstep-by-stepの手順を提供することを目的としています。

詳細については、「AWS Glue ユーザーガイド」の「Amazon Q data integration in AWS Glue」を参照してください。

サードパーティ統合

Amazon Q での GitLab デュオ の使用

GitLab Duo Amazon Q を使用すると、ソフトウェア開発オペレーションとソースコード管理ワークフローを実行できます。GitLab Duo で Amazon Q を設定したら、クイックアクションを呼び出して、アイデアのコードの実装、コードベースの変換、品質と脆弱性のマージリクエストの確認、ユニットテストの提案など、タスクを自動化できます。

詳細については、「GitLab Duo Amazon Q を使用した」を参照してください。

での Amazon Q Developer 機能の使用 GitHub

ソフトウェア開発ワークフローに Amazon Q Developer の機能を活用できます。特殊な開発エージェントを使用すると、新しいアイデアの実装、品質問題のコードのレビュー、ユニットテストによる脆弱性への対応、レガシー Java アプリケーションのモダナイズを行うことができます。

詳細については、「Amazon Q Developer for GitHub (プレビュー) 」を参照してください。

クラウド財務管理

コストを理解する

AWS 請求書とアカウントのコストについては、 AWS マネジメントコンソールで Amazon Q にお問い合わせいただけます。Amazon Q では、コストデータの取得、コストの説明、コスト傾向の分析を行うことができます。

詳細については、「コストに関するチャット」を参照してください。

カスタマーサポート

Amazon Q から直接カスタマーサポートを受ける

Amazon Q は、アカウントのアクティベーション、コストの急増、請求調整、不正イベント、ヘルスイベント、リソースの問題に関する質問に回答できます AWS 。

詳細については、「」およびコストに関するチャットリソースのトラブルシューティングを Amazon Q に依頼する「」を参照してください。

サポートチケットの作成

Amazon Q は、サポートケースを作成し、AWS のサポート担当者につなげるお手伝いをします。

詳細については、「Amazon Q Developer を使用して とチャットする サポート」を参照してください。

チャットアプリケーションの Amazon Q

Slack および Microsoft Teamsアプリケーションで Amazon Q をアクティブ化して、 の構築について質問できます AWS。Amazon Q をチャットアプリケーションに追加するには、チャットア

プリケーションでの Amazon Q Developer とのチャット「」を参照してください。詳細については、「Amazon Q Developer in chat applications Administrator Guide」の「Get started with Slack」および「Get started with Microsoft Teams」を参照してください。

Amazon Q Developer の使用開始

このセクションでは、Amazon Q Developer の使用開始に関する包括的なガイダンスを提供します。さまざまなサービス階層 (無料および Pro) について説明し、使用可能なさまざまなサインイン方法について説明し、個人アカウント (ビルダー IDs) とワークフォース ID のセットアッププロセスを順を追って説明します AWS IAM Identity Center。個々の開発者でも管理者でも、組織の Amazon Q Developer をセットアップしていても、このセクションは Amazon Q をすぐに使い始めるための適切なアプローチを選択するのに役立ちます。

トピック

- Q Developer のサービス階層 – 無料および Pro
- 個人アカウントの開始方法 (ビルダー ID)
- IAM Identity Center の開始方法
- Amazon Q Developer Pro サブスクリプション

Q Developer のサービス階層 – 無料および Pro

Amazon Q Developer では、Amazon Q Developer を無料または Pro 階層で使用します。以下の情報では、各階層で何が提供されているかを説明しています。

無料利用枠

Amazon Q Developer には、月額制限付きの永久無料利用枠があります。

無料利用枠で利用できる内容の詳細については、Amazon Q Developer の料金ページを参照してください。

無料利用枠は、個人アカウント (ビルダー IDsを持つユーザー、IAM アイデンティティセンターの ID を持つユーザー、IAM 認証情報を持つユーザーが利用できます。次の表を参照して、サインイン方法に応じて無料利用枠のさまざまなインターフェイスで Amazon Q が利用できるかどうかを確認してください。

	サインイン方法 (無料利用枠)		
インターフェイス	個人アカウント (ビルダー ID)	IAM アイデンティティセンター	IAM 認証情報
AWS Management Console、、および AWS アプリケーションとウェブサイト	いいえ	はい	はい
IDE	はい	いいえ	いいえ
[Command line]	はい	いいえ	いいえ

無料利用枠でサービスを改善するためにコンテンツがどのように使用されるかの詳細については、「」を参照してくださいAmazon Q Developer のサービスの改善。

Pro 階層

Pro 階層は、Amazon Q Developer Pro とも呼ばれ、無料階層よりも高い使用制限を提供する Amazon Q Developer サービスの有料バージョンです。また、カスタマイズなどの高度な機能にもアクセスできます。

Pro 階層の料金については、Amazon Q Developer の料金ページを参照してください。

Pro 階層は、個人アカウント (ビルダー IDsを持つユーザー、および IAM アイデンティティセンターで ID を持つユーザーが使用できます。次の表を参照して、サインイン方法に応じて Pro 階層のさまざまなインターフェイスで Amazon Q が使用可能かどうかを確認してください。

	サインイン方法 (Pro 階層)		
インターフェイス	個人アカウント (ビルダー ID)	IAM アイデンティティセンター	
AWS Management Console、、および AWS アプリケーションとウェブサイト	いいえ	はい	
IDE	はい	はい	
[Command line]	はい	はい	

個人アカウントの開始方法 (ビルダー ID)

個人プロジェクトに Amazon Q Developer を使用し、他のユーザーを管理する必要がない場合は、ビルダー ID とも呼ばれる個人アカウントの使用を開始する必要があります。Builder ID は、統合開発環境 (IDE) とターミナルのコマンドラインで Amazon Q を使用できるようにする特殊なタイプの AWS アカウントです。通常の AWS アカウントとは異なり、ビルダー ID はユーザー自身で使用することを意図しており、 へのアクセスを許可せず AWS Management Console、IAM ロールやアクセス許可を割り当てることはできません。

Builder ID は無料でセットアップできます。準備ができたら、 ビルダー ID を に接続して Pro 階層にアップグレード AWS アカウント し、より高い使用制限を活用できます。

Pro 階層で使用できる機能のリストについては、Amazon Q Developer の料金ページを参照してください。

無料または Pro 階層で個人アカウント (ビルダー ID) の使用を開始するには

- 開始する前に: 個人アカウント (ビルダー IDs) の制限を理解する

- ステップ 1: サインアップする
- ステップ 2: Amazon Q をインストールする
- ステップ 3: (オプション) Pro 階層へのアップグレード

開始する前に: 個人アカウント (ビルダー IDs) の制限を理解する

Amazon Q で使用する個人アカウント (ビルダー ID) を作成する前に、その制限を理解してください。

- ビルダー ID が無料利用枠にある場合、使用制限が適用されます。これらの制限の詳細については、「 料金表」ページを参照してください。使用制限を増やす必要がある場合は、以下の手順に従って Builder ID を Pro 階層にサブスクライブするか、「」のガイダンスに従って IAM Identity Center を使用しますIAM Identity Center の開始方法。
- ビルダー ID が Pro 階層の場合、使用制限は高くなりますが、カスタマイズなどの Pro 階層のみの機能は使用できません。利用できない Pro 階層機能のリストについては、Amazon Q Developer の料金ページの下部にある脚注を参照してください。Pro 階層機能が必要な場合は、IAM Identity Center を使用します。詳細については、「IAM Identity Center の開始方法」を参照してください。
- ビルダー ID が無料利用枠と Pro 利用枠の両方で、Amazon Q は IDE とコマンドラインでのみサポートされています。これは、、アプリ AWS Management Console、 AWS ウェブサイトではサポートされていません。および AWS アプリ AWS Management Consoleやウェブサイトで Q を使用する必要がある場合は、IAM Identity Center を使用します。詳細については、「IAM Identity Center の開始方法」を参照してください。

ステップ 1: サインアップする

無料の個人アカウント (ビルダー ID) にサインアップします。詳細については、AWS 「 サインインユーザーガイド」のAWS 「ビルダー ID の作成」を参照してください。

ステップ 2: Amazon Q をインストールする

Amazon Q を統合開発環境 (IDE) またはコマンドラインにインストールし、個人アカウント (ビルダー ID) を使用して認証します。インストールと認証情報については、以下を参照してください。

- IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする
- コマンドライン用の Amazon Q をインストールする

ステップ 3: (オプション) Pro 階層へのアップグレード

Pro 階層にアップグレードして、制限の引き上げを活用します。「個人アカウントのアップグレード (ビルダー ID)」を参照してください。

IAM Identity Center の開始方法

IAM Identity Center は、管理者がエンドユーザーの ID を管理するために使用されるサービスです。Amazon Q Developer のコンテキストでは、管理者は IAM Identity Center を使用して、Amazon Q Developer Pro へのサブスクライブを計画しているユーザーの ID を管理します。

IAM アイデンティティセンター、または IAM アイデンティティセンターに接続されているディレクトリまたはデータベースに ID を持つユーザーは、このガイドで IAM アイデンティティセンターのワークフォースユーザーと呼ばれます。

以下の場合は、IAM Identity Center の使用を開始する必要があります。

- Pro 階層で Amazon Q Developer を使用して複数のユーザーを設定する管理者です。IAM Identity Center を使用すると、ユーザーは Amazon Q Developer 機能の完全なスイートを取得し、管理する Amazon Q Developer サブスクリプションをエンタープライズに制御できます。たとえば、ユーザーのサブスクリプションをキャンセルしたり、ユーザーを一括サブスクライブしたり、ダッシュボードで Amazon Q の使用状況を追跡したりできます。
- お客様は個人ユーザーであり、その制限により個人アカウント (ビルダー ID) を使用することはできません。

IAM Identity Center の使用を開始するには、次の手順に従います。

IAM Identity Center の使用を開始するには

- ステップ 1: デプロイオプションを選択する
- ステップ 2: ワークフォースユーザーを Amazon Q Developer Pro にサブスクライブする

ステップ 1: デプロイオプションを選択する

ユーザーをサブスクライブする前に、作業する AWS アカウントを決定する必要があります。次の 3 つの重要な決定を行う必要があります。

- 決定事項 1: IAM アイデンティティセンターを有効にする場所 – IAM アイデンティティセンターの詳細については、 AWS IAM Identity Center ユーザーガイドの「IAM アイデンティティセンターとは」を参照してください。
- 決定事項 2: Amazon Q Developer プロファイルを作成する場所 – プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。
- 決定事項 3: ワークフォースユーザーをサブスクライブする場所 – サブスクリプションの詳細については、「」を参照してくださいAmazon Q Developer Pro サブスクリプション。

これらの 3 つの決定の特定の組み合わせは、デプロイオプションを構成します。

デプロイオプションについては、次の表で説明します。に進む前に、オプションを選択しますステップ 2: ワークフォースユーザーを Amazon Q Developer Pro にサブスクライブする。

この表では、次の用語を使用します：

- スタンドアロンアカウント — によって管理 AWS アカウント される組織の一部ではない AWS Organizations。
- 管理アカウント — によって管理 AWS アカウント される組織の一部である AWS Organizations。組織の最終的な所有者であり、組織内のアカウントによって発生したすべての料金を支払う責任があります。
- メンバーアカウント — によって管理される組織の一部である、管理アカウント AWS アカウント 以外の AWS Organizations。

デプロイオプション	説明	利点	欠点
デプロイオプション 1 (最も簡単) : スタンドアロンアカウントにデプロイする	エンドユーザーで、自分 (およびオプションで小規模なユーザーチーム) をサブスクライブして Amazon Q の機能をすばやく評価する場合は、このオプションを使用します。	デモに適しています。エンタープライズ全体の実装を行うことなく、Pro 階層機能を試すことができます。 個人アカウント (ビルダー IDsよりも多くの機能。詳細については、「Builder IDs の	機能が少ない IAM Identity Center はスタンドアロンアカウントで有効になっているため、組織インスタンスよりも機能が少ないアカウントインスタンスと見なされます[1]。

デプロイオプション	説明	利点	欠点
	このデプロイオプションでは、スタンドアロンアカウントで次の操作を行います。 • IAM アイデンティティセンターの有効化 • Amazon Q Developer プロファイルを作成する • 自分 (およびチームメンバー) をサブスクライブします。 詳細な手順については、「スタンドアロンアカウントで Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。	制限」を参照してください。	

デプロイオプション	説明	利点	欠点

デプロイオプション 2: 管理アカウントとメンバーアカウントにデプロイする	このオプションは、複数のユーザーの管理者である場合に使用します。
	
	このデプロイオプションでは、次のようになります。
	
	• 管理アカウントで、IAM Identity Center を有効にします。
	• メンバーアカウントでは、次の操作を行います。
	  • Amazon Q Developer プロファイルを作成する
	  • ユーザーをサブスクライブします。
	
	詳細な手順については、「メンバーアカウントの Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。
		その他の機能。IAM Identity Center は管理アカウントにインストールされるため、アカウントインスタンス2 よりも多くの機能を持つ組織インスタンスと見なされます。
		
		分散管理。サブスクリプション管理タスクはメンバーアカウントに分散されるため、ベストプラクティスです。
			複雑さ。複数の管理者によるアカウント間の調整が必要です。
			
			アカウントの制限。によって管理される組織ごとに AWS リージョン、最大 10 個のアカウントでユーザーをサブスクライブできますAWS Organizations。ユーザーベースが 1 つの組織で同じリージョン内の 10 を超えるアカウントに分散されている場合は、別のオプションを選択します。

デプロイオプション	説明	利点	欠点
デプロイオプション 3: メンバーアカウントにのみデプロイする	このオプションは、複数のユーザーの管理者である場合に使用します。 このデプロイオプションでは、メンバーアカウントで次の操作を行います。 • IAM アイデンティティセンターの有効化 • Amazon Q Developer プロファイルを作成する • ユーザーをサブスクライブします。 詳細な手順については、「メンバーアカウントの Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。	クイックセットアップ。個々のメンバーアカウント管理者は、エンタープライズ全体の実装を待ったり承認を必要とせずにデプロイできます。 複雑な組織向けの柔軟性。このオプションは、Pro 階層にサブスクライブするユーザーベース全体を含む統合 ID プロバイダーまたは ID ストアがない場合に使用します。	機能が少なくなります。IAM Identity Center はメンバーアカウントで有効になっているため、組織インスタンスよりも機能が少ないアカウントインスタンスと見なされます[1]。

デプロイオプション	説明	利点	欠点
デプロイオプション 4: 管理アカウントにのみデプロイする	このオプションは、複数のユーザーの管理者である場合に使用します。 このデプロイオプションでは、管理アカウントで次の操作を行います。 • IAM アイデンティティセンターの有効化 • Amazon Q Developer プロファイルを作成し、 • ユーザーをサブスクライブする • オプションで、メンバーアカウントと Amazon Q Developer プロファイルを共有します。 詳細な手順については、「管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。	その他の機能。IAM Identity Center は管理アカウントにインストールされるため、アカウントインスタンス2 よりも多くの機能を持つ組織インスタンスと見なされます。	ベストプラクティスに準拠していません。ユーザーは管理アカウントにサブスクライブされているため、委任管理がサポートされていない Amazon Q Developer の制限により、管理アカウント管理者はサブスクリプション管理タスクを処理する必要があります。メンバーアカウントにタスクを委任する推奨プラクティスに従うことはできません。

[1] アカウントインスタンスは、組織インスタンスよりも少ない機能をサポートします。例えば、アカウントインスタンスはアクセス許可セットをサポートしていません。つまり、ユーザーは やAWS アプリ AWS Management Console、ウェブサイトで Pro 階層サブスクリプションを使用できません。アカウントインスタンスの制限の一覧については、「 AWS IAM Identity Center ユーザーガイド」の「アカウントインスタンスに関する考慮事項」を参照してください。

[2] Organization インスタンスは、アカウントインスタンスと比較して、すべての IAM Identity Center 機能を含む幅広い機能を提供します。組織インスタンスでサポートされている機能のリストについては、AWS IAM Identity Center 「 ユーザーガイド」の「組織インスタンスを使用するタイミング」を参照してください。

ステップ 2: ワークフォースユーザーを Amazon Q Developer Pro にサブスクライブする

「」で説明されているようにデプロイオプションを選択したらステップ 1: デプロイオプションを選択する、ワークフォースユーザーをサブスクライブする準備が整います。ワークフォースユーザーのサブスクライブには、IAM アイデンティティセンターの有効化、Amazon Q Developer プロファイルの作成、ユーザーのサブスクライブの 3 つの主なステップがあります。すべてのステップを完了する方法については、以下の各セクションを参照してください。複数のアカウントでステップを実行する予定がある場合は、複数のセクションを読む必要がある場合があります。

- スタンドアロンアカウントで Amazon Q Developer Pro にユーザーをサブスクライブする
- 管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする
- メンバーアカウントの Amazon Q Developer Pro にユーザーをサブスクライブする

スタンドアロンアカウントで Amazon Q Developer Pro にユーザーをサブスクライブする

スタンドアロンアカウントは、 によって管理される組織の一部ではないアカウントですAWS Organizations。

スタンドアロンの所有者は AWS アカウント、以下の手順に従って Amazon Q Developer Pro にサブスクライブし、サービスの機能と機能を評価します。

このページのステップを完了したら、どのようなリソースが作成されましたか？最後に読んで、サブスクライブ時にユーザーに代わってインストールおよび設定されたリソースを確認してください。これにより、テストが終了したときにすべてをクリーンに削除できます。

前提条件

開始する前に、以下を確認してください。

- スタンドアロン AWS アカウントがあります。
- ユーザーをサブスクライブし、Amazon Q Developer 設定を管理するために必要な最小限のアクセス許可があります。詳細については「管理者に Amazon Q コンソールの使用を許可する」および「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。
- (オプション) スタンドアロンアカウントに IAM Identity Center のアカウントインスタンスがセットアップされている。この IAM アイデンティティセンターには、Amazon Q Developer Pro にサブスクライブするユーザーの ID が含まれており、「」で説明されているように、サポートされている AWS リージョンにデプロイする必要がありますAmazon Q Developer でサポートされている IAM Identity Center リージョン。IAM Identity Center インスタンスがインストールされていない場合は、問題ありません。最初のユーザー (自分自身) をサブスクライブすると、1 つの がインストールされます。IAM Identity Center の詳細については、AWS IAM Identity Center 「 ユーザーガイド」の「IAM Identity Center の組織インスタンスとアカウントインスタンス」を参照してください。

> **Note**
>
> このページの手順では、スタンドアロンアカウントに IAM Identity Center インスタンスがまだインストールされていないことを前提としています。

ステップ 1: Amazon Q Developer Pro プロファイルを作成し、自分でサブスクライブする

1. スタンドアロン AWS Management Console を使用して にサインインします AWS アカウント。ルートユーザーとして、または で説明されているアクセス許可を持つ IAM ユーザーとしてサインインします前提条件。
2. Amazon Q Developer コンソールに切り替えます。
3. Amazon Q Developer プロファイルを作成する AWS リージョン と、ユーザーデータを保存するにいることを確認します。サポートされているリージョンについては、「」を参照してくださいQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。
4. 開始方法ボタンを選択します。

> **Note**
>
> 開始方法ボタンの代わりに設定ボタンが表示された場合は、「開始方法」ワークフローをすでに実行しており、 にスキップできることを意味しますステップ 2: チームメンバーをサブスクライブする。

ユーザーの作成ダイアログボックスが表示されます。

5. 情報を入力します。E メールアドレスは、 へのサインアップに使用したものと同じでも異なっていてもかまいません AWS アカウント。

   [続行] をクリックしてください。

   Amazon Q Developer プロファイルの作成ダイアログボックスが表示されます。

6. ダイアログボックスの内容を確認し、プロファイル名にプロファイルの名前を指定します。クロスリージョン推論については、「」を参照してくださいAmazon Q Developer でのクロスリージョン処理。ダッシュボードメトリクスを無効にする方法については、「」を参照してくださいAmazon Q Developer ダッシュボードの無効化。

   [Create application] を選択します。

   Amazon Q Developer プロファイルとマネージドアプリケーションが作成され、サブスクリプションが作成されます。

7. (オプション) サブスクリプションが作成されたことを確認します。

   1. Amazon Q Developer コンソールのナビゲーションペインで、サブスクリプションを選択します。
   2. メインペインで、ユーザータブを選択します。

   サブスクリプションは、保留状態のリストに表示されます。そうでない場合は、ブラウザタブを更新します。

> **Note**
>
> Amazon Q Developer 機能を初めて使用した後、サブスクリプションはアクティブ状態に変更されます。

サブスクライブしたので、サブスクリプションをアクティブ化する必要があります。これは、次のセクションで説明するように、現在、またはチームメンバーをサブスクライブした後に実行できます。サブスクリプションをアクティブ化するには、受信トレイで「Invitation to join AWS IAM Identity Center and Activate Your Amazon Q Developer Pro Subscription」というタイトルの E メールを確認します。これらの E メールの指示に従って Amazon Q Developer Pro サブスクリプションをアクティブ化し、IDE で Amazon Q Developer Pro をセットアップします。これらの E メールは 24 時間以内に届きます。

ステップ 2: チームメンバーをサブスクライブする

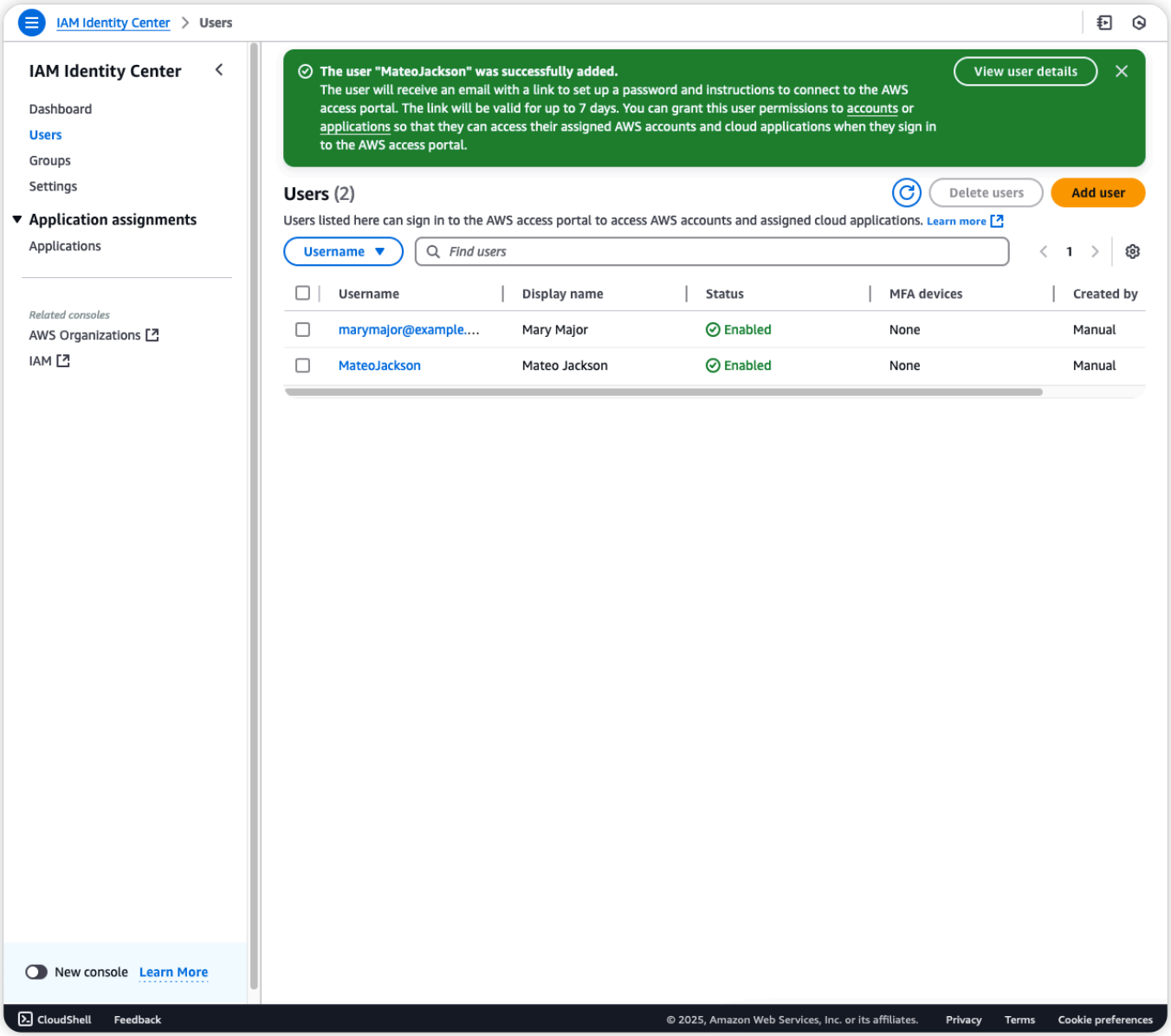
Amazon Q Developer Pro を試すことができるように、他のチームメンバーをサブスクライブすることもできます。サブスクライブするには、以下の手順に従います。

チームメンバーを追加するには

1. (IAM コンソールではなく) IAM Identity Center コンソールに切り替えます。

> **Note**
>
> IAM Identity Center は、サブスクライブ時にユーザーに代わってセットアップされました。セットアップされた IAM アイデンティティセンターの詳細については、「」を参照してください[どのようなリソースが作成されましたか？](#)。

2. ユーザーとグループを追加します。手順については、「 AWS IAM Identity Center ユーザーガイド」の「IAM Identity Center ディレクトリにユーザーを追加する」を参照してください。



3. 次のステップに進み、チームメンバーをサブスクライブします。

チームメンバーをサブスクライブするには

1. Amazon Q Developer コンソールに戻ります。
2. ナビゲーションペインで、サブスクリプションを選択し、サブスクリプションを選択します。

   ユーザーとグループの割り当てダイアログボックスが表示されます。
3. 追加したチームメンバーまたはグループの名前の入力を開始します。名前は自動的に入力されます。

> **Note**
>
> ダイアログボックスは、ユーザー名またはグループ名にのみ一致します。E メールアドレスでは一致しません。

4. [Assign (割り当てる)] を選択します。
5. ユーザーに E メールを確認してもらいます。Amazon Q Developer Pro サブスクリプションのアクティブ化というタイトルの E メールが 24 時間以内に届きます。この E メールには、 AWS Management Console および統合開発環境 (IDE) で Amazon Q Developer Pro ライセンスの使用を開始する方法に関するガイダンスが記載されています。E メールには、認証用のユーザー固有の開始 URL と AWS リージョンが含まれており、IDE で Amazon Q Developer を使用するためのクイックスタート手順が記載されています。この E メールにより、オンボーディングプロセスが合理化され、新しいユーザーごとに手動で通知する必要がなくなるため、貴重な時間を節約できます。

どのようなリソースが作成されましたか？

自分 (およびオプションでチームメンバー) をサブスクライブすると、Amazon Q はユーザーに代わって次の AWS リソースを作成しました。

- IAM Identity Center のアカウントインスタンス。IAM Identity Center のアカウントインスタンスの詳細については、 AWS IAM Identity Center ユーザーガイドの「IAM Identity Center のアカウントインスタンス」を参照してください。

> **Note**
>
> IAM Identity Center のアカウントインスタンスには制限があります。たとえば、アカウントインスタンスはコンソールアクセスをサポートしていません。(ユーザーは引き続きコンソールで Amazon Q を使用できますが、無料利用枠の月額制限が適用されます)。コンソールやその他の AWS ウェブサイトで Amazon Q Developer Pro を使用する場合は、管理アカウントの IAM アイデンティティセンターの組織インスタンスのユーザーである必要があります。詳細については、「管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。

> **Note**
>
> IAM Identity Center のアカウントインスタンスを組織インスタンスに変換またはマージすることはできません。

- IAM Identity Center の最初のユーザー。チームメンバーを手動で追加した可能性もあります。
- Amazon Q Developer の最初のユーザーとチームメンバーの Pro 階層サブスクリプション。
- Amazon Q Developer コンソールの「設定」にある Amazon Q Developer プロファイル。
- スタンドアロンアカウントで設定された IAM アイデンティティセンターの QDevProfile-***region*** と呼ばれるマネージドアプリケーション。アプリケーションは Amazon Q Developer プロファイルに関連付けられています。Amazon Q Developer プロファイルと同様に、アプリケーションは一度作成され、スタンドアロンアカウントのすべての Amazon Q サブスクライバー間で共有されます。

管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする

管理アカウントは、 によって管理 AWS アカウント される組織の一部である ですAWS Organizations。組織の最終的な所有者であり、組織内のアカウントによって発生したすべての料金を支払う責任があります。

管理アカウントの所有者は、以下の手順に従って、アカウントの Amazon Q Developer Pro にユーザーをサブスクライブします。

> **Note**
>
> 可能であれば、管理アカウントではなくメンバーアカウントにユーザーをサブスクライブします。詳細については、「ステップ 1: デプロイオプションを選択する」を参照してください。

組織と管理アカウントの詳細については、「 AWS Organizations ユーザーガイド」の「 の用語と概念 AWS Organizations」を参照してください。

前提条件

開始する前に、以下を確認してください。

- 管理 AWS アカウントがあります。
- ユーザーをサブスクライブし、Amazon Q Developer 設定を管理するために必要な最小限のアクセス許可があります。詳細については「管理者に Amazon Q コンソールの使用を許可する」および「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。
- 管理アカウントに IAM Identity Center の組織インスタンスがセットアップされている。この IAM アイデンティティセンターには、Amazon Q Developer Pro にサブスクライブするユーザーの ID が含まれており、「」で説明されているように、サポートされている AWS リージョンにデプロイする必要がありますAmazon Q Developer でサポートされている IAM Identity Center リージョン。IAM アイデンティティセンターの詳細については、AWS IAM Identity Center 「 ユーザーガイド」の「IAM アイデンティティセンターの組織インスタンス」を参照してください。

ステップ 1: Amazon Q Developer プロファイルを作成する

1. AWS 管理アカウント AWS Management Console を使用して にサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. Amazon Q Developer プロファイルを作成する AWS リージョン と、ユーザーデータを保存するにいることを確認します。サポートされているリージョンについては、「」を参照してくださいQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。
4. [開始する] を選択します。

   Amazon Q Developer プロファイルの作成ダイアログボックスが表示されます。
5. ダイアログボックスの内容を確認し、プロファイル名にプロファイルの名前を指定します。以下のヘルプを参照してください。

   - クロスリージョン推論については、「」を参照してくださいAmazon Q Developer でのクロスリージョン処理。
   - Amazon Q Developer 設定をメンバーアカウントと共有チェックボックスについては、Amazon Q Developer でのプロファイル共有の有効化「」および「」を参照してくださいステップ 1: デプロイオプションを選択する。
   - ダッシュボードメトリクスを無効にするには、「」を参照してくださいAmazon Q Developer ダッシュボードの無効化。

   [Create application] を選択します。

   Amazon Q Developer プロファイルとマネージドアプリケーションが作成されます。

ステップ 2: ユーザーをサブスクライブする

1. Amazon Q Developer コンソールのナビゲーションペインで、サブスクリプションを選択します。
2. [サブスクライブ] を選択します。

   ユーザーとグループの割り当てダイアログボックスが表示されます。
3. サブスクライブするグループまたはユーザーの入力を開始します。グループまたはユーザーは、管理アカウントで設定された IAM Identity Center で使用可能なものを自動的に入力します。

   > **Note**
   >
   > ダイアログボックスは、ユーザー名またはグループ名にのみ一致します。E メールアドレスでは一致しません。
4. [Assign (割り当てる)] を選択します。
5. ユーザーに E メールを確認してもらいます。Amazon Q Developer Pro ライセンスの使用を開始する方法については、24 時間以内に「Amazon Q Developer Pro サブスクリプションのアクティブ化」というタイトルの E メールが届きます。

ステップ 3: ID 拡張コンソールセッションを有効にする

ユーザーが およびAWS アプリケーション AWS Management Consoleやウェブサイトで Amazon Q Developer Pro サブスクリプションを使用できるようにする場合は、アイデンティティ強化コンソールセッションを有効にします。詳細については、「 AWS IAM Identity Center ユーザーガイド」の「ID 拡張コンソールセッションの有効化」を参照してください。

> **Note**
>
> ID 拡張コンソールセッションを有効にしない場合でも、ユーザーは AWS Management Consoleおよび AWS アプリやウェブサイトで Amazon Q を使用できますが、無料利用枠に制限されます。

どのようなリソースが作成されましたか？

Amazon Q Developer プロファイルを作成し、管理アカウントにユーザーをサブスクライブすると、Amazon Q はユーザーに代わって次のリソースを作成しました。

- Amazon Q Developer でのユーザーの Pro 階層サブスクリプション。
- Amazon Q Developer コンソールの「設定」にある Amazon Q Developer プロファイル。
- 管理アカウントで設定された IAM アイデンティティセンターの QDevProfile-***region*** と呼ばれるマネージドアプリケーション。アプリケーションは Amazon Q Developer プロファイルに関連付けられています。Amazon Q Developer プロファイルと同様に、アプリケーションは 1 回作成され、管理アカウントのすべての Amazon Q サブスクライバー間で共有されます。

> **Note**
>
> Amazon Q は、組織内で QDevProfile-***region*** マネージドアプリケーションを最大 10 AWS アカウント 個作成できます。 AWS リージョン

メンバーアカウントの Amazon Q Developer Pro にユーザーをサブスクライブする

メンバーアカウントは AWS アカウント、 によって管理される組織の一部である管理アカウント以外の ですAWS Organizations。

メンバーアカウントの所有者は、以下の手順に従って、アカウントの Amazon Q Developer Pro にユーザーをサブスクライブします。

メンバーアカウントまたは管理アカウントのユーザーをサブスクライブするかどうかわからない場合ヘルプステップ 1: デプロイオプションを選択するについては、「」を参照してください。

組織、メンバーアカウント、管理アカウントの詳細については、AWS Organizations 「 ユーザーガイド」の「 の用語と概念 AWS Organizations」を参照してください。

前提条件

開始する前に、以下を確認してください。

- メンバー AWS アカウントがあります。
- ユーザーをサブスクライブし、Amazon Q Developer 設定を管理するために必要な最小限のアクセス許可があります。詳細については「管理者に Amazon Q コンソールの使用を許可する」および「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。
- (オプション) 管理アカウントに設定された IAM アイデンティティセンターの組織インスタンス、またはメンバーアカウントに設定された IAM アイデンティティセンターのアカウントインスタンスがある。この IAM Identity Center インスタンスには、Amazon Q Developer Pro にサブスク

ライブするユーザーの ID が含まれており、「」で説明されているように、サポートされている AWS リージョンにデプロイする必要がありますAmazon Q Developer でサポートされている IAM Identity Center リージョン。IAM Identity Center インスタンスがインストールされていない場合は、問題ありません。最初のユーザーをサブスクライブすると、メンバーアカウントに 1 つの がインストールされます。IAM Identity Center の詳細については、AWS IAM Identity Center 「 ユーザーガイド」の「IAM Identity Center の組織インスタンスとアカウントインスタンス」を参照してください。

ステップ 1: Amazon Q Developer Pro プロファイルを作成し、最初のユーザーをサブスクライブする

1. メンバー AWS Management Console を使用して にサインインします AWS アカウント。
2. Amazon Q Developer コンソールに切り替えます。
3. Amazon Q Developer プロファイルを作成する AWS リージョン と、ユーザーデータを保存するにいることを確認します。サポートされているリージョンについては、「」を参照してくださいQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。
4. 開始方法ボタンを選択します。

> **Note**
>
> 開始方法ボタンの代わりに設定ボタンが表示された場合は、「開始方法」ワークフローをすでに実行しており、 にスキップできることを意味しますステップ 2: 他のユーザーをサブスクライブする。

5. 画面上のプロンプトに従って、最初のユーザーをサブスクライブします。

   - 最初のユーザーの E メールアドレスがメンバーアカウントまたは管理アカウントの既存の IAM アイデンティティセンターのものと一致する場合、Amazon Q はその IAM アイデンティティセンターに接続します。
   - 最初のユーザーの E メールアドレスが既存の IAM アイデンティティセンターの E メールアドレスと一致しない場合、Amazon Q はメンバーアカウントに IAM アイデンティティセンターアカウントインスタンスを作成し、最初のユーザーを追加します。以下の点に注意してください。
     - Amazon Q は、メンバーアカウントに既に IAM Identity Center がない場合にのみ、IAM Identity Center アカウントインスタンスを作成します。

- メンバーアカウントに IAM Identity Center アカウントインスタンスがあっても、そのインスタンスにユーザーがいない場合、Amazon Q は既存の IAM Identity Center にユーザーを作成します。

Amazon Q Developer プロファイルの作成ダイアログボックスが表示されます。

6. ダイアログボックスの内容を確認し、プロファイル名にプロファイルの名前を指定します。クロスリージョン推論については、「」を参照してくださいAmazon Q Developer でのクロスリージョン処理。ダッシュボードメトリクスを無効にする方法については、「」を参照してくださいAmazon Q Developer ダッシュボードの無効化。

   [Create application] を選択します。

   Amazon Q Developer プロファイルとマネージドアプリケーションが作成され、最初のユーザーがサブスクライブされます。

7. (オプション) 最初のユーザーのサブスクリプションが作成されていることを確認します。

   1. Amazon Q Developer コンソールのナビゲーションペインで、サブスクリプションを選択します。
   2. メインペインで、ユーザータブを選択します。

   最初のユーザーのサブスクリプションは、保留状態のリストに表示されます。そうでない場合は、ブラウザタブを更新します。

   > **Note**
   >
   > サブスクリプションは、ユーザーが Amazon Q Developer 機能を初めて使用した後、アクティブ状態に変更されます。

8. 最初のユーザーに E メールを確認してもらいます。Amazon Q Developer Pro サブスクリプションのアクティブ化というタイトルの E メールが 24 時間以内に届きます。この E メールには、AWS Management Console および統合開発環境 (IDE) で Amazon Q Developer Pro ライセンスの使用を開始する方法に関するガイダンスが記載されています。E メールには、認証用のユーザー固有の開始 URL と AWS リージョンが含まれており、IDE で Amazon Q Developer を使用するためのクイックスタート手順が記載されています。この E メールにより、オンボーディングプロセスが合理化され、新しいユーザーごとに手動で通知する必要がなくなるため、貴重な時間を節約できます。

ステップ 2: 他のユーザーをサブスクライブする

他のユーザーをサブスクライブするには、まだ IAM アイデンティティセンターインスタンスに追加してから、Amazon Q Developer コンソールで Subscribe を選択して Amazon Q Developer Pro にサブスクライブします。

IAM Identity Center にユーザーを追加する手順については、 AWS IAM Identity Center ユーザーガイドの「IAM Identity Center ディレクトリにユーザーを追加する」を参照してください。

ステップ 3: ID 拡張コンソールセッションを有効にする

ユーザーが およびAWS アプリケーション AWS Management Consoleやウェブサイトで Amazon Q Developer Pro サブスクリプションを使用できるようにする場合は、ID 拡張コンソールセッションを有効にします。詳細については、「 AWS IAM Identity Center ユーザーガイド」の「ID 拡張コンソールセッションの有効化」を参照してください。

ID 拡張コンソールセッションを有効にしない場合でも、ユーザーは AWS Management Consoleおよび AWS アプリやウェブサイトで Amazon Q を使用できますが、無料利用枠に制限されます。

> **Note**
>
> ID 拡張コンソールセッションを有効にする機能、つまり および AWS アプリケーションやウェブサイトで Amazon Q Developer Pro サブスクリプションを使用する機能は AWS Management Console、アカウントインスタンスではなく、IAM Identity Center の組織インスタンスでのみサポートされます。

どのようなリソースが作成されましたか？

メンバーアカウントのユーザーをサブスクライブすると、Amazon Q はユーザーに代わって次の AWS リソースを作成しました。

- IAM Identity Center のアカウントインスタンス。このインスタンスは、サブスクライブした最初のユーザーがメンバーアカウントまたは管理アカウントの既存の IAM アイデンティティセンターで見つからない場合にのみ作成されます。IAM Identity Center のアカウントインスタンスの詳細については、 AWS IAM Identity Center ユーザーガイドの「IAM Identity Center のアカウントインスタンス」を参照してください。

> **Note**
>
> IAM Identity Center のアカウントインスタンスには制限があります。たとえば、アカウントインスタンスはコンソールアクセスをサポートしていません。(ユーザーは引き続きコンソールで Amazon Q を使用できますが、無料利用枠の月額制限が適用されます)。ユーザーがコンソールやその他の AWS ウェブサイトで Amazon Q Developer Pro を使用できるようにするには、IAM Identity Center の組織インスタンスの管理アカウントに存在する必要があります。詳細については、「管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする」を参照してください。

> **Note**
>
> IAM Identity Center のアカウントインスタンスを組織インスタンスに変換またはマージすることはできません。

- IAM Identity Center の最初のユーザー。(チームメンバーも追加されている可能性があります)。
- Amazon Q Developer の最初のユーザーと他のユーザーの Pro 階層サブスクリプション。
- Amazon Q Developer コンソールの「設定」にある Amazon Q Developer プロファイル。
- IAM アイデンティティセンターの QDevProfile-***region*** と呼ばれるマネージドアプリケーション。アプリケーションは Amazon Q Developer プロファイルに関連付けられています。Amazon Q Developer プロファイルと同様に、アプリケーションは一度作成され、メンバーアカウントのすべての Amazon Q Developer Pro サブスクライバー間で共有されます。

> **Note**
>
> Amazon Q は、組織内で QDevProfile-***region*** マネージドアプリケーションを最大 10 AWS アカウント 個作成できます。 AWS リージョン

Amazon Q Developer Pro サブスクリプション

Pro 階層サブスクリプションとも呼ばれる Amazon Q Developer Pro サブスクリプションは、Amazon Q Developer サービスの有料バージョンです。 高度な機能と無料利用枠を超えた使用制限を必要とするプロフェッショナルデベロッパーやチーム向けに設計された、強化された AI 開発

機能を提供します。料金階層と利用可能な機能の詳細については、Amazon Q Developer の料金ページを参照してください。

この章では、個人アカウント (ビルダー IDs) とワークフォースユーザー (IAM アイデンティティセンター) の両方の Amazon Q Developer Pro サブスクリプションを管理するための重要な情報について説明します。リージョンの可用性とサポート、請求の詳細、サブスクリプションステータス情報について説明します。また、サブスクリプション解除や Pro 階層へのアップグレードなどの主要なタスクに関するstep-by-stepガイダンスも提供します。

トピック

- Amazon Q Developer Pro リージョンのサポート
- Amazon Q Developer Pro サブスクリプションの請求
- Amazon Q Developer サブスクリプションのステータス
- Amazon Q Developer で使用する開始 URL の検索
- Amazon Q Developer での暗号化方法の管理
- Amazon Q Developer プロファイルとは
- Amazon Q Developer Pro サブスクリプションのトラブルシューティング
- Amazon Q Developer サブスクリプションの集約リストの表示
- Amazon Q Developer Pro からのサブスクリプション解除
- Amazon Q Developer Pro へのアップグレード

Amazon Q Developer Pro リージョンのサポート

Pro 階層のリージョン情報は、個人アカウント (ビルダー ID) を持つエンドユーザーであるか、IAM Identity Center ワークフォースユーザーの管理者であるかによって異なります。

個人アカウント (ビルダー ID) ユーザー

個人アカウント (ビルダー ID) の所有者である場合、Pro 階層サブスクリプションは次のリージョンでサポートされています。

- 米国東部 (バージニア北部)

IAM Identity Center ワークフォースユーザー

IAM Identity Center ワークフォースユーザーの管理者の場合は、このセクションをお読みください。

トピック

- Amazon Q Developer でサポートされている IAM Identity Center リージョン
- Q Developer コンソールと Q Developer プロファイルでサポートされているリージョン
- AWS リージョン間で Amazon Q Developer Pro にユーザーをサブスクライブする

Amazon Q Developer でサポートされている IAM Identity Center リージョン

Amazon Q Developer Pro にサブスクライブするワークフォースユーザーは、オプトインリージョンを除き、サポートされているリージョンページに記載されているいずれかのリージョンの IAM アイデンティティセンターインスタンス (または接続された ID プロバイダー) に ID を持っている必要があります。ユーザーがオプトインリージョンの IAM Identity Center インスタンスに ID を持っている場合、サブスクライブすることはできません。つまり、ユーザーは 、 AWS アプリ AWS Management Console、ウェブサイトの無料利用枠にのみアクセスでき、IDE またはコマンドラインで Amazon Q にアクセスすることはできません。

Amazon Q は、IAM Identity Center ワークフォースユーザーのサブスクリプションを IAM Identity Center インスタンスと同じリージョンに保存します。

IAM アイデンティティセンターのリージョンに関係なく、データは Amazon Q Developer プロファイルを作成するリージョンに保存されます。

Amazon Q Developer プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。

データ保護の詳細については、「Amazon Q Developer でのデータ保護」を参照してください。

Q Developer コンソールと Q Developer プロファイルでサポートされているリージョン

Amazon Q Developer コンソールと Amazon Q Developer プロファイルは、次のリージョンでサポートされています。

- 米国東部 (バージニア北部)
- 欧州 (フランクフルト)

Amazon Q Developer プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。

> **Note**
>
> 欧州 (フランクフルト) リージョンで作成された Q Developer プロファイルでは、以下の機能はサポートされていません。
>
> - とのチャット サポート
> - Amazon Q を使用したリソースのトラブルシューティング
> - IDE での .NET 変換
> - Amazon Q in チャットアプリケーション (Slack 用)
> - の「Amazon Q AWS Console Mobile Application」

AWS リージョン間で Amazon Q Developer Pro にユーザーをサブスクライブする

IAM Identity Center ワークフォースユーザーを Amazon Q Developer Pro にサブスクライブする場合、次の 2 つの異なる AWS リージョンで作業する必要がある場合があります。

- IAM Identity Center インスタンスの 1 つのリージョン (ユーザー ID が管理され、サブスクリプションが保存されるリージョン)
- Amazon Q Developer コンソールの別のリージョン (Amazon Q Developer プロファイル、カスタマイズ、サブスクリプションが管理されるリージョン)

Amazon Q Developer コンソールが IAM Identity Center よりも少数のリージョンでサポートされているため、リージョンは常に同じとは限りません。

IAM アイデンティティセンターが Amazon Q Developer コンソールとは異なるリージョンにあるシナリオでは、次の例のガイダンスを使用してユーザーをサブスクライブします。

マルチリージョンシナリオのサブスクリプションプロセスの例

次の場合にユーザーをサブスクライブする手順を説明します。

- IAM Identity Center インスタンスは米国西部 (北カリフォルニア) にあります。
- Amazon Q Developer コンソールは米国東部 (バージニア北部) にあります。これは、Amazon Q Developer コンソールがサポートする IAM Identity Center インスタンスに最も近いリージョンです。

ユーザーをサブスクライブするには

1. 米国西部 (北カリフォルニア) の IAM アイデンティティセンターインスタンスにユーザーを追加します。
2. 米国東部 (バージニア北部) の Amazon Q Developer コンソールに切り替えます。
3. 米国東部 (バージニア北部) の Amazon Q Developer コンソールを使用してユーザーをサブスクライブします。

サブスクライブ時 :

- ユーザーのサブスクリプションは米国西部 (北カリフォルニア) で作成されます。
- ユーザーのサブスクリプションは、米国西部 (北カリフォルニア) のユーザーエントリに関連付けられています。
- ユーザーのサブスクリプションは、米国東部 (バージニア北部) の Amazon Q Developer プロファイルに関連付けられています。

さらに、Amazon Q Developer がユーザーに代わって保存する必要があるデータは、米国東部 (バージニア北部) に保存されます。データストレージとセキュリティの詳細については、「」を参照してください保管中の暗号化。

ユーザーをサブスクライブする詳細な手順については、「」を参照してくださいIAM Identity Center の開始方法。

Amazon Q Developer Pro サブスクリプションの請求

Pro 階層の請求情報は、個人アカウント (ビルダー ID) を持つエンドユーザーであるか、IAM Identity Center ワークフォースユーザーの管理者であるかによって異なります。

個人アカウント (ビルダー ID) ユーザー

個人アカウント (ビルダー ID) で Pro 階層にサブスクライブした場合、毎月課金されます。Builder ID にリンク AWS アカウント されている が請求書を取得します。

サブスクライブした最初の月には、日割り計算された料金が請求されます。たとえば、4 月 15 日にサブスクライブすると、サブスクリプション料金の半分が課金されます。その後、全額が請求されます。

サブスクリプションを解除すると、請求サイクルの終了時に請求が停止します。詳細については、「Amazon Q Developer Pro からのサブスクリプション解除」を参照してください。

IAM Identity Center ワークフォースユーザー

一連の IAM Identity Center ワークフォースユーザーを Pro 階層にサブスクライブしている管理者には、サブスクライブしているユーザーごとに毎月課金されます。詳細については、「Amazon Q Developer の料金」を参照してください。

ビジネスで AWS Organizationsが設定されている場合、Amazon Q Developer Pro の使用量の請求は AWS 組織ごとに行われます。管理アカウントは請求書を取得します。同じユーザーが同じ組織内の複数のアカウントで Amazon Q Developer にサブスクライブしている場合、二重請求されることはありません。

AWS Organizations をセットアップしていない場合、ユーザーがサブスクライブしている AWS アカウント が請求を受け取ります。

請求とコスト管理コンソールで請求を表示できます。Amazon Q の経費は、Q の「サービス別の料金」タブに表示されます。 Billing and Cost Management コンソールの詳細については、AWS Billing 「 ユーザーガイド」の「What is AWS Billing and Cost Management？」を参照してください。

AWS Billing and Cost Managementを通じて、リソース ID を持つ特定のユーザーの Amazon Q サブスクリプションのコストを特定できます。これを行うには、Billing and Cost Management コンソールの [データエクスポート] で [リソース ID を含める] オプションを選択し、標準データエクスポートまたはレガシー CUR エクスポートを作成します。詳細については、「 AWS Data Exports ユーザーガイド」の「データエクスポートの作成」を参照してください。

ユーザーのサブスクリプションを解除すると、請求サイクルの終了時に請求が停止します。詳細については、「Amazon Q Developer Pro からのサブスクリプション解除」を参照してください。

Amazon Q Developer サブスクリプションのステータス

サブスクリプションのステータス情報は、個人アカウント (ビルダー ID) を持つエンドユーザーであるか、IAM Identity Center ワークフォースユーザーの管理者であるかによって異なります。

個人アカウント (ビルダー ID) ユーザー

個人アカウント (ビルダー ID) を持つユーザーは、Amazon Q Developer コンソールのサブスクリプションページで Amazon Q Developer Pro サブスクリプションのステータスを表示できます。

サブスクリプションの可能なステータスは次のとおりです。

- アクティブ – Amazon Q Developer 機能を使用してサブスクリプションをアクティブ化しました。サブスクリプションに対して課金されます。
- キャンセル済み – Pro 階層からサブスクリプションを解除してサブスクリプションをキャンセルしました。Amazon Q Developer の機能と制限にアクセスできなくなりました。詳細については、「Amazon Q Developer Pro からのサブスクリプション解除」を参照してください。

IAM Identity Center ワークフォースユーザー

IAM Identity Center ワークフォースユーザーのセットを Pro 階層にサブスクライブしている管理者は、Amazon Q Developer コンソールのサブスクリプションページでユーザーのサブスクリプションのステータスを表示できます。

ステータスは、グループタブとユーザータブのどちらを表示するかによって若干異なります。

Groups タブのステータスは次のとおりです。

- Subscribed – グループは Amazon Q Developer Pro にサブスクライブされています。グループ内のアクティブなユーザーサブスクリプションに対して課金されます。
- キャンセル済み – グループは管理者によってキャンセル (サブスクリプション解除) されました。グループのユーザーは Amazon Q Developer Pro 機能にアクセスできなくなります。詳細については、「Amazon Q Developer Pro からのサブスクリプション解除」を参照してください。

ユーザータブのステータスは次のとおりです。

- アクティブ – ユーザーは Amazon Q Developer 機能を使用してサブスクリプションをアクティブ化しました。このサブスクリプションに対して課金されます。
- 保留中 – ユーザーはサブスクライブされていますが、サブスクリプションをアクティブ化していません。このサブスクリプションには課金されません。
- キャンセル済み – ユーザーのサブスクリプションは管理者によってキャンセル (サブスクリプション解除) され、ユーザーは Amazon Q Developer 機能にアクセスできなくなります。詳細については、「Amazon Q Developer Pro からのサブスクリプション解除」を参照してください。

> **Note**
>
> Amazon Q Developer コンソールのユーザータブには、グループの一部としてサブスクライブされているユーザーが表示されません。これらのユーザーを表示するには、Amazon Q コンソールの (Amazon Q Developer コンソールではなく) サブスクリプションページ

に移動します。このページには、グループサブスクライブしているユーザーのステータスが利用不可と表示されます。実際のステータスを確認するには、テーブルからユーザーを選択し、ユーザーの関連付けでそのステータスを探します。

Amazon Q Developer で使用する開始 URL の検索

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

一連の IAM Identity Center ワークフォースユーザーを Pro 階層にサブスクライブしている管理者の場合、それらのユーザーは IDE または IAM Identity Center の開始 URL とリージョンを使用してコマンドラインで Amazon Q にサインインする必要があります。この URL をユーザーに提供する必要がある場合は、Amazon Q Developer コンソールの設定ページで確認できます。

開始 URL を検索するには

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。

   Amazon Q Developer コンソールを使用するには、 管理者に Amazon Q Developer コンソールの使用を許可する で定義されているアクセス許可が必要です。
3. [設定] を選択します。

   開始 URL は、ページの上部付近の開始 URL に表示されます。

Amazon Q Developer での暗号化方法の管理

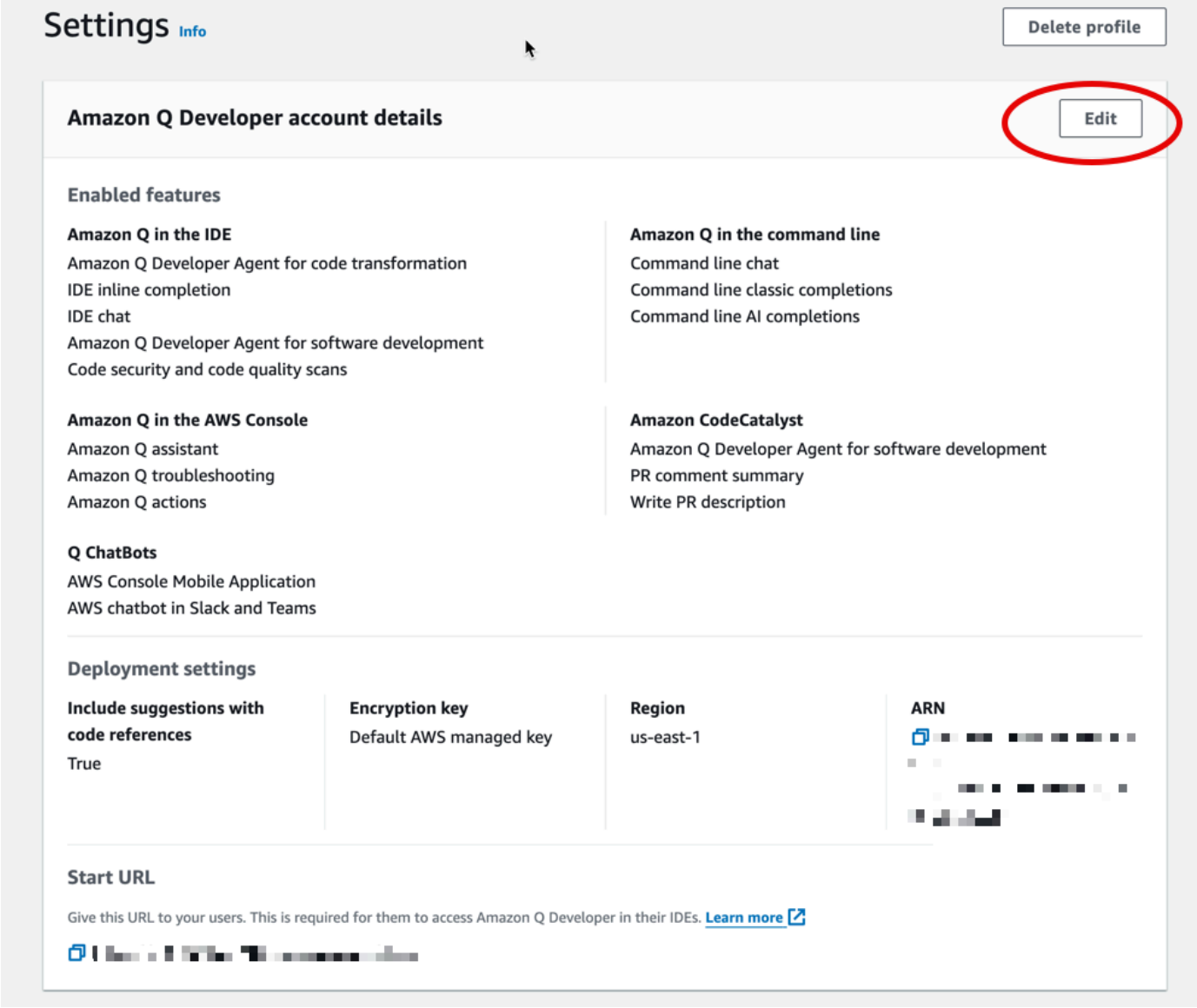
> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

デフォルトでは、Amazon Q Developer は暗号化に AWS マネージドキーを使用します。一部の機能では、カスタマーマネージドキーを設定してデータを暗号化できます。カスタマーマネージドキーによる暗号化をサポートする機能のリストについては、「データ暗号化」を参照してください。

暗号化に使用するキーを設定するには、次の手順を実行します。

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。

   Amazon Q Developer コンソールを使用するには、 管理者に Amazon Q Developer コンソールの使用を許可する で定義されているアクセス許可が必要です。
3. [設定] を選択します。
4. Amazon Q Developer アカウントの詳細パネルで編集を選択します。



5. 詳細の編集ページで、暗号化キー - オプションセクションを展開します。

6. 暗号化にカスタマーマネージドキーを使用するには、暗号化設定をカスタマイズ (詳細) を選択します。
7. 表示される検索バーで、暗号化に使用するキーの名前を検索するか、キー ARN を入力します。

   キーをまだ作成していない場合は、 AWS KMS キーの作成を選択し、このページに戻ってキーを追加します。
8. カスタマーマネージドキーによる暗号化を無効にし、暗号化のために AWS マネージドキーに戻すには、暗号化設定のカスタマイズ (アドバンスド) の選択を解除します。

Amazon Q Developer プロファイルとは

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

Amazon Q Developer プロファイルは、設定プロファイルとも呼ばれ、一連の IAM Identity Center ワークフォースユーザー Pro 階層サブスクリプションに関連付けられた Amazon Q Developer 設定のコレクションです。プロファイルは、IAM アイデンティティセンターのユーザーの ID を Amazon Q Developer のサブスクリプションに関連付ける Amazon Q Developer マネージドアプリケーションにも関連付けられます。

ユーザーを初めてサブスクライブすると、このプロファイルを作成するように求められます。プロファイルを作成すると、Amazon Q Developer Pro 機能を設定できる Amazon Q Developer コンソールのサイドナビゲーションに複数のページが表示されます。アカウントに (最初のサブスクリプションプロセス中およびそれ以降に) 追加したサブスクリプションはすべて、このプロファイルに関連付けられます。

Amazon Q Developer プロファイルのその他の特性は次のとおりです。

- プロファイルは、IAM Identity Center ワークフォースユーザーでの使用に必須です。ワークフォースユーザーをサブスクライブすることはできません。これは、Amazon Q Developer Pro にユーザーをサブスクライブする AWS アカウントで作成する必要があります。
- プロファイルは、サポートされているアカウントごとに 1 AWS リージョン回作成できます AWS 。Amazon Q Developer プロファイルでサポートされている AWS リージョンのリストについては、「」を参照してくださいQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。

Amazon Q Developer プロファイルの作成

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

Amazon Q Developer プロファイルを作成すると、Amazon Q Developer コンソールの設定がロック解除され、Amazon Q Developer Pro にユーザーをサブスクライブするための前提条件となります。プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。

Amazon Q Developer プロファイルを作成するには

- 「」の指示に従ってユーザーをサブスクライブしますIAM Identity Center の開始方法。サブスクリプションプロセス中に、プロファイルを作成するように求められます。

Amazon Q Developer プロファイルの削除

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

Amazon Q Developer プロファイルを削除して、すべてのサブスクリプションをすばやくキャンセルし、 からすべての Q Developer 設定を削除できます AWS アカウント。Amazon Q Developer プロファイルを削除すると、次のようになります。

- キャンセル済みとマークされたプロファイルに関連付けられているすべてのサブスクリプション、およびユーザーは Amazon Q Developer 機能にアクセスできなくなります。最終的な月額サブスクリプション料金は、アクティブなサブスクリプションを持つすべてのユーザーの現在の請求サイクルの終了時に課金されます。1 か月分の料金が請求されます。料金は日割り計算されません。サブスクリプションは、月末まで Amazon Q Developer コンソールに表示され続けます。月末になると、表示から削除されます。
- プロファイルの作成の結果として使用可能になった Amazon Q Developer コンソールのすべての設定とオプションは、表示または反映されなくなります。たとえば、Q Developer ダッシュボードは表示されなくなり、カスタマイズは設定または適用されなくなり、ユーザーアクティビティレポートは設定または生成されなくなります。

- マネージドアプリケーション (QDevProfile-*region* と呼ばれる) は、Amazon Q Developer に接続されている IAM アイデンティティセンターインスタンスから削除されます。（ この IAM Identity Center インスタンスは、デプロイオプションによっては、プロファイルが削除されているアカウントとは異なる場合があります ）。

> **Note**
>
> プロファイルを誤って削除した場合：
>
> - 再作成してから、ユーザーを再サブスクライブする必要があります。また、Amazon Q Developer コンソールを使用して以前に設定した設定もリセットする必要があります。IAM Identity Center インスタンスは残るため、ユーザー ID を再作成する必要はありません。
> - プロファイルを再作成した後、Amazon Q Developer ダッシュボードに履歴データを表示することはできません。新しいプロファイルの作成日以降のデータのみを表示できます。

Amazon Q Developer プロファイルを削除するには、次の手順に従います。

[開始する前に]

- プロファイルを正常に削除できるように作成したカスタマイズをすべて削除します。

Amazon Q Developer プロファイルを削除するには

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。

   Amazon Q Developer コンソールを使用するには、 で定義されているアクセス許可が必要です管理者に Amazon Q Developer コンソールの使用を許可する。
3. [設定] を選択します。
4. ページの上部で、プロファイルの削除を選択します。

Amazon Q Developer でのプロファイル共有の有効化

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

が管理する組織内の管理アカウント管理者はAWS Organizations、プロファイル共有機能を有効にできます。プロファイル共有を有効にすると、管理アカウントで作成された Amazon Q Developer プロファイルがメンバーアカウントと共有されます。プロファイルを共有すると、管理アカウントで Amazon Q Developer Pro にサブスクライブしている IAM Identity Center ワークフォースユーザーが、メンバーアカウントにサインインしている間にAWS Management Console、 および AWS アプリケーションやウェブサイトで Amazon Q Developer Pro サブスクリプションを使用できるようになります。プロファイル共有が無効になっている場合、これらのユーザーはメンバーアカウントにサインインしている間も AWS Management Console、 および AWS アプリやウェブサイトで Amazon Q を引き続き使用できますが、無料利用枠の制限と機能が適用されます。

プロファイル共有を有効にしても、統合開発環境 (IDE) またはコマンドラインで Amazon Q を使用するユーザーの機能には影響しません。 ???

プロファイル共有を有効にするには、次の手順に従います。

前提条件

開始する前に、以下を確認してください。

- お客様は AWS 管理アカウントの管理者です。
- 管理アカウントに IAM Identity Center インスタンスが設定され、Amazon Q に接続されている。確認するには、管理アカウントにサインインし、Amazon Q Developer コンソールに移動して、設定を選択し、開始 URL が表示されることを確認します。
- 管理アカウントで Amazon Q Developer Pro にユーザーをサブスクライブしています。
- Amazon Q Developer コンソールにアクセスするために必要な最小限のアクセス許可があります。詳細については、「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。

プロファイル共有を有効にするには

1. AWS 管理アカウント AWS Management Console を使用して にサインインします。

2. Amazon Q Developer コンソールに切り替えます。
3. [設定] を選択します。
4. メンバーアカウント設定セクションまでスクロールし、編集を選択します。
5. Q Developer マネージドアプリケーションと設定プロファイルを有効にします。
6. [保存] を選択します。

管理アカウントで Amazon Q Developer Pro にサブスクライブしているユーザーは AWS Management Console、メンバーアカウントにサインインしている間、 および AWS アプリやウェブサイトで Amazon Q Developer Pro サブスクリプションを使用できるようになりました。

Amazon Q Developer Pro サブスクリプションのトラブルシューティング

Amazon Q Developer Pro サブスクリプションに問題がある場合は、以下の問題を確認して解決方法を理解してください。

トピック

- ユーザーをサブスクライブできない
- アクティベーション E メールを受信しないユーザー
- ウェブサイトで AWS サブスクリプションを使用できないユーザー
- IDE でサブスクリプションを使用できないユーザー
- サブスクライブしているユーザーが表示されない

ユーザーをサブスクライブできない

問題: Amazon Q Developer Pro にユーザーをサブスクライブできない

解決方法

- ユーザーをサブスクライブするために必要な最小限のアクセス許可があることを確認します。詳細については、「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。必要なアクセス許可を取得したら、コンソールページを再び読み込んで Amazon Q にアクセスします。
- Amazon Q Developer コンソール (Amazon Q コンソールではない) を使用していることを確認します。

- Amazon Q Developer AWS リージョン でサポートされている にいることを確認します。詳細については、「Q Developer コンソールと Q Developer プロファイルでサポートされているリージョン」を参照してください。
- 使用しているアカウントのタイプに適したワークフローに従っていることを確認します。詳細については、個人アカウントの開始方法 (ビルダー ID)「」、スタンドアロンアカウントで Amazon Q Developer Pro にユーザーをサブスクライブする「」、管理アカウントの Amazon Q Developer Pro にユーザーをサブスクライブする「」、または「」を参照してくださいメンバーアカウントの Amazon Q Developer Pro にユーザーをサブスクライブする。

アクティベーション E メールを受信しないユーザー

問題: ユーザーがアクティベーション E メールを受信していない

解決方法

- E メールアドレスが正しいことを確認します AWS IAM Identity Center。
- ユーザーに、Amazon Q Developer Pro サブスクリプションのアクティブ化というタイトルの E メールのスパムフォルダまたは迷惑メールフォルダを確認してもらいます。
- アクティベーション E メールの配信には最大 24 時間かかります。
- ユーザーが適切に IAM Identity Center に追加されていることを確認します。詳細については、「AWS IAM Identity Center ユーザーガイド」の「IAM Identity Center ディレクトリにユーザーを追加する」を参照してください。

ウェブサイトで AWS サブスクリプションを使用できないユーザー

問題: ユーザーがウェブサイトで AWS サブスクリプションを使用できない

AWS Management Consoleおよび AWS アプリやウェブサイトで Amazon Q を使用しようとすると、ブラウザに次のメッセージが表示されます。

```
Your account has not been configured to use an Amazon Q subscription.
You currently have access to the Free tier of Amazon Q. Contact your AWS
administrator to configure your subscription.
```

解決方法

- ID 拡張コンソールセッションが有効になっていることを確認します (IAM Identity Center の組織インスタンスでのみ使用可能) 。ID 拡張コンソールセッションを有効にする方法については、「ID 拡張コンソールセッションの有効化」を参照してください。
- ユーザーにアクティブな Amazon Q Developer Pro サブスクリプションがあることを確認します。詳細については、「Amazon Q Developer サブスクリプションのステータス」を参照してください。
- IAM Identity Center のアカウントインスタンスを使用していないことを確認します。アカウントインスタンスはコンソールアクセスをサポートしていません。詳細については、「 AWS IAM Identity Center ユーザーガイド」の「アカウントインスタンスに関する考慮事項」を参照してください。

  IAM Identity Center のアカウントインスタンスに ID を持つユーザーは、コンソールで Amazon Q を引き続き使用できますが、無料利用枠に制限されます。
- ユーザーが最近無料利用枠から Pro 利用枠に切り替えた場合は、 AWS Management Console または他の AWS ウェブサイトからサインアウトして再度サインインしてください。
- グループの一部としてユーザーをサブスクライブした場合は、サブスクリプションが有効になるまで最大 24 時間かかります。ユーザーがグループに追加されてからサブスクリプションがアクティブになるまでに遅延が生じる場合があります。
- Amazon Q Developer Pro マネージドアプリケーションへのユーザーのアクセスが取り消されていないか、マネージドアプリケーションが削除されていないことを確認します。必要に応じて、マネージドアプリケーションへのアクセスを復元します。
- ユーザーがアクティブなサブスクリプションを持っていない場合は、無料利用枠を使用できるようにページを更新してください。

IDE でサブスクリプションを使用できないユーザー

問題: IAM Identity Center ワークフォースユーザーが IDE で Pro サブスクリプションを使用できない

解決方法

- ユーザーにアクティブな Amazon Q Developer Pro サブスクリプションがあることを確認します。詳細については、「Amazon Q Developer サブスクリプションのステータス」を参照してください。
- ユーザーが最近無料利用枠から Pro 利用枠に切り替えた場合は、IDE で Amazon Q からサインアウトし、再度サインインします。

- グループの一部としてユーザーをサブスクライブした場合は、サブスクリプションが有効になるまで最大 24 時間かかります。ユーザーがグループに追加されてからサブスクリプションがアクティブになるまでに遅延が生じる場合があります。
- Amazon Q Developer Pro マネージドアプリケーションへのユーザーのアクセスが取り消されていないか、マネージドアプリケーションが削除されていないことを確認します。必要に応じて、マネージドアプリケーションへのアクセスを復元します。
- サブスクリプションがアクティブになるのを待っている間に、ユーザーにビルダー ID でサインインして無料利用枠を使用します。詳細については、「IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする」を参照してください。

サブスクライブしているユーザーが表示されない

問題: サブスクライブしているユーザーが Amazon Q Developer コンソールに表示されません。

Pro 階層に 1 人以上のユーザーをサブスクライブしていますが、Amazon Q Developer コンソールのサブスクリプションページに移動すると、それらは表示されません。

解決方法

- 正しい AWS アカウント と にサインインしていることを確認します AWS リージョン。
- Amazon Q コンソールに切り替えてみてください。Amazon Q コンソールは、グループの一部としてサブスクライブされたユーザーを表示でき、 が管理する組織内の複数のアカウントにわたるサブスクリプションを表示することもできます AWS Organizations。
- Amazon Q コンソールに切り替えてもユーザーが表示されない場合は、次の操作を行います。
  - が正しいことを確認してください AWS リージョン。IAM Identity Center インスタンスがデプロイされているリージョンにいる必要があります。これは、Amazon Q Developer コンソールとプロファイルとは異なるリージョンである可能性があります。
  - を使用している場合は AWS Organizations、管理アカウントとメンバーアカウントの両方でサブスクリプションを表示できるように、信頼されたアクセスを有効にしてみてください。詳細については、「Amazon Q Developer サブスクリプションの集約リストの表示」を参照してください。

Amazon Q Developer サブスクリプションの集約リストの表示

> **Note**
>
> このセクションは、個人アカウント (ビルダー IDsには適用されません。

が管理する組織内の管理アカウント管理者はAWS Organizations、管理アカウントへのサインイン中に、Amazon Q コンソール (Amazon Q Developer コンソールではない) のサブスクリプションページに、管理アカウントとメンバーアカウントの両方からの Amazon Q Developer Pro サブスクリプションを 1 つの統合リストに表示するように Amazon Q を設定できます。この組織全体の可視性により、複数のアカウントにサインインしてサブスクリプションを追跡する必要がなくなります。

> **Note**
>
> 統一されたサブスクリプション情報は、組織全体の可視性を有効にすると、Amazon Q Developer コンソールのダッシュボードページにも表示されます。

メンバーアカウント管理者は、管理するメンバーアカウント内のサブスクリプションのみを表示できます。これは、管理アカウントで組織全体の可視性が有効になっているかどうかに関係なく当てはまります。

Amazon Q Developer サブスクリプションを組織全体で可視化できるようにするには、組織内の Amazon Q への信頼されたアクセスを有効にする必要があります。信頼されたアクセスは、Amazon Q を組織の構造のクエリを許可する信頼されたサービスとして指定できる AWS Organizations 機能です。このクエリは、サブスクリプションのステータスを表示するために必要です。

信頼されたアクセスの詳細については、「AWS Organizations ユーザーガイド」の「AWS アカウント管理用の信頼されたアクセスの有効化」を参照してください。

メンバーアカウントと管理アカウントの詳細については、AWS Organizations 「 ユーザーガイド」の「 の用語と概念 AWS Organizations」を参照してください。

以下の手順に従って、組織内の Amazon Q への信頼されたアクセスを有効にします。

前提条件

開始する前に、以下を確認してください。

- お客様は AWS 管理アカウントの管理者です。
- 管理アカウントに IAM Identity Center の組織インスタンスを設定し、Amazon Q Developer に接続している。
- IAM Identity Center の組織インスタンスには、メンバーアカウントで Amazon Q Developer Pro にサブスクライブしているユーザーが含まれています。
- Amazon Q または Amazon Q Developer コンソールでアクションを実行するために必要な最小限のアクセス許可があります (いずれかのコンソールを使用して信頼されたアクセスを有効にすることができます)。詳細については、「管理者に Amazon Q コンソールの使用を許可する」および「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。

信頼されたアクセスを有効にするには (サブスクリプションの組織全体の可視性を有効にするには)

1. AWS 管理アカウント AWS Management Console を使用して にサインインします。
2. 使用するコンソールに応じて、次のいずれかを実行します。
   - Amazon Q コンソールに切り替えます。

     [サブスクリプション] を選択します。

     ページの下部にあるサブスクリプションビュー設定セクションで、編集を選択します。

     [オン] をクリックします。
   - Amazon Q Developer コンソールに切り替えます。

     [設定] を選択します。

     サブスクリプションビュー設定セクションで、編集を選択します。

     トグルを有効にします。
3. [保存] を選択します。

   Amazon Q への信頼されたアクセスが有効になりました。メンバーアカウントにサブスクライブしているユーザーとグループは、管理アカウント管理者としてサインインすると、Amazon Q コンソール (Amazon Q Developer コンソールではない) に表示されるようになりました。

Amazon Q Developer Pro からのサブスクリプション解除

Pro 階層のサブスクリプションを解除する方法は、個人アカウント (ビルダー ID) を持つエンドユーザーであるか、IAM Identity Center ワークフォースユーザーの管理者であるかによって異なります。

個人アカウントのサブスクリプション解除 (ビルダー ID)

Pro 階層から個人アカウント (ビルダー ID) のサブスクリプションを解除する場合は、このセクションをお読みください。

Builder ID のサブスクリプションを解除すると、サブスクリプションはキャンセル済みとマークされ、Amazon Q Developer の機能にアクセスできなくなります。(ただし、無料利用枠の制限を超えない限り、 無料利用枠を使用できます。) 最終的な月額サブスクリプション料金は、現在の請求サイクルの終了時に請求されます。1 か月分の料金が請求されます。料金は按分されません。

> **Warning**
>
> ビルダー ID を削除しても、サブスクリプションは解除されません。請求を停止するには、このセクションの手順を使用して、積極的にサブスクリプションを解除する必要があります。

次のいずれかのインターフェイスからサブスクリプションを解除できます。

- IDE
- コマンドライン
- AWS Management Console

IDE

IDE からビルダー ID のサブスクリプションを解除するには

1. 個人アカウント (ビルダー ID) を使用して IDE で Amazon Q に認証します。詳細については、「IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする」を参照してください。
2. Amazon Q メニューから、Q Developer Pro サブスクリプションの管理を選択します。

   ブラウザウィンドウが開きます。

3. プロンプトが表示されたら、個人アカウント (ビルダー ID) にリンク AWS アカウント されている AWS Management Console を使用して にサインインします。これは、ビルダー ID を Pro 階層にアップグレードしたときに指定 AWS アカウント しました。詳細については、「個人アカウントのアップグレード (ビルダー ID)」を参照してください。

   Amazon Q Developer コンソールのサブスクリプションページが表示されます。

4. ビルダー ID ユーザー セクションで、サブスクリプション解除を選択します。

Command line

コマンドラインから Builder ID のサブスクリプションを解除するには

1. コマンドライン用の Amazon Q がインストールされているコンピュータで、ターミナルを開きます。
2. コマンドラインで と入力して、個人アカウント (ビルダー ID) `q whoami`でサインインしていることを確認します。

   「`Logged in with Builder ID`」メッセージが表示されます。
3. ターミナルのプロンプト`q chat`で と入力してチャットセッションを開始します。

   インタラクティブチャットセッションが開きます。
4. タイプ **`/subscribe --manage`**。

   はブラウザウィンドウで AWS Management Console 起動します。

   > **Note**
   >
   > が自動的に起動 AWS Management Console しない場合は、ターミナルからブラウザウィンドウに URL をコピーして貼り付けます。

5. プロンプトが表示されたら、個人アカウント (ビルダー ID) にリンク AWS アカウント されている AWS Management Console を使用して にサインインします。これは、ビルダー ID を Pro 階層にアップグレードしたときに指定 AWS アカウント しました。詳細については、「個人アカウントのアップグレード (ビルダー ID)」を参照してください。

   Amazon Q Developer コンソールのサブスクリプションページが表示されます。

6. ビルダー ID ユーザー セクションで、サブスクリプション解除を選択します。

AWS Management Console

から Builder ID のサブスクリプションを解除するには AWS Management Console

1. 個人アカウント (ビルダー ID) にリンク AWS アカウント されている AWS Management Console を使用して にサインインします。これは、ビルダー ID を Pro 階層にアップグレード AWS アカウント したときに指定しました。詳細については、「個人アカウントのアップグレード (ビルダー ID)」を参照してください。
2. Amazon Q Developer コンソールに切り替えます。
3. [サブスクリプション] を選択します。
4. ビルダー ID ユーザー セクションで、サブスクリプション解除を選択します。

IAM Identity Center ワークフォースユーザーのサブスクリプション解除

AWS スタンドアロンアカウント、 AWS 管理アカウント、または AWS メンバーアカウントの管理者である場合は、次の手順を使用してアカウントから IAM Identity Center ワークフォースユーザーのサブスクリプションを解除します。

AWS 管理アカウントとメンバーアカウントの詳細については、「 を使用して組織内の AWS アカウントを管理する AWS Organizations」を参照してください。

ユーザーのサブスクリプション解除に関する注意事項：

- が管理する組織内の管理アカウントまたはメンバーアカウントの管理者である場合はAWS Organizations、 がサブスクリプションを作成した場合にのみ、ユーザーのサブスクリプションを解除できます。
- ユーザーがメンバーアカウントと管理アカウントの両方にサブスクライブしている場合、両方のアカウント管理者は、ユーザーのサブスクライブを完全に解除するために、それぞれのアカウントからユーザーのサブスクライブを解除する必要があります。
- 管理アカウント管理者は、Amazon Q Developer コンソールの設定ページでメンバーアカウントからサブスクリプションを表示を選択して、ユーザーがサブスクライブしている他のアカウントを表示できます。これにより、メンバーアカウント管理者と連携してサブスクリプションを解除できます。または、適切なアクセス許可がある場合は、メンバーアカウント管理者としてサインインし、ユーザーを直接サブスクリプション解除することもできます。メンバーアカウントのサブスクリプションを管理アカウント管理者として表示する方法の詳細については、「」を参照してくださいAmazon Q Developer サブスクリプションの集約リストの表示。

- ユーザーまたはグループのサブスクリプションを解除すると、サブスクリプションはキャンセル済みとマークされ、Amazon Q Developer の機能にアクセスできなくなります。(ただし、無料利用枠の制限を超えない限り、無料利用枠を使用できます)。最終的な月額サブスクリプション料金は、アクティブなサブスクリプションを持つすべてのユーザーの現在の請求サイクルの終了時に課金されます。1 か月分の料金が請求されます。料金は按分されません。

管理するユーザーまたはグループのサブスクリプションを解除するには

1. AWS スタンドアロン、管理、またはメンバーアカウント AWS Management Console を使用して にサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. ID プロバイダーのユーザーとグループセクションで、ユーザーまたはグループタブを選択します。
4. サブスクリプションを解除するユーザーまたはグループを選択します。
5. その他のアクションを選択します。
6. [サブスクリプション解除] を選択します。

Amazon Q Developer Pro へのアップグレード

無料利用枠から Pro 階層にアップグレードする方法は、個人アカウント (ビルダー ID) を持つエンドユーザーであるか、IAM Identity Center ワークフォースユーザーの管理者であるかによって異なります。

個人アカウントのアップグレード (ビルダー ID)

個人アカウント (ビルダー ID) を無料利用枠から Pro 利用枠の月額サブスクリプションにアップグレードする場合は、このセクションをお読みください。アップグレードを検討する理由については、サービスの階層「」および「」を参照してくださいBuilder IDs の制限。

次のいずれかのインターフェイスからアップグレードできます。

- IDE
- コマンドライン

[開始する前に]

- まだ作成 AWS アカウント していない場合は、 を作成します。このアカウントは Builder ID にリンクされ、月額サブスクリプション料金が請求されます。アカウントの作成の詳細については、AWS 「 アカウント管理リファレンスガイド」の「 AWS アカウントの作成」を参照してください。

> **Note**
>
> 複数の Builder IDs 1 つの AWS アカウントにリンクすることはできません。アップグレードする複数の Builder IDs がある場合は、それぞれに個別の AWS アカウントを作成します。

IDE

IDE からアップグレードするには

1. 個人アカウント (ビルダー ID) を使用して IDE で Amazon Q に認証します。詳細については、「IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする」を参照してください。
2. IDE で、次のいずれかを実行します。

   - Amazon Q チャットウィンドウに`Monthly request limit reached`メッセージが表示された場合は、「Q Developer Pro にサブスクライブする」を選択します。

     Or

   - Amazon Q メニューから、Q Developer Pro にサブスクライブを選択します。

     ブラウザウィンドウが開きます。
3. プロンプトが表示されたら、 を使用してサインインします AWS アカウント。「開始する前に」の説明に従って、このアカウントを事前に作成しておく必要があります。

   Q Developer Pro ページのアップグレードの詳細を確認するが表示されます。このページには、ビルダー ID、 AWS アカウント 番号、サブスクリプション料金などが表示されます。サブスクリプション料金の詳細については、「」を参照してくださいサブスクリプションの請求。アクティベーショントークンフィールドには、 AWS アカウントを Builder ID にリンクする 1 回限りのトークンが表示されます。このトークンは二度と使用されません。

4. アップグレードの確認 を選択します。

   Amazon Q Developer コンソールのサブスクリプションページの上部に、Q Developer Pro へのアップグレードの成功メッセージが表示されます。Builder ID users セクションにユーザー名が表示されます。

   これで、個人アカウント (ビルダー ID) が Pro 階層にサブスクライブされました。
5. (オプション) IDE の Amazon Q チャットウィンドウに戻ります。

   Amazon Q Developer Pro に正常にサブスクライブされたメッセージが表示されます。

Command line

コマンドラインからアップグレードするには

1. コマンドライン用の Amazon Q がインストールされているコンピュータで、ターミナルを開きます。
2. コマンドラインで と入力して、個人アカウント (ビルダー ID) `q whoami`でサインインしていることを確認します。

   「`Logged in with Builder ID`」メッセージが表示されます。
3. ターミナルのプロンプト`q chat`で と入力してチャットセッションを開始します。

   インタラクティブチャットセッションが開きます。
4. タイプ **`/subscribe`**。

   はブラウザウィンドウで AWS Management Console 起動します。

   > **Note**
   >
   > が自動的に起動 AWS Management Console しない場合は、ターミナルからブラウザウィンドウに URL をコピーして貼り付けます。

5. プロンプトが表示されたら、 を使用してサインインします AWS アカウント。「開始する前に」の説明に従って、このアカウントを事前に作成しておく必要があります。

   Q Developer Pro ページのアップグレードの詳細を確認するが表示されます。このページには、ビルダー ID、 AWS アカウント 番号、サブスクリプション料金などが表示されます。サブスクリプション料金の詳細については、「」を参照してくださいサブスクリプションの請

求。アクティベーショントークンフィールドには、 AWS アカウントを Builder ID にリンクする 1 回限りのトークンが表示されます。このトークンは二度と使用されません。

6. アップグレードの確認 を選択します。

   Amazon Q Developer コンソールのサブスクリプションページの上部に、Q Developer Pro へのアップグレードの成功メッセージが表示されます。Builder ID users セクションにユーザー名が表示されます。

   これで、個人アカウント (ビルダー ID) が Pro 階層にサブスクライブされました。

7. (オプション) Q CLI プロンプト**`/subscribe`**で と入力して、サブスクライブしていることを確認します。Amazon Q は、すでにサブスクライブしていることを示す必要があります。

IAM Identity Center ワークフォースユーザーを Pro 階層にアップグレードする

IAM Identity Center ワークフォースユーザーの管理者は、「」の手順に従って、これらのユーザーを Pro 階層にアップグレードできますIAM Identity Center の開始方法。

AWS アプリケーションとウェブサイトでの Amazon Q Developer の使用

、 AWS Management Console、 AWS マーケティングウェブサイト AWS Console Mobile Application、 AWS Documentation ウェブサイト、サポートされているチャットアプリケーションで Amazon Q Developer を使用して質問します AWS。 AWS アーキテクチャ、ベストプラクティス、サポート、ドキュメントについては、Amazon Q にお問い合わせください。Amazon Q は、 AWS SDKs と AWS Command Line Interface () を使用して記述するコードにも役立ちますAWS CLI。

では AWS Management Console、 AWS リソースとコストについて Amazon Q に質問したり、 に直接問い合わせ サポート たり、一般的なコンソールエラーを診断したりできます。

で Amazon Q Developer の機能にすばやくアクセスできるようにするには AWS、Amazon Q を使用して `AmazonQDeveloperAccess` AWS マネージドポリシーを IAM ID にアタッチします。特定の機能に必要なアクセス許可については、使用する機能のトピックを参照してください。

トピック

- Amazon Q Developer Pro サブスクリプションへの認証
- に関する Amazon Q Developer とのチャット AWS
- Amazon Q Developer プラグインの使用
- Amazon Q Developer Console-to-Code による AWS サービスの自動化
- Amazon Q Developer でコンソールでの一般的なエラーを診断する
- Amazon Q Developer を使用して とチャットする サポート

Amazon Q Developer Pro サブスクリプションへの認証

無料利用枠で Amazon Q にアクセスするには、 AWS Management Consoleにサインインします。必要なアクセス許可がある限り、無料利用枠機能を使用できます。

Pro 階層で Amazon Q にアクセスするには、IAM Identity Center を使用してコンソールにサインインします。IAM Identity Center に接続されている外部 ID プロバイダーを介した認証など、IAM Identity Center でサインインすると、IAM Identity Center アイデンティティが Amazon Q Developer Pro にサブスクライブされている場合、Pro 階層に自動的にアクセスできます。

Amazon Q Developer Pro 階層の詳細については、「Q Developer のサービス階層 – 無料および Pro」を参照してください。

> **Note**
>
> で始まるエラーメッセージが表示された場合は、トラブルシューティングAmazon Q Developer Pro サブスクリプションのトラブルシューティングのヒントについてはYour account has not been configured to use an Amazon Q subscription、「」を参照してください。

IAM または IAM とのフェデレーションを使用して AWS コンソールにサインインすると、無料利用枠の制限に達するか、Pro 階層でのみ利用可能な機能を使用しようとすると、IAM Identity Center で認証するように求められます。

に関する Amazon Q Developer とのチャット AWS

、 AWS Management Console、 AWS ウェブサイト AWS Console Mobile Application、 AWS Documentation ウェブサイト、チャットアプリケーションで Amazon Q とチャットして、 AWS サービスについて説明します。

ベストプラクティス、推奨事項、 AWS タスクのstep-by-stepの手順、 AWS リソースとワークフローの設計については、Amazon Q にお問い合わせください。 AWS リソースとアカウントのコストについて質問することもできます。Amazon Q は、 AWS SDKsと の使用を開始するのに役立つ短いスクリプトまたはコードスニペットも生成します AWS CLI。

以下のトピックでは、Amazon Q チャットの使用方法とチャットできるトピックについて説明します。

トピック

- アクセス許可を追加する
- 会話を開始する
- コンソールで会話を管理する
- チャット設定の設定
- 質問例
- Amazon Q Developer とのリソースに関するチャット
- リソースのトラブルシューティングを Amazon Q に依頼する
- コストに関するチャット

- ネットワークセキュリティに関するチャット
- テレメトリとオペレーションに関するチャット

アクセス許可を追加する

Amazon Q とのチャットに必要なアクセス許可を付与する IAM ポリシーについては、「ユーザーに Amazon Q とのチャットを許可する」を参照してください。

会話を開始する

で Amazon Q チャットパネルを開くには AWS Management Console、右側のサイドバーで Amazon Q アイコンを選択します。 AWS ウェブサイトまたは AWS サービスのドキュメントページでパネルを開くには、右下隅にある Amazon Q アイコンを選択します。

Amazon Q に質問するには、Amazon Q パネルのテキストバーに質問を入力します。質問に対して、参考文献へリンクする出典セクションを含む回答が生成されます。

回答を受け取ったら、オプションで、「高評価」と「低評価」のアイコンを使ってフィードバックを残すことができます。[コピー] アイコンを選択して、回答をクリップボードにコピーすることもできます。

コンソールで新しい会話を開始するには：

1. ウェルカム画面から、チャットパネルの下部にある新しい会話を選択します。
2. アクティブな会話から、チャットパネルの右上隅にあるプラスアイコンを選択して、新しい会話を開始できます。ウェルカム画面から、チャットパネルの下部にある新しい会話を選択します。
3. 会話の名前または名前を変更するには、チャットパネルの上部にあるテキストを選択し、会話名を入力します。

コンソールで会話を管理する

Amazon Q は、特定の会話内で以前に質問された質問とレスポンスの履歴を保持し、レスポンスを通知するコンテキストとして使用します。 AWS コンソールで Amazon Q チャットとの個別の会話を最大 1,000 件保存できます。

会話を開始すると、自動的に新しい会話として保存されます。会話にタイトルを付けることができます。そうしないと、Amazon Q は会話の最初のいくつかの質問に基づいてタイトルを生成します。

会話を切り替えると、質問した以前のトピックについて Amazon Q とのチャットを続けることができます。新しい質問をしない非アクティブな会話は、90 日後に削除されます。会話がまだアクティブであっても、90 日以上経過したメッセージは削除されます。

会話を切り替えるには：

1. チャットパネルの左上にあるメニューアイコンを選択します。チャットパネルで会話ページが開きます。
2. 再開する会話の名前を選択します。Amazon Q とのチャットを続行できるチャットパネルにリダイレクトされます。

会話を削除するには：

1. チャットパネルの左上にあるメニューアイコンを選択します。チャットパネルで会話ページが開きます。
2. 削除する会話の名前の横にある削除アイコンを選択します。表示されるポップアップで削除を選択して、会話を削除することを確認します。

コンソールで Amazon Q を使用している場合、コンソールの別の場所、または別のブラウザやタブに移動すると、現在の会話と関連するコンテキストが維持されます。 AWS のウェブサイト、ドキュメントウェブサイト、またはコンソールモバイルアプリケーションで Amazon Q を使用している場合、新しいページ、ブラウザ、またはタブに移動すると、コンテキストなしで新しい会話が開始されます。

チャット設定の設定

Amazon Q でチャット設定を更新するには、チャットパネルの右上隅にある歯車アイコンをクリックします。

以下の設定を指定することができます。

- リージョン — チャットパネルを開く AWS Management Console と、Amazon Q はデフォルトでの AWS リージョン セットになります。Amazon Q で使用されるリージョンを更新するには、コンソールリージョンを変更します。

質問例

適切なサービスの検索やベストプラクティスの理解など AWS のサービス、 AWS と について Amazon Q に質問できます。

AWS SDKs と を使用したソフトウェア開発について質問することもできます AWS CLI。コンソールの Amazon Q は、 AWS SDKs と の使用を開始するのに役立つ短いスクリプトまたはコードスニペットを生成できます AWS CLI。

以下は、Amazon Q が AWSでのビルドにどのように役立つかを示す質問例です。

- Lambda 関数の最大ランタイムはどのくらいですか？
- VPC にリソースをいつ配置すべきですか?
- コストを低く抑える必要がある場合、ワークロードの実行に使用するのに最適なコンテナサービスは何ですか？
- Amazon S3 バケットをリストするにはどうすればよいですか?
- でウェブサイトを作成してホストするにはどうすればよいですか AWS？

Amazon Q Developer とのリソースに関するチャット

Amazon Q Developer は、自然言語プロンプトを通じて AWS インフラストラクチャを理解するのに役立つ AWS アカウントリソースに関する質問に回答します。高度な推論機能を使用して、Amazon Q はリソースを分析してインサイトを提供するため、複数のサービスコンソール、APIs、複雑なスクリプトに頼ることなく、必要な情報をすばやく取得できます。

Amazon Q が実行できるリソース分析のタイプは次のとおりです。

- リソースのリストと詳細 – アカウント内のリソースのリストまたは特定の詳細を尋ねます。
- フィルタリングされたクエリ – リージョンや設定状態などの基準に基づいてリソース情報をリクエストします。
- クロスサービス分析 – 複数の AWS リソースとサービスにわたるインフラストラクチャ、設定、依存関係について複雑な質問をします。
- サポートのトラブルシューティング – リソースの問題の特定と解決に役立ちます。詳細については、「リソースのトラブルシューティングを Amazon Q に依頼する」を参照してください。

質問の例については、「」を参照してくださいAmazon Q にリソース情報を依頼する。

トピック

- 仕組み
- 前提条件
- Amazon Q にリソース情報を依頼する
- でリソースをカウントする AWS Resource Explorer

仕組み

リソースに関する質問に回答するために、Amazon Q はサービス APIsと AWS クラウドコントロール API を使用してリクエストされた情報を取得します。Amazon Q がリクエストされたリソース情報を取得するために必要な APIs を呼び出せるようにするには、IAM ID にそれらの APIsを使用するためのアクセス許可が必要です。詳細については、「前提条件」を参照してください。

Amazon Q は、一度に複数の AWS リソースに関する情報を取得するためのアクションを取得、一覧表示、および記述できます。複雑なリソースの質問をすると、Amazon Q は、 AWS 環境の理解を深めるために実行しているアクションの背後にある推論を説明する動的な複数ステップの計画を作成します。初期計画が失敗した場合、Amazon Q は代替方法を試みるか、続行するために必要な追加情報を求めます。

Amazon Q は、Amazon S3 バケット内のオブジェクトの一覧表示や、アカウントのセキュリティ、アイデンティティ、認証情報、暗号化に関連する質問など、リソースに保存されているデータに関する質問には回答できません。

前提条件

アカウントリソースに関するチャットは AWS Management Console AWS Console Mobile Application、、、および設定されたチャットアプリケーションで Amazon Q と行うことができます。

リソースについてチャットするには、IAM ID に次のアクセス許可が必要です。

- Amazon Q とチャットし、Cloud Control API を使用し、Amazon Q がリソースにアクセスできるようにするアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「ユーザーに Amazon Q とのリソースに関するチャットを許可する」を参照してください。
- リクエストの対象となるリソースにアクセスするためのアクセス許可。例えば、Amazon Q に Amazon S3 バケットを一覧表示するようにリクエストする場合は、`s3:ListAllMyBuckets` のアクセス許可が必要です。

Amazon Q は、利用するユーザーの IAM ID でアクセスできないリソースにはアクセスしません。

> **Important**
>
> Amazon Q に読み取り、一覧表示、または記述のアクションの実行をリクエストする場合、通常の料金が適用されます。詳細については、Amazon Q に尋ねる AWS サービスの料金ページを参照してください。

Amazon Q にリソース情報を依頼する

リソースについて Amazon Q に尋ねるときは、Amazon Q AWS リージョン がリソースを見つけるために呼び出す を指定できます。特定のクエリでリージョンが指定されていない場合、Amazon Q は必要に応じて会話で以前に指定したリージョンを使用し、それ以外の場合は現在のコンソールリージョン (グローバルコンソールリージョンを使用している場合は最新のコンソールリージョン) を使用します。

Amazon Q では、リソースの質問に回答するために追加情報が必要になる場合があります。Amazon Q にフォローアップを求められたら、リクエストされた詳細を返します。

リソースについて Amazon Q に尋ねることができる質問の例を次に示します。

- S3 バケット *<##>* の暗号化設定を記述して
- Lambda 関数を呼び出す SQS キュー
- 更新が必要な MySQL RDS クラスターはありますか？
- *<region>* で EC2 インスタンスを一覧表示する
- Lambda 関数 *<##>* の設定を取得して
- インスタンス *<###### ID>* にはどのようなアラームが設定されていますか?

でリソースをカウントする AWS Resource Explorer

「アカウントで実行されている EC2 リソースの数」など、リソースカウントが必要な質問をすると、Amazon Q はデフォルトで Cloud Control API を使用して、リクエストされたリソースの数を返します。Amazon Q でリソースのカウントを高速化するために、Resource Explorer を有効にして設定することもできます。

Resource Explorer が有効になっている場合、Amazon Q はリソースをカウントする必要があるレスポンスを生成するときにそのリソースの使用を試みます。Amazon Q は Resource Explorer を使用して、すべての で単一のタイプのリソースをカウントできます AWS リージョン。Resource Explorer を使用すると、サービス APIsを一覧表示して結果をカウントするのではなく、Resource Explorer インデックスからカウントを返すことで、Amazon Q がリソースをより迅速にカウントできるようになります。

Resource Explorer でリソースカウントを有効にする場合は、リソース情報が古くなる可能性があることに注意してください。Resource Explorer は、定期的なインベントリを取得してアカウント内のリソースをインデックス化し、最後のインベントリ後にリソースが作成または削除された場合、リソース数は正しくありません。Resource Explorer はリソースフィルタリングもサポートしていません。特定の条件に一致するリソースのカウントを に依頼すると、Amazon Q は Cloud Control API にフォールバックします。

Resource Explorer を有効にして使用できるように設定していない場合、または Amazon Q が Resource Explorer を使用して質問に回答できない場合、Amazon Q は Cloud Control API を使用してリソースをカウントします。Cloud Control API を使用すると、正確なリソース数が確保され、リソースフィルタリングがサポートされますが、Resource Explorer でのカウントと比較してレイテンシーが増加する可能性もあります。多数のリソースをカウントする場合、Cloud Control API もタイムアウトする可能性があります。

Resource Explorer をリソースカウントに使用するには、次の設定が必要です。

- Amazon Q を操作するユーザーは、Resource Explorer のデフォルトビューが設定され、アグリゲーターインデックスがデフォルトビューと同じリージョンに作成されたアカウントに存在する必要があります。詳細については、「 AWS Resource Explorer ユーザーガイド」の「高度なセットアップを使用した Resource Explorer のセットアップ」を参照してください。
- ユーザーの IAM ID には、デフォルトビューの読み取りアクセス許可が必要です。詳細については、「 AWS Resource Explorer ユーザーガイド」の「検索のために Resource Explorer ビューへのアクセスを許可する」を参照してください。

リソースのトラブルシューティングを Amazon Q に依頼する

では AWS Management Console、 AWS リソースで発生している問題のトラブルシューティングを Amazon Q に依頼できます。問題が発生した場合は、チャットパネルを開き、Amazon Q に状況を記述します。例えば、S3 バケットにオブジェクトを追加できません」または「ロードバランサーが 503 エラーを返しています」と入力します。Amazon Q は、提供された情報を分析して潜在的な根本

原因を特定します。次に、カスタマイズされたソリューション、step-by-stepの手順、またはベストプラクティスを提供して、問題を効率的に解決します。

Amazon Q は現在、次の表に示す問題の英語プロンプトを受け入れています。

AWS サービス	Amazon Q が役立つ問題のタイプ	プロンプトの例
Amazon S3	アクセス許可の問題	S3 バケットにオブジェクトを配置できないのはなぜですか？ バケット ID は amzn-s3-demo-bucket です。 オブジェクト s3://amzn-s3-demo-bucket-locked/Q-Stream2.jpg を削除できないのはなぜですか？ S3 でオブジェクトを削除できないのはなぜですか？
AWS Glue	ジョブの失敗	ジョブ名 'Run111B11B11-<...>' の Glue ジョブと、'us-west-2' リージョンのジョブ実行 ID 'bb_b1b111<...>' が失敗しました。 GlueRun00AA00A00A-<...> という Glue ジョブが失敗したのはなぜですか？
Amazon Athena	クエリの問題	Athena クエリが結果を返さなかった。クエリ ID: 222c22cc-2c022-<...> region id: us-east-2 実行 ID が 333d33dd-3d33-<...>、リージョンが us-east-1 の Athena クエリを

AWS サービス	Amazon Q が役立つ問題のタイプ	プロンプトの例
		実行しましたが、結果は返されませんでした。
Amazon ECS	タスク停止の問題、Fargate ヘルスチェックの問題、切断されたエージェントの問題	ECS タスクが停止し、理由がわかりません。タスクの詳細は次のとおりです。クラスター: my-ecs-cluster、サービス: my-ecs-service、タスク定義: my-task-definition、タスク ARN: arn:aws:ecs:us-west-2:444444444444:task/my-ecs-cluster/4ee4ee4ee444<...> ECS タスクに問題がある。タスクのヘルスチェックは、「my-ecs-cluster」クラスターとサービスのタスクで常に失敗します。 いずれかのコンテナインスタンスの Amazon ECS エージェントは切断されているようです。エージェントは応答せず、ステータスを更新していないため、タスクが保留中の状態のままになります。

AWS サービス	Amazon Q が役立つ問題のタイプ	プロンプトの例
Amazon EC2 Elastic Load Balancing	ヘルスチェックの問題、504、503、502、500 エラー	ターゲットグループのヘルスチェックが「my-target-group」という名前で失敗するのはなぜですか？ ロードバランサーの「my-elb」から 503 エラーが表示されるのはなぜですか？
Amazon EKS	Application Load Balancer (ALB) Ingress Controller の問題、マネージドアドオンの問題	EKS クラスターに ALB イングレスコントローラーがあり、WebIdentityErr:failed to retrieve credentials」というエラーメッセージが表示されています。 AWS リージョンは us-west-2 です。 us-west-2 リージョンの my-eks-cluster という EKS クラスターのアドオンに問題があるようです。
Amazon ECR	セカンダリアカウントアクセスの問題	Amazon ECR イメージリポジトリへのアクセスを別のリポジトリから許可するのが難しい AWS アカウント。具体的には、アカウント 222222222222 がリージョン (usmy-ecr-repo」という名前のリポジトリからイメージをプッシュおよびプルすることを許可する必要があります。 (111111111111

Amazon Q がリソースをトラブルシューティングするには、「」で説明されているのと同じアクセス許可が必要ですAmazon Q Developer とのリソースに関するチャット。

コストに関するチャット

履歴コストデータと予測 AWS コストデータ、およびコストの最適化方法について Amazon Q に質問できます。Amazon Q では、コストデータの取得、コストの説明、コスト傾向の分析を行うことができ、ドキュメントを参照したりワークフローを中断したりすることなくコストを把握できます。Amazon Q は、インスタンスの適正化、アイドルまたは未使用のリソースの停止または削除、EC2 インスタンスの AWS Gravitonベースのインスタンスへの移行、Savings Plans の購入など、 AWS コストを最適化する方法に関する推奨事項を提供することもできます。

コストについて Amazon Q に尋ねると、そのレスポンスには、データの取得に使用される特定のパラメータと、 AWS マネジメントコンソールで詳細を知るためのリンクが含まれます。

Amazon Q でのコスト分析の詳細については、「 コスト管理ユーザーガイド」の「Amazon Q Developer で生成 AI を使用したコストの分析と最適化」を参照してください。 AWS

前提条件

AWS Management Console および設定されたチャットアプリケーションで、 AWS コストについてチャットできます。 ???

Amazon Q がコストに関する質問に回答するには、次の前提条件を満たす必要があります。

アクセス許可を追加する

コストについてチャットするには、IAM ID に Amazon Q とチャットし、請求データにアクセスし、コスト最適化の推奨事項を取得するアクセス許可が必要です。必要なアクセス許可を付与する IAM ポリシーの例については、「Amazon Q がコストデータにアクセスし、コスト最適化に関する推奨事項を提供できるようにする」を参照してください。

AWS Cost Explorer と を有効にする AWS Cost Optimization Hub

Amazon Q とコストについてチャットするには、アカウント AWS Cost Explorer AWS で を有効にする必要があります。Cost Explorer を有効にするには、Cost Explorer コンソールを開きます。詳細については、「AWS コスト管理ユーザーガイド」の「Cost Explorer を有効にする」を参照してください。

Amazon Q からコスト最適化の推奨事項を受け取るには、 を有効にする必要があります AWS Cost Optimization Hub。詳細については、「AWS コスト管理ユーザーガイド」の「Cost Optimization Hub の開始方法」を参照してください。

質問例

以下は、Amazon Q に尋ねることができるコストに関する質問の例です。

- 1 月に SageMaker AI にいくらかかりましたか？
- `'eu-central-1'` リージョンの AWS 請求書に最も貢献しているサービスは何ですか？
- 先週のインスタンスタイプ別の Amazon EC2 のコストはいくらでしたか?
- 過去 3 か月間のサービス別のコスト内訳は?
- AWS 請求額を下げるにはどうすればよいですか？
- 最高のコスト最適化の機会は何ですか？
- Amazon EC2 インスタンスのいずれかが過剰にプロビジョニングされていますか？
- どの Savings Plans を購入すればよいですか？

ネットワークセキュリティに関するチャット

> ネットワークセキュリティに関するチャットはプレビュー中であり、変更される可能性があります。

Amazon Q は、ネットワークセキュリティ設定の分析、 AWS ネットワークセキュリティサービスの欠落や誤設定の特定、ネットワークセキュリティ体制の強化に関する推奨事項の提供に役立ちます。これにより、ワークフローを中断することなく、ネットワークセキュリティの検出結果を理解し、修復手順を実装し、セキュリティのベストプラクティスに従うことができます。

ネットワークセキュリティについて Amazon Q に尋ねると、そのレスポンスには、 リソースに関する特定の情報、関連するセキュリティの検出結果、詳細な修復手順、および で詳しく知るためのリンクが含まれます AWS Management Console。

Amazon Q でのネットワークセキュリティ分析の詳細については、AWS Shield 「ネットワークセキュリティディレクターデベロッパーガイド」の「Amazon Q Developer でインサイトを取得する」を参照してください。

前提条件

AWS Management Console および設定されたチャットアプリケーションで、 AWS ネットワークセキュリティについてチャットできます。 ???

Amazon Q がネットワークセキュリティに関する質問に回答するには、次の前提条件を満たす必要があります。

アクセス許可を追加する

ネットワークセキュリティについてチャットするには、IAM ID に Amazon Q とチャットするためのアクセス許可が必要です。必要なアクセス許可を付与する IAM ポリシーについては、「Amazon Q とのチャットをユーザーに許可する」を参照してください。

AWS Shield ネットワークセキュリティディレクターを有効にする

Amazon Q とネットワークセキュリティについてチャットするには、 AWS アカウントでネットワークセキュリティディレクターを有効にする AWS Shield 必要があります。ネットワークセキュリティディレクターを有効にする AWS Shield には：

1. https://console.aws.amazon.com/nsd/ で AWS Shield ネットワークセキュリティディレクターコンソールを開きます。
2. セットアップ手順に従って、サービスを有効にします。
3. スキャンを実行して、 リソースに関するセキュリティ情報を収集します。

質問例

Amazon Q に尋ねることができるネットワークセキュリティに関する質問の例を次に示します。

- ネットワークセキュリティに関する上位の検出結果を特定する
- 環境のネットワークセキュリティを要約する
- システムが DDoS 攻撃の危険にさらされていますか？
- ネットワークセキュリティを向上させるにはどうすればよいですか？
- WAF 保護のないリソースはありますか？
- 一般的なウェブ脆弱性から保護されていないリソースはどれですか？
- EC2 インスタンスの一般的なネットワークセキュリティの問題は何ですか？
- 何も保護していない WAF WebACLs はありますか？

テレメトリとオペレーションに関するチャット

Amazon Q は CloudWatch テレメトリと運用データを分析して AWS 、環境の管理に役立てます。リソースのヘルス情報を取得し、アラームをモニタリングして、トラブルシューティングのガイダンス

を提供します。質問すると、Amazon Q は正確なサポートを確保するために、リソース名や時間範囲などの特定の詳細の入力を求められることがあります。

AWS サービスヘルスチェック： 指定された AWS サービスのリソースの状態を評価し、お客様がこれらのリソースで発生した問題やエラーのトラブルシューティングと解決を支援します。

- Lambda 関数 X は正常ですか？
- Amazon ECS クラスターに何か問題がありますか？
- X 時間から Y 時間までの DynamoDB テーブルのトラブルシューティングを行います。
- 時間 X と Y の間の Amazon S3 に関連する異常を調査します。

アラームのトラブルシューティング： アラーム状態のアラームと、アラームをトリガーした基盤となるテレメトリを識別し、アラームalarm/alert/pages背後にある理由をお客様が診断するのに役立ちます。

- X という名前のアラームが発せられるのはなぜですか？

Application Signals 固有のトラブルシューティング： CloudWatch Application Signals のサービスレベルの目標とインジケータを分析して、サービスの全体的な状態を判断し、アプリケーションのパフォーマンスを評価および維持できるようにします。

- 環境 Y のサービス X は正常ですか？

Amazon Q が CloudWatch テレメトリと運用データを分析する方法の詳細については、「Amazon CloudWatch ユーザーガイド」の「CloudWatch 調査」を参照してください。 Amazon CloudWatch

Amazon Q Developer プラグインの使用

Amazon Q Developer はサードパーティーのモニタリングツールやセキュリティプラットフォームと統合されているため、 AWS ビルダー環境を離れることなく AWS アプリケーションインサイトにアクセスできます。では AWS Management Console、これらのツールが提供するメトリクスについてチャットして、アプリケーションのパフォーマンス、エラー、脆弱性を理解して対処できます。

プラグインを設定したら、 AWS コンソールで Amazon Q とチャットするときに、質問の先頭にプラグインエイリアスを追加します。Amazon Q はサードパーティープロバイダー APIsを呼び出してリソースを取得し、外部リソースへのディープリンクを含むレスポンスを生成します。

Amazon Q がサードパーティー API を呼び出すと、API は AWS CloudTrail ログに表示されません。CloudTrail ログには、Amazon Q がサードパーティープロバイダーに接続するための認証情報を取得するために AWS Secrets Manager シークレットにアクセスした場合にのみ表示されます。

Amazon Q は、プラグインを設定または使用する際に、サードパーティープロバイダーと情報を共有しません。Amazon Q がデータを使用する方法の詳細については、「」を参照してくださいデータ保護。

> **Note**
>
> AWS 組織内のメンバーアカウントは、組織の管理アカウントプロファイルで設定されたプラグインにアクセスできません。各メンバーアカウントは、アカウントでプラグインを設定して使用する前に、独自の Q Developer プロファイルを作成する必要があります。

> **Warning**
>
> サードパーティープロバイダーのユーザーアクセス許可は、Amazon Q Developer プラグインでは検出されません。管理者が AWS アカウントでプラグインを設定すると、そのアカウントのプラグイン権限を持つユーザーは、プラグインによって取得可能なサードパーティープロバイダーアカウントの任意のリソースにアクセスできます。
> IAM ポリシーを設定して、ユーザーがアクセスできるプラグインを制限できます。詳細については、「ユーザーが 1 つのプロバイダーのプラグインとチャットできるようにする」を参照してください。

開始するには、Amazon Q Developer で使用するプラグインのトピックを参照してください。

トピック

- Amazon Q Developer CloudZero プラグインの設定
- Amazon Q Developer Datadog プラグインの設定
- Amazon Q Developer Wiz プラグインの設定

Amazon Q Developer CloudZero プラグインの設定

CloudZero は、クラウド効率を向上させるためにコストを評価するクラウドコスト最適化プラットフォームです。CloudZero を使用して AWS コストをモニタリングする場合は、Amazon Q

Developer チャットの CloudZeroプラグインを使用して、 を離れることなくコストインサイトにアクセスできます AWS Management Console。

CloudZero プラグインを使用して、 AWS コストの把握、コスト最適化のインサイトの取得、請求の追跡を行うことができます。回答を受け取ったら、CloudZeroインサイトのステータスやコストへの影響など、フォローアップの質問をすることができます。

プラグインを設定するには、CloudZeroアカウントから認証情報を指定して、Amazon Q と 間の接続を有効にしますCloudZero。プラグインを設定したら、Amazon Q チャットで質問の先頭**@cloudzero**に を追加してCloudZeroデータにアクセスできます。

> **Warning**
>
> CloudZero ユーザーアクセス許可は、Amazon Q のCloudZeroプラグインによって検出されません。管理者が AWS アカウントのCloudZeroプラグインを設定すると、そのアカウントのプラグインアクセス許可を持つユーザーは、プラグインによって取得可能なCloudZeroアカウントの任意のリソースにアクセスできます。
> IAM ポリシーを設定して、ユーザーがアクセスできるプラグインを制限できます。詳細については、「ユーザーアクセス許可を設定する」を参照してください。

前提条件

アクセス許可を追加する

プラグインを設定するには、次の管理者レベルのアクセス許可が必要です。

- Amazon Q Developer コンソールへのアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者に Amazon Q Developer コンソールの使用を許可する。
- プラグインを設定するアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者にプラグインの設定を許可する。

認証情報の取得

開始する前に、CloudZeroアカウントから次の情報を書き留めます。これらの認証情報は、プラグインを設定するときに AWS Secrets Manager シークレットに保存されます。

- API キー – Amazon Q が CloudZero API を呼び出して組織のコストインサイトと請求情報にアクセスできるようにするアクセスキー。API キーはCloudZeroアカウント設定にあります。詳細については、 CloudZeroドキュメントの「認可」を参照してください。

CloudZero アカウントから認証情報を取得する方法の詳細については、 CloudZeroドキュメントを参照してください。

シークレットとサービスロール

AWS Secrets Manager シークレット

プラグインを設定すると、Amazon Q はCloudZero認証情報を保存するための新しい AWS Secrets Manager シークレットを作成します。または、自分で作成した既存のシークレットを使用することもできます。

シークレットを自分で作成する場合は、API キーをプレーンテキストとして入力します。

```
your-api-key
```

シークレットの作成の詳細については、「 AWS Secrets Manager ユーザーガイド」の「シークレットの作成」を参照してください。

サービス役割

Amazon Q Developer でCloudZeroプラグインを設定するには、Secrets Manager シークレットへのアクセス許可を Amazon Q に付与するサービスロールを作成する必要があります。Amazon Q はこのロールを引き受けて、CloudZero認証情報が保存されているシークレットにアクセスします。

AWS コンソールでプラグインを設定する場合、新しいシークレットを作成するか、既存のシークレットを使用するかを選択できます。新しいシークレットを作成すると、関連付けられたサービスロールが自動的に作成されます。既存のシークレットと既存のサービスロールを使用する場合は、サービスロールに次のアクセス許可が含まれ、次の信頼ポリシーがアタッチされていることを確認します。必要なサービスロールは、シークレットの暗号化方法によって異なります。

シークレットがマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:{{your-account-id}}:secret:
[[secret-id]]"
            ]
        }
    ]
}
```

シークレットがカスタマーマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:us-east-1:{{accountId}}:secret:
{{secretId}}"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:{{accountId}}:key/{{keyId}}",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
                }
```

```
            }
        }
    ]
}
```

Amazon Q がサービスロールを引き受けることを許可するには、サービスロールに次の信頼ポリシーが必要です。

> **Note**
>
> codewhisperer プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "q.amazonaws.com"
      },
      "Action": ["sts:AssumeRole", "sts:SetContext"],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}",
          "aws:SourceArn": "arn:aws:codewhisperer:{{region}}:
{{accountId}}:profile/{{profileId}}"
        }
      }
    }
  ]
}
```

サービスロールの詳細については、「 AWS Identity and Access Management ユーザーガイド」の「 AWS サービスにアクセス許可を委任するロールを作成する」を参照してください。

CloudZero プラグインを設定する

Amazon Q Developer コンソールでプラグインを設定します。Amazon Q は、 に保存されている認証情報 AWS Secrets Manager を使用して、 とのやり取りを有効にしますCloudZero。

CloudZero プラグインを設定するには、次の手順を実行します。

1. https://console.aws.amazon.com/amazonq/developer/home で Amazon Q Developer コンソールを開きます。
2. Amazon Q Developer コンソールのホームページで、設定を選択します。
3. ナビゲーションバーで、プラグインを選択します。
4. プラグインページで、CloudZeroパネルのプラス記号を選択します。プラグイン設定ページが開きます。
5. Configure AWS Secrets Manager で、新しいシークレットを作成するか、既存のシークレットを使用するを選択します。Secrets Manager シークレットは、CloudZero認証情報が保存される場所です。

   新しいシークレットを作成する場合は、次の情報を入力します。

   a. CloudZero API キーには、CloudZero組織の API キーを入力します。
   b. Amazon Q がCloudZero認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

   既存のシークレットを使用する場合は、シークレットドロップダウンメニューからAWS Secrets Manager シークレットを選択します。シークレットには、前のステップで指定したCloudZero認証情報が含まれている必要があります。

   必要な認証情報の詳細については、「認証情報の取得」を参照してください。
6. IAM サービスロールを設定する AWS で、新しいサービスロールを作成する または既存のサービスロールを使用する を選択します。

> **Note**
>
> ステップ 6 で新しいシークレットを作成するを選択した場合、既存のサービスロールを使用することはできません。新しいロールが作成されます。

新しいサービスロールを作成すると、Amazon Q がCloudZero認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

既存のサービスロールを使用する場合は、表示されるドロップダウンメニューからロールを選択します。サービスロールに、 で定義されているアクセス許可と信頼ポリシーがあることを確認しますサービス役割。

7. [設定の保存] を選択します。
8. CloudZero プラグインページの「設定済みプラグイン」セクションにプラグインパネルが表示されると、ユーザーはプラグインにアクセスできます。

プラグインの認証情報を更新する場合は、現在のプラグインを削除して新しいプラグインを設定する必要があります。プラグインを削除すると、以前の仕様がすべて削除されます。新しいプラグインを設定するたびに、新しいプラグイン ARN が生成されます。

ユーザーアクセス許可を設定する

プラグインを使用するには、次のアクセス許可が必要です。

- コンソールで Amazon Q とチャットするアクセス許可。チャットに必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してくださいユーザーに Amazon Q とのチャットを許可する。
- `q:UsePlugin` アクセス許可

IAM ID に設定されたCloudZeroプラグインへのアクセスを許可すると、ID はプラグインによって取得可能なCloudZeroアカウント内のリソースへのアクセスを取得します。 CloudZero ユーザーのアクセス許可はプラグインによって検出されません。プラグインへのアクセスを制御するには、IAM ポリシーでプラグイン ARN を指定します。

プラグインを作成または削除して再設定するたびに、新しい ARN が割り当てられます。ポリシーでプラグイン ARN を使用する場合は、新しく設定されたプラグインへのアクセスを許可する場合は更新する必要があります。

CloudZero プラグイン ARN を見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定されたCloudZeroプラグインを選択します。プラグインの詳細ページで、プラグイン

ARN をコピーします。この ARN をポリシーに追加して、CloudZeroプラグインへのアクセスを許可または拒否できます。

CloudZero プラグインへのアクセスを制御するポリシーを作成する場合は、ポリシーでプラグインプロバイダーCloudZeroに を指定します。

プラグインアクセスを制御する IAM ポリシーの例については、「」を参照してくださいユーザーが 1 つのプロバイダーのプラグインとチャットできるようにする。

CloudZero プラグインとチャットする

CloudZero プラグインを使用するには、 CloudZero または AWS アプリケーションモニターとケースに関する質問の冒**@cloudzero**頭に と入力します。Amazon Q からのフォローアップの質問や質問への回答には、 も含める必要があります**@cloudzero**。

以下は、Amazon Q CloudZeroプラグインを最大限に活用するためのユースケースの例と、関連する質問です。

- CloudZero で を使用する方法を学ぶ AWS – CloudZero機能の仕組みについて質問します。Amazon Q では、最善の回答を提供するために何をしようとしているかの詳細を求められる場合があります。
  - **@cloudzero how do I use CloudZero?**
  - **@cloudzero how do I get started with CloudZero?**
- コストインサイトのリスト — コストインサイトのリストを取得したり、特定のインサイトの詳細を確認したりできます。
  - **@cloudzero list my top cost insights**
  - **@cloudzero tell me more about insight <insight ID>**
- 請求情報の取得 – AWS 請求情報については、Amazon Q CloudZero プラグインにお問い合わせください。
  - **@cloudzero what were my AWS costs for December 2024?**

Amazon Q Developer Datadog プラグインの設定

Datadog は、インフラストラクチャ、アプリケーション、ネットワークのモニタリングと分析を提供するモニタリングおよびセキュリティプラットフォームです。Datadog を使用して AWS アプリケーションをモニタリングする場合は、Amazon Q Developer チャットの Datadogプラグインを使用して、 を離れることなくモニタリング情報にアクセスできます AWS Management Console。

Datadog プラグインを使用して、 について学びDatadog、 AWS サービスとの連携を理解し、Datadogケースやモニターについて質問できます。応答を受け取ったら、問題に対処する方法やDatadogリソースの詳細など、フォローアップの質問をすることができます。

プラグインを設定するには、Datadogアカウントから認証情報を指定して、Amazon Q と 間の接続を有効にしますDatadog。プラグインを設定したら、Amazon Q チャットの質問の先頭**@datadog**にを追加してDatadogメトリクスにアクセスできます。

> ⚠ Warning
>
> Datadog ユーザーアクセス許可は、Amazon Q のDatadogプラグインによって検出されません。管理者が AWS アカウントのDatadogプラグインを設定すると、そのアカウントのプラグインアクセス許可を持つユーザーは、プラグインによって取得可能なDatadogアカウントの任意のリソースにアクセスできます。
> IAM ポリシーを設定して、ユーザーがアクセスできるプラグインを制限できます。詳細については、「ユーザーアクセス許可を設定する」を参照してください。

前提条件

アクセス許可を追加する

プラグインを設定するには、次の管理者レベルのアクセス許可が必要です。

- Amazon Q Developer コンソールへのアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者に Amazon Q Developer コンソールの使用を許可する。
- プラグインを設定するアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者にプラグインの設定を許可する。

認証情報の取得

開始する前に、 Datadogアカウントの次の情報をメモしておきます。これらの認証情報は、プラグインを設定するときに AWS Secrets Manager シークレットに保存されます。

- サイトパラメータ – 使用するDatadogサイトパラメータ。例えば、`us3.datadoghq.com`。詳細については、 Datadogドキュメントの「Datadog サイトの開始方法」を参照してください。
- API キーとアプリケーションキー – Amazon Q が Datadog API を呼び出してイベントとメトリクスにアクセスできるようにするアクセスキー。これらは、Datadogアカウントの組織設定にありま

す。詳細については、 Datadogドキュメントの「API とアプリケーションキー」を参照してください。

シークレットとサービスロール

AWS Secrets Manager シークレット

プラグインを設定すると、Amazon Q はDatadog認証情報を保存するための新しい AWS Secrets Manager シークレットを作成します。または、自分で作成した既存のシークレットを使用することもできます。

シークレットを自分で作成する場合は、次の認証情報が含まれていることを確認し、次の JSON 形式を使用してください。

```
{
   "ApiKey": "<your-api-key>",
   "AppKey": "<your-applicaiton-key>"
}
```

シークレットの作成の詳細については、「 AWS Secrets Manager ユーザーガイド」の「シークレットの作成」を参照してください。

サービス役割

Amazon Q Developer でDatadogプラグインを設定するには、Secrets Manager シークレットへのアクセス許可を Amazon Q に付与するサービスロールを作成する必要があります。Amazon Q は、Datadog認証情報が保存されているシークレットにアクセスするためにこのロールを引き受けます。

AWS コンソールでプラグインを設定する場合、新しいシークレットを作成するか、既存のシークレットを使用するかを選択できます。新しいシークレットを作成すると、関連付けられたサービスロールが自動的に作成されます。既存のシークレットと既存のサービスロールを使用する場合は、サービスロールに次のアクセス許可が含まれ、次の信頼ポリシーがアタッチされていることを確認します。必要なサービスロールは、シークレットの暗号化方法によって異なります。

シークレットがマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:{{your-account-id}}:secret:
[[secret-id]]"
            ]
        }
    ]
}
```

シークレットがカスタマーマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:us-east-1:{{accountId}}:secret:
{{secretId}}"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:{{accountId}}:key/{{keyId}}",
```

```
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
                }
            }
        }
    ]
}
```

Amazon Q がサービスロールを引き受けることを許可するには、サービスロールに次の信頼ポリシーが必要です。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "q.amazonaws.com"
      },
      "Action": ["sts:AssumeRole", "sts:SetContext"],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}",
          "aws:SourceArn": "arn:aws:codewhisperer:{{region}}:
{{accountId}}:profile/{{profileId}}"
        }
      }
    }
  ]
}
```

サービスロールの詳細については、「 AWS Identity and Access Management ユーザーガイド」の「 AWS サービスにアクセス許可を委任するロールを作成する」を参照してください。

Datadog プラグインを設定する

Amazon Q Developer コンソールでプラグインを設定します。Amazon Q は、 に保存されている認証情報 AWS Secrets Manager を使用して、 とのやり取りを有効にしますDatadog。

Datadog プラグインを設定するには、次の手順を実行します。

1. https://console.aws.amazon.com/amazonq/developer/home で Amazon Q Developer コンソールを開きます。
2. Amazon Q Developer コンソールのホームページで、設定を選択します。
3. ナビゲーションバーで、プラグインを選択します。
4. プラグインページで、Datadogパネルのプラス記号を選択します。プラグイン設定ページが開きます。
5. サイト URL には、使用するDatadogサイトの URL を入力します。
6. Configure AWS Secrets Manager で、新しいシークレットを作成するか、既存のシークレットを使用するを選択します。Secrets Manager シークレットは、Datadog認証情報が保存される場所です。

   新しいシークレットを作成する場合は、次の情報を入力します。

   a. Datadog API キーには、Datadog組織の API キーを入力します。
   b. Datadog アプリケーションキーには、Datadogアカウントのアプリケーションキーを入力します。
   c. Amazon Q がDatadog認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

   既存のシークレットを使用する場合は、シークレットドロップダウンメニューからAWS Secrets Manager シークレットを選択します。シークレットには、前のステップで指定したDatadog認証情報が含まれている必要があります。

   必要な認証情報の詳細については、「認証情報の取得 」を参照してください。
7. IAM サービスロールを設定する AWS で、新しいサービスロールを作成する または既存のサービスロールを使用する を選択します。

> **Note**
>
> ステップ 6 で新しいシークレットを作成するを選択した場合、既存のサービスロールを使用することはできません。新しいロールが作成されます。

新しいサービスロールを作成すると、Amazon Q がDatadog認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

既存のサービスロールを使用する場合は、表示されるドロップダウンメニューからロールを選択します。サービスロールに、 で定義されているアクセス許可と信頼ポリシーがあることを確認しますサービス役割。

8. [設定の保存] を選択します。
9. Datadog プラグインページの「設定済みプラグイン」セクションにプラグインパネルが表示されると、ユーザーはプラグインにアクセスできます。

プラグインの認証情報を更新する場合は、現在のプラグインを削除して新しいプラグインを設定する必要があります。プラグインを削除すると、以前の仕様がすべて削除されます。新しいプラグインを設定するたびに、新しいプラグイン ARN が生成されます。

ユーザーアクセス許可を設定する

プラグインを使用するには、次のアクセス許可が必要です。

- コンソールで Amazon Q とチャットするアクセス許可。チャットに必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してくださいユーザーに Amazon Q とのチャットを許可する。
- `q:UsePlugin` アクセス許可

IAM ID に設定されたDatadogプラグインへのアクセスを許可すると、ID はプラグインによって取得可能なDatadogアカウント内のリソースへのアクセスを取得します。 Datadog ユーザーのアクセス許可はプラグインによって検出されません。プラグインへのアクセスを制御するには、IAM ポリシーでプラグイン ARN を指定します。

プラグインを作成または削除して再設定するたびに、新しい ARN が割り当てられます。ポリシーでプラグイン ARN を使用する場合は、新しく設定されたプラグインへのアクセスを許可する場合は更新する必要があります。

Datadog プラグイン ARN を見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定されたDatadogプラグインを選択します。プラグインの詳細ページで、プラグイン ARN をコピーします。この ARN をポリシーに追加して、Datadogプラグインへのアクセスを許可または拒否できます。

Datadog プラグインへのアクセスを制御するポリシーを作成する場合は、ポリシーでプラグインプロバイダーDatadogに を指定します。

プラグインアクセスを制御する IAM ポリシーの例については、「」を参照してくださいユーザーが 1 つのプロバイダーのプラグインとチャットできるようにする。

Datadog プラグインとチャットする

Datadog プラグインを使用するには、 Datadog または AWS アプリケーションモニターとケースに関する質問の冒**@datadog**頭に と入力します。Amazon Q からのフォローアップの質問や質問への回答には、 も含める必要があります**@datadog**。

以下は、Amazon Q Datadogプラグインを最大限に活用するためのユースケースの例と、関連する質問です。

- ワークロードでの Datadog 機能の使用について – 機能が特定の サービスとどのように連携するかについて質問します AWS 。 Datadog AWS Amazon Q では、最善の回答を提供するために何をしようとしているかの詳細を求められる場合があります。
  - **`@datadog how do I use APM on EC2?`**
- ケースとモニターの取得と要約 – 特定のケースやモニターについて質問するか、プロパティを指定して、モニターや作成日、ステータス、作成者などのケースに関する情報を取得します。プロパティの詳細については、 Datadogドキュメントの「プロパティ」を参照してください。
  - **`@datadog summarize the global outage case`**
  - **`@datadog summarize my top cases`**
- アラーム状態のモニターを確認する – Amazon Q Datadog プラグインに、アラーム状態の AWS アプリケーションモニターを見つけるよう依頼します。リストされているモニターに関する質問をフォローアップできます。
  - **`@datadog what monitors are in alarm?`**

- **`@datadog what is the status for monitor <monitor ID>?`**

Amazon Q Developer Wiz プラグインの設定

Wiz は、セキュリティ体制管理、リスク評価と優先順位付け、脆弱性管理を提供するクラウドセキュリティプラットフォームです。Wiz を使用して AWS アプリケーションを評価およびモニタリングする場合は、Amazon Q チャットの プラグインを使用して、 を離れWizることなく からインサイトにアクセスできます AWS Management Console。

プラグインを使用して、Wiz問題の特定と取得、最もリスクの高いアセットの評価、脆弱性や露出の把握を行うことができます。応答を受け取ったら、問題の修正方法など、フォローアップの質問をすることができます。

プラグインを設定するには、Wizアカウントから認証情報を指定して、Amazon Q と 間の接続を有効にしますWiz。プラグインを設定したら、Amazon Q チャットの質問の先頭**`@wiz`**に を追加してWizメトリクスにアクセスできます。

> **Warning**
>
> Wiz ユーザーアクセス許可は、Amazon Q のWizプラグインによって検出されません。管理者が AWS アカウントのWizプラグインを設定すると、そのアカウントのプラグインアクセス許可を持つユーザーは、プラグインによって取得可能なWizアカウントの任意のリソースにアクセスできます。
> IAM ポリシーを設定して、ユーザーがアクセスできるプラグインを制限できます。詳細については、「ユーザーアクセス許可を設定する」を参照してください。

前提条件

アクセス許可を追加する

プラグインを設定するには、次の管理者レベルのアクセス許可が必要です。

- Amazon Q Developer コンソールへのアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者に Amazon Q Developer コンソールの使用を許可する。
- プラグインを設定するアクセス許可。必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してください管理者にプラグインの設定を許可する。

認証情報の取得

開始する前に、 Wizアカウントの次の情報をメモしておきます。これらの認証情報は、プラグインを設定するときに AWS Secrets Manager シークレットに保存されます。

- API エンドポイント URL – にアクセスする URLWiz。例えば、`https://api.us1.app.Wiz.io/graphql`。詳細については、 Wizドキュメントの「API エンドポイント URL」を参照してください。
- クライアント ID とクライアントシークレット – Amazon Q が Wiz APIs を呼び出してアプリケーションにアクセスできるようにする認証情報。詳細については、 Wizドキュメントのクライアント ID とクライアントシークレットを参照してください。

シークレットとサービスロール

AWS Secrets Manager シークレット

プラグインを設定すると、Amazon Q はWiz認証情報を保存するための新しい AWS Secrets Manager シークレットを作成します。または、自分で作成した既存のシークレットを使用することもできます。

シークレットを自分で作成する場合は、次の認証情報が含まれていることを確認し、次の JSON 形式を使用してください。

```
{
   "ClientId": "<your-client-id>",
   "ClientSecret": "<your-client-secret>"
}
```

シークレットの作成の詳細については、「 AWS Secrets Manager ユーザーガイド」の「シークレットの作成」を参照してください。

サービス役割

Amazon Q Developer でWizプラグインを設定するには、Secrets Manager シークレットへのアクセス許可を Amazon Q に付与するサービスロールを作成する必要があります。Amazon Q はこのロールを引き受けて、Wiz認証情報が保存されているシークレットにアクセスします。

AWS コンソールでプラグインを設定する場合、新しいシークレットを作成するか、既存のシークレットを使用するかを選択できます。新しいシークレットを作成すると、関連付けられたサービス

ロールが自動的に作成されます。既存のシークレットと既存のサービスロールを使用する場合は、サービスロールにこれらのアクセス許可が含まれ、次の信頼ポリシーがアタッチされていることを確認します。必要なサービスロールは、シークレットの暗号化方法によって異なります。

シークレットがマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:{{your-account-id}}:secret:
[[secret-id]]"
            ]
        }
    ]
}
```

シークレットがカスタマーマネージド AWS KMS キーで暗号化されている場合は、次の IAM サービスロールが必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:us-east-1:{{accountId}}:secret:
{{secretId}}"
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:{{accountId}}:key/{{keyId}}",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
                }
            }
        }
    ]
}
```

Amazon Q がサービスロールを引き受けることを許可するには、サービスロールに次の信頼ポリシーが必要です。

> **Note**
>
> codewhisperer プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "q.amazonaws.com"
      },
      "Action": ["sts:AssumeRole", "sts:SetContext"],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}",
```

```
          "aws:SourceArn": "arn:aws:codewhisperer:{{region}}:
{{accountId}}:profile/{{profileId}}"
        }
      }
    }
  ]
}
```

サービスロールの詳細については、「 AWS Identity and Access Management ユーザーガイド」の「 AWS サービスにアクセス許可を委任するロールを作成する」を参照してください。

Wiz プラグインを設定する

Amazon Q Developer コンソールでプラグインを設定します。Amazon Q は、 に保存されている認証情報 AWS Secrets Manager を使用して、 とのやり取りを有効にしますWiz。

Wiz プラグインを設定するには、次の手順を実行します。

1. https://console.aws.amazon.com/amazonq/developer/home で Amazon Q Developer コンソールを開きます。
2. Amazon Q Developer コンソールのホームページで、設定を選択します。
3. ナビゲーションバーで、プラグインを選択します。
4. プラグインページで、Wizパネルのプラス記号を選択します。プラグイン設定ページが開きます。
5. API エンドポイント URL には、 にアクセスする API エンドポイントの URL を入力しますWiz。
6. Configure AWS Secrets Manager で、新しいシークレットを作成するか、既存のシークレットを使用するを選択します。Secrets Manager シークレットは、Wiz認証情報が保存される場所です。

   新しいシークレットを作成する場合は、次の情報を入力します。

   a. クライアント ID には、Wizアカウントのクライアント ID を入力します。
   b. クライアントシークレットには、Wizアカウントのクライアントシークレットを入力します。
   c. Amazon Q がWiz認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

既存のシークレットを使用する場合は、シークレットドロップダウンメニューからAWS Secrets Manager シークレットを選択します。シークレットには、前のステップで指定したWiz認証情報が含まれている必要があります。

必要な認証情報の詳細については、「認証情報の取得 」を参照してください。

7. IAM サービスロールを設定する AWS で、新しいサービスロールを作成する または既存のサービスロールを使用する を選択します。

> **Note**
>
> ステップ 6 で新しいシークレットを作成するを選択した場合、既存のサービスロールを使用することはできません。新しいロールが作成されます。

新しいサービスロールを作成すると、Amazon Q がWiz認証情報が保存されているシークレットにアクセスするために使用するサービスロールが作成されます。作成されたサービスロールを編集しないでください。

既存のサービスロールを使用する場合は、表示されるドロップダウンメニューからロールを選択します。サービスロールに、 で定義されているアクセス許可と信頼ポリシーがあることを確認しますサービス役割。

8. [設定の保存] を選択します。
9. Wiz プラグインページの「設定済みプラグイン」セクションにプラグインパネルが表示されると、ユーザーはプラグインにアクセスできます。

プラグインの認証情報を更新する場合は、現在のプラグインを削除して新しいプラグインを設定する必要があります。プラグインを削除すると、以前の仕様がすべて削除されます。新しいプラグインを設定するたびに、新しいプラグイン ARN が生成されます。

ユーザーアクセス許可を設定する

プラグインを使用するには、次のアクセス許可が必要です。

- コンソールで Amazon Q とチャットするアクセス許可。チャットに必要なアクセス許可を付与する IAM ポリシーの例については、「」を参照してくださいユーザーに Amazon Q とのチャットを許可する。

- `q:UsePlugin` アクセス許可

IAM ID に設定されたWizプラグインへのアクセスを許可すると、ID はプラグインによって取得可能なWizアカウント内のリソースへのアクセスを取得します。 Wiz ユーザーのアクセス許可はプラグインによって検出されません。プラグインへのアクセスを制御するには、IAM ポリシーでプラグイン ARN を指定します。

プラグインを作成または削除して再設定するたびに、新しい ARN が割り当てられます。ポリシーでプラグイン ARN を使用する場合は、新しく設定されたプラグインへのアクセスを許可する場合は更新する必要があります。

Wiz プラグイン ARN を見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定されたWizプラグインを選択します。プラグインの詳細ページで、プラグイン ARN をコピーします。この ARN をポリシーに追加して、Wizプラグインへのアクセスを許可または拒否できます。

Wiz プラグインへのアクセスを制御するポリシーを作成する場合は、ポリシーでプラグインプロバイダー`Wiz`に を指定します。

プラグインアクセスを制御する IAM ポリシーの例については、「」を参照してくださいユーザーが 1 つのプロバイダーのプラグインとチャットできるようにする。

Wiz プラグインとチャットする

Amazon Q Wizプラグインを使用するには、Wiz問題に関する質問の冒**`@Wiz`**頭に「」と入力します。Amazon Q からのフォローアップの質問または質問への回答には、 も含める必要があります**`@Wiz`**。

以下は、Amazon Q Wizプラグインを最大限に活用するためのユースケースの例と、関連する質問です。

- 重大度が重大な問題を表示する – Amazon Q Wizプラグインに、重大度が重大または高い問題を一覧表示するよう依頼します。プラグインは最大 10 個の問題を返すことができます。最も重大な問題の上位 10 件までを一覧表示するように に依頼することもできます。
  - **`@wiz what are my critical severity issues?`**
  - **`@wiz can you specify the top 5?`**
- 日付またはステータスに基づいて問題を一覧表示する – 作成日、期日、または解決日に基づいて問題を一覧表示するよう依頼します。ステータス、重要度、タイプなどのプロパティに基づいて問題を指定することもできます。

  - **`@wiz which issues are due before <date>?`**
  - **`@wiz what are my issues that have been resolved since <date>?`**
- セキュリティの脆弱性に関する問題を評価する – 問題にセキュリティの脅威をもたらしている脆弱性や露出について質問します。
  - **`@wiz which issues are associated with vulnerabilities or external exposures?`**

Amazon Q Developer Console-to-Code による AWS サービスの自動化

Console-to-Code とは

Console-to-Code は、Amazon Q Developer の機能であり、他の AWS サービスの使用を自動化するためのコードを記述するのに役立ちます。Console-to-Code はコンソールのアクションを記録し、生成 AI を使用して好みの言語と形式でコードを提案します。

サービスの階層

Console-to-Code は Amazon Q Developer の一部であるため、その使用には Amazon Q Developer のサービス階層が適用されます。

- 無料利用枠では、コンソールアクションを記録し、こうしたアクションに基づいて CLI コマンドを生成できる回数に、月単位の決まった上限はありません。ただし、記録されたアクション AWS CloudFormation に基づいて、 AWS CDK または で使用するコードを 1 か月に何回生成できるかには制限があります。

  無料利用枠にアクセスするには、 AWS Management Consoleにサインインします。月単位のコード生成の上限に達したら、さらにコードを生成するには、Pro 階層への認証を行う必要があります。
- Pro 階層では、 AWS CDK または CloudFormation のコードを生成できる回数に制限はありません。

  Pro 階層にアクセスするには、IAM アイデンティティセンターに登録されているユーザーであり、IAM アイデンティティセンターの ID を Amazon Q Developer Pro にサブスクライブしている必要があります。詳細については、「」を参照Amazon Q Developer Pro サブスクリプションへの認証するか、 AWS 管理者にお問い合わせください。

料金階層の詳細については、Amazon Q Developer の料金ページを参照してください。

> **Note**
>
> アクションを記録した場合でも、必要に応じてアクション自体の料金が請求されます。例えば、Amazon EC2 インスタンスのプロビジョニングを記録した場合でも、インスタンスに対して料金が請求されます。アクションの記録に追加料金は発生しません。

サポートされるコードの形式

現在、Console-to-Code は次の言語と形式で Infrastructure as Code (IaC) を生成できます。

- CDK Java
- CDK Python
- CDK TypeScript
- CloudFormation JSON
- CloudFormation YAML

Console-to-Code は使用できる場所

複数のサービスをまたいで Console-to-Code を使用する

Console-to-Code は複数のサービスで機能します。また、ブラウザタブが開いている限り、独自の状態を保存します。

例えば、ウェブサーバーの完全なセットアップ中にアクションを記録するとします。

- Amazon VPC コンソールでは、2 つのサブネット (1 つのパブリックと 1 つのプライベート) 、セキュリティグループ、NACLs、カスタムルーティングテーブル、インターネットゲートウェイをプロビジョニングします。
- Amazon EC2 コンソールで、Amazon EC2 インスタンスをプロビジョニングし、パブリックサブネットに配置します。
- Amazon RDS コンソールで、Amazon RDS DB インスタンスをプロビジョニングし、プライベートサブネットに配置します。

コンソールのさまざまな部分でアクションを実行し、異なる AWS サービスを使用している場合でも、Console-to-Code はそれらを 1 回の録画に含めることができます。

AWS Console-to-Code をサポートする サービス

現在、Console-to-Code は、 AWS マネジメントコンソールを次のサービスで使用する場合にアクションの記録に使用できます。

- Amazon EC2
- Amazon VPC
- Amazon RDS

Console-to-Code を使用するアクセス許可を付与する

Console-to-Code を使用するには、次のアクセス許可が必要です。

- Console-to-Code を使用するための `q:GenerateCodeFromCommands`。必要なアクセス許可を付与する IAM ポリシーの例については、「Amazon Q を使用して CLI コマンドからコードを生成することをユーザーに許可する」を参照してください。
- 記録するアクションを実行するためのアクセス許可。

Console-to-Code を使用する

Console-to-Code の使用には 3 つのステップがあります。

ステップ 1: 記録を開始する

Console-to-Code で記録を開始する手順は次のとおりです。

1. 統合サービス (Amazon VPC、Amazon RDS、または Amazon EC2) のいずれかのコンソールに移動します。
2. ブラウザウィンドウの右端で、Console-to-Codeアイコンを選択します。
3. Console-to-Code サイドパネルで、[記録を開始] を選択します。

ステップ 2: アクションを実行する

統合サービスのコンソールで、記録するアクションを実行します。

Console-to-Code サイドパネルは、独自の状態を保持します。統合サービスのコンソール間を移動して、複数のサービスのアクションを含む 1 つの記録を作成できます。

Console-to-Code サイドパネルには、Console-to-Code セッションが終了するまでアクションが保持されます。セッションは、ブラウザタブを閉じたとき、または AWS Management Console セッションが終了したときのいずれか早い方で終了します。

コードに変換するアクションの実行が完了したら、Console-to-Code パネルの上部から [停止] を選択します。

ステップ 3: CLI コマンドを収集し、コードを生成する

ステップ 3a またはステップ 3b のいずれかを実行できます。

ステップ 3a: CLI コマンドを収集する

Console-to-Code を使用してアクションに基づいて CLI コマンドを生成する手順は次のとおりです。

1. Console-to-Code パネルで、記録したアクションを確認します。

   Console-to-Code パネルの上部にあるドロップダウン、検索ボックス、またはフィルターウィジェットを使用して、記録されたアクションをフィルタリングできます。
2. Console-to-Code パネルの上部で、[CLI を表示] 設定のトグルを切り替えます。これにより、選択した各アクションに対応する CLI コマンドが表示されます。
3. 使用するテンプレートを選択します。次のステップでは、チェックボックスの付いたコマンドのみが使用されます。
4. CLI コマンドを統合します。Console-to-Code パネルの下部近くで、アコーディオンアイコン (^) を選択します。選択した CLI コマンドがパネルに個別に表示されます。
5. 選択したコマンドをコピーまたはダウンロードします。

の詳細については AWS CLI、 ユーザーガイドの「 とは AWS Command Line Interface」を参照してください。 AWS Command Line Interface

ステップ 3b: コードを生成する

1. Console-to-Code パネルで、記録したアクションを確認します。Console-to-Code パネルの上部にあるドロップダウン、検索ボックス、またはフィルターウィジェットを使用して、記録されたアクションをフィルタリングできます。
2. コードに変換するアクションを選択します。次のステップでは、チェックボックスの付いたアクションのみが使用されます。
3. 生成するコードのタイプを指定します。Console-to-Code パネルの右下にある逆のドロップダウンメニューから、生成するコードの言語と (該当する場合) 形式を選択します。
4. [選択した言語を生成] を選択します。

   生成されたコードが、同等の CLI コマンドと共にに表示されます。

Amazon Q Developer でコンソールでの一般的なエラーを診断する

では AWS Management Console、Amazon Q Developer は、アクセス許可の不足、誤った設定、サービス制限の超過など、AWS のサービスの使用中に発生する一般的なエラーを診断します。Amazon Q は、AWS コンソールで次のサービスを使用しているときに発生するエラーをトラブルシューティングします。

- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Simple Storage Service (Amazon S3)
- AWS Lambda

さらに、Amazon Q はすべての AWS コンソールページで IAM アクセス許可エラーをトラブルシューティングし、一部のサービスではサービス固有のエラーの数を制限します AWS 。Amazon Q では、以前のエラー診断セッションの履歴は保持されません。

Amazon Q でエラーを診断できない場合は、Amazon Q を使用して サポートでサポートケースを作成できます。詳細については、「Amazon Q Developer を使用して とチャットする サポート」を参照してください。Amazon Q エラー診断機能に固有の問題がある場合は、低評価アイコンを使用して問題を報告できます。

アクセス許可を追加する

コンソールエラーの診断に必要なアクセス許可を付与する IAM ポリシーについては、「ユーザーが Amazon Q でコンソールエラーをトラブルシューティングすることを許可する」を参照してください。

コンソールでの一般的なエラーの診断

Amazon Q を使用して のエラーを診断するには AWS Management Console、次の手順を使用します。

1. Amazon Q で解決を支援できるエラーが発生した場合、エラーメッセージに [Amazon Q で診断] ボタンが表示されます。Amazon Q を使用してエラーを診断する場合は、[Amazon Q で診断] を選択して続行します。
2. Amazon Q が最初にエラーに関する情報を提供するウィンドウが表示されます。次に、エラーを解決するために実行できる一連のステップを提供します。Amazon Q が指示を生成するために数秒かかる場合があります。
3. 高評価アイコンまたは低評価アイコンを使用してフィードバックを送信できます。詳細なフィードバックを送信するには、アイコンをクリックした後に表示される [もっと詳しく] ボタンをクリックします。

Amazon Q Developer を使用して とチャットする サポート

Amazon Q Developer を使用して、 AWS Management Consoleを含む の サポート どこからでもサポートケースと連絡先を作成できます AWS Support Center Console。Amazon Q は、ユーザーの会話のコンテキストを使用し、ユーザーに代わってサポートケースの下書きを自動的に作成します。最近の会話もサポートケースの説明に追加されます。ケースを作成すると、Amazon Q は、同じインターフェイスのライブチャットなど、お客様が選択した方法でお客様をサポートエージェントに転送できます。

Amazon Q でサポートケースを作成すると、ケースはサポートセンターコンソールでも更新されます。Amazon Q で作成されたケースの更新情報を追跡するには、サポートセンターコンソールを使用します。

サポート 使用できる のタイプは、 のサポートプランによって異なります AWS アカウント。すべての AWS ユーザーは、ベーシックサポートプランの一環として、アカウントと請求のサポートにアクセスできます。テクニカルサポートに関する質問については、ベーシックサポートプラン以外の

サポートプランを持つユーザーのみが Amazon Q サポート に連絡できます。詳細については AWS サポート、AWS サポート 「 ユーザーガイド」の「 の開始方法 AWS サポート」を参照してください。

> **Tip**
>
> サポートチケットを作成する前に、Amazon Q に問題を解決するよう依頼してみてください。詳細については、「リソースのトラブルシューティングを Amazon Q に依頼する」を参照してください。利用可能な場合は、Amazon Q で診断ボタンを試すこともできます。詳細については、「コンソールエラーの診断」を参照してください。

前提条件

Amazon Q でケースを作成するには、次の要件を満たしている必要があります。

- ベーシックサポートプランより上位のサポートプランを持っていること。基本サポートプラン以外のサポートプランを持つユーザーのみが Amazon Q サポート に連絡できます。
- Amazon Q とのチャットに必要なアクセス許可があること。詳細については、「ユーザーに Amazon Q とのチャットを許可する」を参照してください。
- サポート ケースを作成するアクセス許可があります。詳細については、「 サポート センターへのアクセスを管理する」を参照してください。

適切なサービスを指定する

Amazon Q でサポートケースを作成すると、質問に基づいてサービスフィールドが入力されます。Amazon Q が間違ったサービスを選択した場合は、正しいサービスでケースを更新します。複数のサービスに関係する質問の場合は、最も関連性の高いサービスを選択します。

別の の一部である Amazon Q 機能 サポート について に問い合わせるには AWS のサービス、Amazon Q ではなく AWS のサービス別のサポートケースを作成します。例えば、Amazon VPC Reachability Analyzer で Amazon Q ネットワークのトラブルシューティングを使用している場合は、サポートケースでサービスの Amazon VPC を選択します。

Amazon Q Developer または Amazon Q Business の機能 サポート について に問い合わせるには、Amazon Q のサポートケースを作成します。

サポートケースの作成

Amazon Q で サポート ケースを作成するには、次の手順を実行します。

1. Amazon Q を使用して サポート ケースを作成するには、次の 2 つの方法があります。

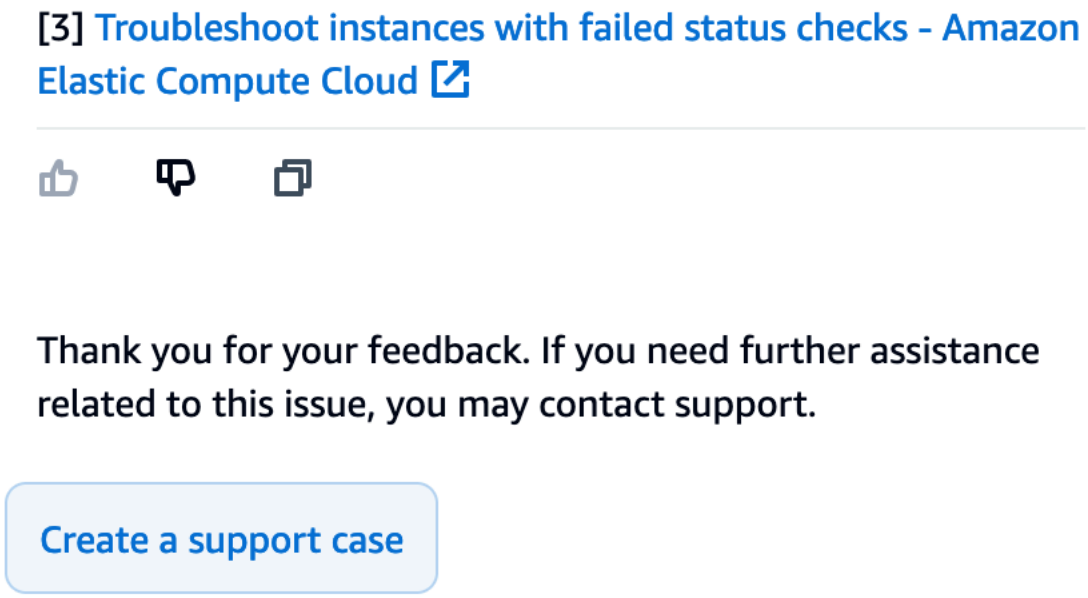
   a. 「誰かと話したい」や「サポートを受ける」などの質問を入力して、直接サポートを求めることができます。

      Amazon Q がサポートケースを作成するためのコンテキストを詳しく提示するには、直接サポートをリクエストするときに情報を追加できます。以下は、リクエストで詳細情報を提供する例です。

      「踏み台のインスタンスに接続できません。再起動して新しいキーペアの生成を試みましたが、まだ何も解決されません。これは今朝、計画的なデプロイの後に発生しました。ネットワークに関連するその他の変更は行われていないことを確認しました。誰かと話せますか？」

   b. Amazon Q の応答が役に立たなかった場合は、応答に対して低評価アイコンをクリックし、低評価のフィードバックを提供する理由を選択します。に問い合わせるには サポート、サポートケースの作成を選択します。

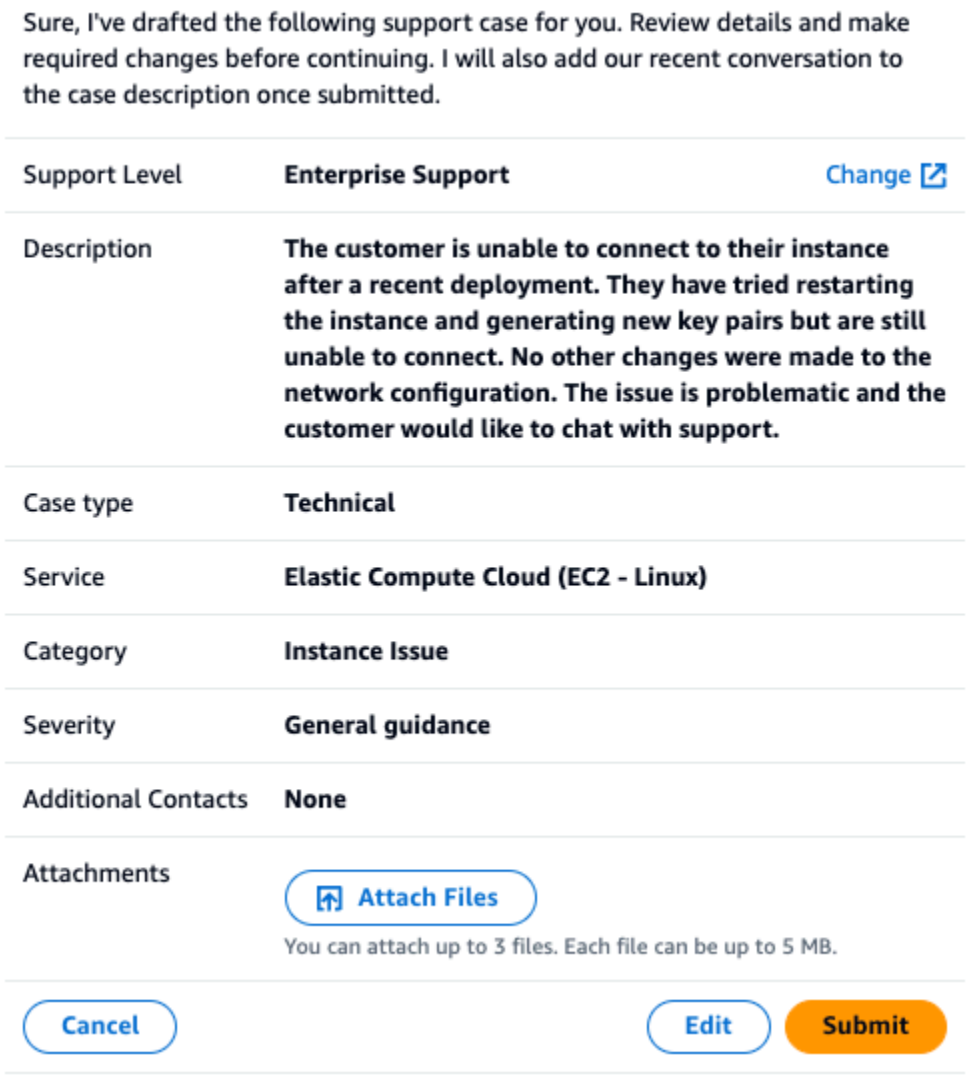
      次の図は、Amazon Q チャットパネルの [サポートケースの作成] ボタンを示しています。これは、フィードバックを送った後に表示されます。



2. チャットパネルにサポートケースが表示されます。サポートをリクエストする前に Amazon Q と会話していた場合、会話のコンテキストを使用してケースのフィールドへの自動入力が行われます。サポートケースのフィールドを更新するには、[編集] を選択します。問題内容を裏付けるファイルを添付することもできます。

   サポートをリクエストする前に Amazon Q とチャットしていない場合や、Amazon Q がサポートケースのフィールドに入力できない場合は、サポートケースの情報をケースに手動で入力できます。

   次の図は、Amazon Q チャットパネルの記入されているサポートケースの例です。



3. サポートケースに適切に入力されていることを確認したら、[送信] を選択してサポートケースを作成します。サポートケースを作成する必要がなくなった場合は、[キャンセル] を選択します。

4. に連絡するには サポート、使用する方法を選択します。ケースの詳細に応じてライブサポートエージェントからチャット、メール、電話をリクエストすることができます。

   a. チャット – エージェントとのチャットを選択した場合、サポートエージェントが会話に参加します。いつでも [このチャットを終了] を選択し、サポートエージェントとのチャットを終了できます。

      ページを更新したり、別のコンソールに移動したり、またはセッションの有効期限切れによりコンソールからサインアウトしたりすると、会話は終了します。

      チャットパネルを最小化したり、ページを離れたりすると、通知を見逃してしまい、操作がないために接続が切断される場合があります。サポートチャット中は、チャットパネルを開いたままにしておくことをお勧めします。

   b. E メール – E メールメッセージをエージェントに送信することを選択した場合、サポートエージェントは に関連付けられている E メールアドレスでお客様に連絡します AWS アカウント。

   c. 電話 – エージェントに電話することを選択した場合は、プロンプトが表示されたら電話番号を入力して [送信] を選択すると、呼び出しキューに追加されます。

5. フィードバックを残すか、[スキップ] を選択して Amazon Q のチャットパネルに戻ることができます。

フィードバックを残す

サポートチャットが終了したら、オプションとしてフィードバックを残すことができます。

体験を評価して追加フィードバックを入力したら、[フィードバックを送信] を選択します。

IDE での Amazon Q Developer の使用

統合開発環境 (IDEsで Amazon Q Developer を使用して、ソフトウェア開発のニーズについて学び AWS 、サポートを受けることができます。IDEs では、Amazon Q には、コードの構築、生成と更新、セキュリティスキャン、コードの最適化とリファクタリングに関する質問に回答するなど AWS、ソフトウェア開発のさまざまな側面にわたってガイダンスとサポートを提供する機能が含まれています。

IDE に Amazon Q をインストールするには、IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする「」を参照してください。

トピック

- サポートされている IDE と利用可能な機能
- IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする
- コードに関する Amazon Q Developer とのチャット
- Amazon Q Developer によるインライン提案の生成
- Amazon Q Developer を使用した IDE でのコードの変換
- Amazon Q Developer を使用した機能の開発
- Amazon Q を使用したユニットテストの生成
- Amazon Q Developer でコードを確認する
- Amazon Q Developer を使用したドキュメントの生成
- IDE の Amazon Q Developer でサポートされている言語

サポートされている IDE と利用可能な機能

利用アクセスできる機能は、Amazon Q を使用する IDE によって異なります。次の表では、Amazon Q でサポートされている IDE と、各 IDE での機能の提供状況と制限をまとめています。

言語サポートが明記されていない場合、IDE は「サポートされている言語」トピックに記載されている言語をサポートします。

機能	VSCode	JetBrains	Eclipse	Visual Studio
チャット	はい	あり	あり	あり

機能	VSCode	JetBrains	Eclipse	Visual Studio
エージェントコーディング	はい	あり	あり	あり
MCP サーバー	はい	あり	あり	あり
チャットのコンテキスト	はい	あり	あり	あり
インラインチャット	はい	あり	あり	なし
チャットのワークスペースコンテキスト	はい	あり	あり	あり
インライン提案	はい	あり	あり	あり
[変換]	はい	あり	なし	あり
機能開発	はい	あり	なし	なし
ユニットテスト生成	はい	あり	なし	なし
コードレビュー	はい	あり	なし	はい (C# のみ)
ドキュメントの生成	はい	あり	なし	なし
カスタマイズ	はい	あり	あり	なし

AWS コーディング環境でインライン提案を生成することもできます。詳細については、「AWS コーディング環境でインライン提案を生成する」を参照してください。

IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする

統合開発環境 (IDE) で Amazon Q Developer をセットアップする手順は次のとおりです。Amazon Q 拡張機能またはプラグインをインストールしたら、IAM アイデンティティセンターまたは AWS ビルダー IDを使用して認証を行います。Builder ID で認証することで、 AWS アカウントなしで Amazon Q を無料で使用できます。

開始するには、IDE の Amazon Q 拡張機能またはプラグインをダウンロードします。

- の Amazon Q をダウンロードする Eclipse
- の Amazon Q をダウンロードする Visual Studio Code
- Amazon Q for JetBrains IDEsをダウンロードする
- で Amazon Q をダウンロードする AWS Toolkit for Visual Studio

> **Note**
>
> 通常、IAM アイデンティティセンターを介して認証が行われるセッションのデフォルトの有効期間は 8 時間です。ただし、Amazon Q の場合、デフォルトのセッションは 90 日間続きます (2024 年 4 月 18 日以降に IAM アイデンティティセンターをセットアップした場合)。詳細については、「 AWS IAM Identity Center ユーザーガイド」の「IDE」の「Amazon Q のセッション期間を延長する方法」を参照してください。

サインインして認証を行うには、このセクションに記載されている手順を完了します。

ステップ

- 前提条件: サポートされている IDE バージョン
- Eclipse IDE での認証
- JetBrains IDE での認証
- Visual Studio Code での認証
- Visual Studio での認証
- AWS コンソールでの IAM プリンシパルの使用

前提条件: サポートされている IDE バージョン

- Amazon Q でEclipseサポートされている の最小バージョンは 2024 年 6 月 (4.32) です。
- Amazon Q でサポートされている JetBrains IDEs の最小バージョン (IntelliJと を含むPyCharm) は 232.1 です。
- Amazon Q でサポートされている Visual Studio Code の最小バージョンは 1.85.0 です。

- Amazon Q では、Visual Studio for Windows のみがサポートされています。サポートされている Visual Studio の最小バージョンは Visual Studio 2022 バージョン 17.7 です。すべての Visual Studio 2022 エディションがサポートされています。

Eclipse IDE での認証

Amazon Q Developer Pro サブスクリプションでは、 AWS ビルダー ID または IAM Identity Center を使用して無料で認証できます。認証方法を選択すると、Eclipse で Amazon Q の使用を開始する手順が表示されます。

Builder ID

この手順では、Builder ID は必要ありません。Builder ID にまだサインアップしていない場合は、サインインプロセス中にサインアップできます。

1. に Amazon Q プラグインをインストールしますEclipse。
2. IDE の右上隅にある Amazon Q アイコンを選択します。
3. IDE の下部に Amazon Q タブが開きます。「サインインオプションの選択」で「個人アカウント」を選択し、「続行」を選択します。ブラウザにリダイレクトされます。
4. ブラウザの指示に従って Builder ID で認証を行います。認証が完了したら、IDE Eclipse に戻ります。
5. Amazon Q の使用を開始するには、Amazon Q アイコンを選択してチャット Amazon Q パネルを開きます。

IAM Identity Center

この手順を開始する前に、管理者は次のことを行う必要があります。

- IAM アイデンティティセンターで ID を作成する
- Amazon Q Developer Pro にその ID をサブスクライブする

ID を Amazon Q Developer Pro にサブスクライブしたら、次の手順で認証を行います。

1. に Amazon Q プラグインをインストールしますEclipse。
2. IDE の右上隅にある Amazon Q アイコンを選択します。

3. IDE の下部に Amazon Q タブが開きます。「サインインオプションの選択」で、会社のアカウントを選択し、「続行」を選択します。
4. 管理者が Amazon Q サブスクリプションコンソールから取得した開始 URL を入力します。
5. 管理者が IAM Identity Center インスタンスをセットアップ AWS リージョン する を選択します。
6. [続行] をクリックしてください。ブラウザにリダイレクトされます。
7. ブラウザの指示に従って、IAM アイデンティティセンターで認証を行います。認証が完了したら、IDE Eclipse に戻ります。
8. Amazon Q の使用を開始するには、Amazon Q アイコンを選択してチャット Amazon Q パネルを開きます。

JetBrains IDE での認証

Amazon Q Developer Pro サブスクリプションでは、 AWS ビルダー ID または IAM Identity Center を使用して無料で認証できます。認証方法を選択すると、JetBrains IDE で Amazon Q の使用を開始する手順が表示されます。

Builder ID

この手順では、Builder ID は必要ありません。Builder ID にまだサインアップしていない場合は、サインインプロセス中にサインアップできます。

1. IDE に Amazon Q プラグインをインストールします。 JetBrains
2. IDE で Amazon Q アイコンをクリックします。

   アイコンはデフォルトでインターフェイスの側面に表示されます。
3. ブラウザの指示に従って Builder ID で認証を行います。
4. Amazon Q の使用を開始するには、Amazon Q アイコンをクリックして Amazon Q とチャットするか、IDE の下部にあるナビゲーションバーから [Amazon Q] を選択します。

IAM Identity Center

この手順を開始する前に、管理者は次のことを行う必要があります。

- IAM アイデンティティセンターで ID を作成する
- Amazon Q Developer Pro にその ID をサブスクライブする

ID を Amazon Q Developer Pro にサブスクライブしたら、次の手順で認証を行います。

1. IDE に Amazon Q プラグインをインストールします。 JetBrains
2. IDE で Amazon Q アイコンをクリックします。

   アイコンはデフォルトでインターフェイスの側面に表示されます。
3. 会社のアカウントを選択します。
4. 管理者が Amazon Q サブスクリプションコンソールから取得した開始 URL を指定します。
5. 管理者が IAM Identity Center インスタンスをセットアップ AWS リージョン する を指定します。
6. [続行] をクリックしてください。フォーカスがウェブブラウザに切り替わります。
7. ブラウザの指示に従って IAM Identity Center で認証し、IDE に戻ります。
8. 管理者が複数の Amazon Q Developer プロファイルを設定している場合は、アクセスできるプロファイルが表示されます。現在の作業ニーズを満たすプロファイル、または管理者が使用を指示したプロファイルを選択します。プロファイルの詳細については、「Amazon Q Developer プロファイルとは」を参照してください。

   使用可能なプロファイルが 1 つしかない場合、そのプロファイルは自動的に選択され、Amazon Q の使用を開始できます。

   Amazon Q Developer プロファイルを変更するには、IDE の下部から Amazon Q を選択し、プロファイルの変更を選択します。表示されるウィンドウから、使用するプロファイルを選択します。
9. Amazon Q の使用を開始するには、Amazon Q アイコンをクリックして Amazon Q とチャットするか、IDE の下部にあるナビゲーションバーから [Amazon Q] を選択します。

Visual Studio Code での認証

Amazon Q Developer Pro サブスクリプションでは、 AWS ビルダー ID または IAM Identity Center を使用して無料で認証できます。認証方法を選択すると、VS Code で Amazon Q の使用を開始する手順が表示されます。

Builder ID

この手順では、Builder ID は必要ありません。Builder ID にまだサインアップしていない場合は、サインインプロセス中にサインアップできます。

1. に Amazon Q 拡張機能をインストールしますVS Code。
2. IDE で Amazon Q アイコンをクリックします。

   アイコンはデフォルトでインターフェイスの側面に表示されます。
3. ブラウザの指示に従って Builder ID で認証を行います。
4. Amazon Q の使用を開始するには、Amazon Q アイコンをクリックして Amazon Q とチャットするか、IDE の下部にあるナビゲーションバーから [Amazon Q] を選択します。

IAM Identity Center

この手順を開始する前に、管理者は次のことを行う必要があります。

- IAM アイデンティティセンターで ID を作成する
- Amazon Q Developer Pro にその ID をサブスクライブする

ID を Amazon Q Developer Pro にサブスクライブしたら、次の手順で認証を行います。

1. に Amazon Q 拡張機能をインストールしますVS Code。
2. IDE で Amazon Q アイコンをクリックします。

   アイコンはデフォルトでインターフェイスの側面に表示されます。
3. 会社のアカウントを選択します。
4. 管理者が Amazon Q サブスクリプションコンソールから取得した開始 URL を指定します。
5. 管理者が IAM Identity Center インスタンスをセットアップする AWS リージョン を指定します。
6. [続行] をクリックしてください。フォーカスがウェブブラウザに切り替わります。
7. ブラウザの指示に従って IAM Identity Center で認証し、IDE に戻ります。
8. 管理者が複数の Amazon Q Developer プロファイルを設定している場合は、アクセスできるプロファイルが表示されます。現在の作業ニーズを満たすプロファイル、または管理者が使用を指示したプロファイルを選択します。プロファイルの詳細については、「Amazon Q Developer プロファイルとは」を参照してください。

   使用可能なプロファイルが 1 つしかない場合、そのプロファイルは自動的に選択され、Amazon Q の使用を開始できます。

Amazon Q Developer プロファイルを変更するには、IDE の下部から Amazon Q を選択し、プロファイルの変更を選択します。コマンドパレットから、使用するプロファイルを選択します。

9. Amazon Q の使用を開始するには、Amazon Q アイコンをクリックして Amazon Q とチャットするか、IDE の下部にあるナビゲーションバーから [Amazon Q] を選択します。

Visual Studio での認証

Toolkit for Visual Studio から AWS アカウントに接続するには、次の手順を実行して AWS Toolkit ユーザーインターフェイスの開始方法 (接続 UI) を開きます。

1. Visual Studio のメインメニューから [拡張機能] を展開し、[AWS Toolkit] を展開します。
2. [AWS Toolkit] のメニューオプションから、[使用開始] を選択します。
3. Toolkit AWS 接続 UI の開始方法が Visual Studio で開きます。

Amazon Q Developer Pro サブスクリプションでは、 AWS ビルダー ID または IAM Identity Center を使用して無料で認証できます。認証方法を選択すると、Visual Studio で Amazon Q の使用を開始する手順が表示されます。

Builder ID

1. Visual Studio から、メインメニューから拡張機能を展開し、AWS ツールキットサブメニューを展開します。
2. [開始方法] を選択します。Visual Studio エディタウィンドウで開始方法タブが開きます。
3. Amazon Q セクションで、有効化 を選択します。
4. 無料利用枠セクションで、サインアップまたはサインインボタンを選択します。
5. デフォルトのウェブブラウザで Authorize AWS リクエストポータルを開くことを確認します。
6. デフォルトのウェブブラウザのプロンプトに従います。認証プロセスが完了すると通知があり、ブラウザウィンドウを安全に閉じることができます。

IAM Identity Center

1. Visual Studio から、メインメニューから拡張機能を展開し、AWS ツールキットサブメニューを展開します。

2. [開始方法] を選択します。Visual Studio エディタウィンドウで開始方法タブが開きます。
3. Amazon Q セクションで、有効化を選択します。認証するには、Professional Tier セクションに入力します。
4. 認証情報プロファイルは、会社または組織の管理者が提供するプロファイル名、開始 URL、プロファイルリージョン、または SSO リージョンで構成されます。IAM Identity Center 認証情報の詳細については、IAM Identity Center ユーザーガイドの「IAM Identity Center とは」を参照してください。

   既存の認証情報プロファイルがある場合は、Professional 階層パネルのドロップダウンメニューからそのプロファイルを選択し、Connect を選択します。

   新しい認証情報プロファイルを作成するには、Professional 階層セクションから次のフィールドに入力します。

   a. プロファイル名テキストフィールドに、認証する IAM Identity Center プロファイルの名前を入力します。
   b. Start URL テキストフィールドに、IAM Identity Center 認証情報にアタッチされている開始 URL を入力します。
   c. プロファイルリージョン (デフォルトは us-east-1) ドロップダウンメニューから、認証 AWS リージョン する IAM Identity Center ユーザープロファイルで定義されている を選択します。
   d. SSO リージョン (デフォルトは us-east-1) ドロップダウンメニューから、IAM アイデンティティセンターの認証情報で定義されている SSO リージョンを選択し、接続ボタンを選択して IAM AWS アイデンティティセンターでログインダイアログを開きます。
5. デフォルトのウェブブラウザで Authorize AWS リクエストポータルを開くことを確認します。
6. デフォルトのウェブブラウザのプロンプトに従います。認証プロセスが完了すると通知があり、ブラウザウィンドウを安全に閉じることができます。
7. Amazon Q へのサインインウィンドウが表示されます。認証情報プロファイルドロップダウンで、前のステップで認証に使用したプロファイルを選択します。
8. 管理者が複数の Amazon Q Developer プロファイルを設定している場合、ドロップダウンメニューから Q Developer プロファイルを選択するように求められます。現在の作業ニーズを満たすプロファイル、または管理者が使用を指示したプロファイルを選択します。プロファイルの詳細については、「Amazon Q Developer プロファイルとは」を参照してください。

使用可能なプロファイルが 1 つしかない場合、そのプロファイルは自動的に選択され、Amazon Q の使用を開始できます。

Amazon Q Developer プロファイルを変更するには、IDE の下部から Amazon Q を選択し、Q Developer Profile の変更を選択します。表示されるウィンドウから、使用するプロファイルを選択します。

チャットウィンドウの右上隅にあるオーバーフローメニューを選択し、Q Developer Profile の変更を選択することで、プロファイルを変更することもできます。

Toolkit for Visual Studio での認証の詳細については、「AWS Toolkit for Visual Studio ユーザーガイド」の「Getting Started」を参照してください。

AWS コンソールでの IAM プリンシパルの使用

使用方法によっては AWS、IAM 認証情報を使用してすべての AWS サービスのコンソールにサインインすることに慣れている場合があります。ただし、IDE で Amazon Q Developer を IAM プリンシパルとして、または IAM ロールで使用することはできません。IAM アイデンティティセンターまたは Builder ID の認証情報を使用して認証を行う必要があります。

コードに関する Amazon Q Developer とのチャット

統合開発環境 (IDE) で Amazon Q Developer とチャットして、 でのビルドについて質問 AWS したり、ソフトウェア開発の支援を受けたりします。Amazon Q は、コーディングの概念とコードスニペットについて説明できるほか、コードやユニットテストを生成し、デバッグやリファクタリングなどでコードを改善できます。

トピック

- サポートされているすべての IDEs での Amazon Q Developer のエージェントコーディングエクスペリエンス
- 自然言語での Amazon Q とのチャット
- IDE で Amazon Q の操作
- トピックおよび質問の例
- Amazon Q からの応答に関する問題の報告
- Amazon Q Developer でのコードの説明と更新
- Amazon Q Developer とのインラインチャット

- IDE での Amazon Q Developer チャットへのコンテキストの追加
- Amazon Q Developer でのチャット履歴の圧縮
- Amazon Q Developer の会話履歴の表示、削除、エクスポート
- Amazon Q Developer とのチャットでのショートカットキーの使用
- IDEs での Amazon Q チャットのモデルの選択

サポートされているすべての IDEs での Amazon Q Developer のエージェントコーディングエクスペリエンス

IDE の Amazon Q Developer では、エージェントコーディングはデフォルトでオンになっています。

エージェントコーディングを使用すると、Amazon Q はコーディングパートナーとして機能し、開発時にエージェント的にチャットします。

プロジェクトについて Amazon Q と話し合うと、シェルコマンドの提案が提供されます。場合によっては、これらのコマンドを低リスクと見なすと、単独で実行されます。

Amazon Q にコードの改善を依頼すると、コードは直接改善されます。差分で変更を表示し、オプションで元に戻すことができます。

Amazon Q がプロンプト間で考えている間、入力ウィンドウに指示を引き続き追加するオプションがあります。

チャットパネルの下部にある</>アイコンを使用して、エージェントコーディングのオンとオフを切り替えることができます。

自然言語での Amazon Q とのチャット

Amazon Q Developer は、IDE でチャットするときに多言語サポートを提供します。サポートされている自然言語には、北京語、フランス語、ドイツ語、イタリア語、日本語、スペイン語、韓国語、ヒンディー語、ポルトガル語があり、より多くの言語を使用できます。この機能を利用するには、任意の自然言語を使用して IDE で Amazon Q との会話を開始できます。Amazon Q は言語を自動的に検出し、適切な言語でレスポンスを提供します。

IDE で Amazon Q の操作

IDE の Amazon Q のチャットパネルから、Amazon Q への質問、コードの更新、クイックコマンドでアクションの開始をすべて行うことができます。

Amazon Q とのチャットを開始するには、IDE のナビゲーションバーから Amazon Q アイコンを選択し、テキストバーに質問を入力します。Visual Studio で Amazon Q とのチャットを開始するには、メインメニューから [ビュー] を選択し、[Amazon Q チャット] を選択します。

Amazon Q に質問をすると、プログラミング言語やファイルパスを含め、IDE で開かれている現在のファイルがコンテキストとして使用されます。

Amazon Q の応答にコードが含まれている場合、コードをコピーするか、[カーソルの場所に挿入] を選択してファイルに直接挿入できます。Amazon Q の回答には、出典へのインラインリファレンスが含まれる場合があります。ソースのリストを表示するには、レスポンスの下部にある [ソース] セクションを展開します。

Amazon Q は、特定のセッション内の会話をコンテキストとして維持し、将来の回答に役立てます。セッション中は、フォローアップの質問をしたり、以前の質問や回答を参照したりできます。Amazon Q との新しい会話を始めるには、パネルの新しいタブを開きます。一度に最大 10 個のタブを開くことができます。Amazon Q では、異なる会話間でコンテキストは保持されません。

チャットコマンド

チャットパネルに次のコマンドを入力すると、IDE に応じて Amazon Q の機能にアクセスできます。IDE で使用できる Amazon Q の機能の詳細については、「サポートされている IDE」を参照してください。

- **`/transform`** – プロジェクト全体のコード言語バージョンを更新するために使用します。詳細については、「Java バージョンのアップグレード」を参照してください。
- **`/dev`** – Amazon Q で機能を開発するための実装計画を立てるために使用します。詳細については、「機能の開発 (/dev)」を参照してください。
- **`/test`** - このコマンドを使用して、Amazon Q でコードのユニットテストを生成します。詳細については、「」を参照してくださいユニットテストの生成 (/test)。
- **`/review`** - このコマンドを使用して、Amazon Q のセキュリティ脆弱性とコード品質の問題がないかコードベースを確認します。詳細については、「」を参照してくださいコードの確認 (/レビュー) 。
- **`/doc`** - このコマンドを使用して、Amazon Q でプロジェクトベースの READMEs「」を参照してくださいドキュメントの生成 (/doc)。
- **`/clear`** - 現在の会話をクリアするために使用します。これにより、チャットパネルから以前の会話がすべて削除され、Amazon Q の以前の会話に関するコンテキストがクリアされます。

- **`/compact`** - コンテキストウィンドウが容量制限に近づいたときにチャット履歴を圧縮するには、このコマンドを使用します。これにより、重要な情報を保持しながら、会話の簡潔な概要が作成されます。
- **`/help`** – Amazon Q で対応可能および対応不可能なことの概要、質問例、クイックコマンドを確認するために使用します。

トピックおよび質問の例

IDEs 内では、Amazon Q はコードの生成に加えて、 AWS のサービス とソフトウェア開発に関する質問に回答できます。Amazon Q は、特に次の分野に関する質問に回答するために便利です。

- AWS のサービス 選択 AWS、制限、ベストプラクティスを含む の構築
- プログラミング言語の構文やアプリケーション開発を含む一般的なソフトウェア開発の概念
- コードの説明、コードのデバッグ、ユニットテストの記述を含むコードの記述

IDE で Amazon Q を最大限に活用するための質問例は次のとおりです。

- AWS にデプロイする前に、Lambda 関数の問題をローカルでデバッグする方法を教えてください
- スケーラブルなウェブアプリケーションのバックエンドとして、 AWS Lambda と Amazon EC2 のどちらかを選択する方法を教えてください。
- TypeScript の変数を宣言する構文は何ですか?
- React でアプリを記述する方法を教えてください
- この [選択したコードまたはアプリケーション] が何をするのか、またどのように動作するのか説明してください
- [選択したコードまたは関数] のテストケースを生成して

Amazon Q からの応答に関する問題の報告

オプションとして、Amazon Q が生成するすべての応答に対して高評価アイコンまたは低評価アイコンでフィードバックを残すことができます。応答に関する問題を報告するには、低評価アイコンをクリックして表示されるフィードバックウィンドウに情報を入力します。

Amazon Q Developer でのコードの説明と更新

Amazon Q Developer は、統合開発環境 (IDE) で特定のコード行を説明および更新できます。コードを更新するには、Amazon Q にコードの特定の行またはブロックを変更するようにリクエストします。これにより、リクエストした変更を反映する新しいコードが生成されます。その後、更新したコードを、コードが作成されたファイルに直接挿入できます。

次のオプションから選択できます。

- 説明 – コードを自然言語で説明してもらいます。
- リファクタリング – コードの可読性や効率性の向上など、さまざまな改善を行います。
- 修正 – コードをデバッグします。
- テストの生成 – 現在のファイルまたは選択したコードのユニットテストを作成します。
- 最適化 – コードのパフォーマンスを向上させます。
- プロンプトに送信 – 強調表示されたコードを Amazon Q チャットパネルに送信し、コードについて質問します。

Amazon Q にコードを送信

Amazon Q にコードを説明または更新してもらうには、以下のステップを実行します。

1. IDE 内のコードファイルのセクションを強調表示します。
2. 強調表示されたコードを右クリックして、コンテキストウィンドウを開きます。Amazon Q を選択し、説明、リファクタリング、修正、テストの生成、最適化、またはプロンプトへの送信を選択します。

   [プロンプトに送信] を選択すると、Amazon Q は強調表示されたテキストをチャットパネルにコピーします。チャットパネルには、コードに関する質問を入力できます。
3. 強調表示されたコードを新しく生成されたコードに置き換えるには、コードをコピーするか、[コードを挿入] を選択してファイルに直接挿入します。Amazon Q は元のコードを更新されたコードに置き換えます。

Amazon Q Developer とのインラインチャット

インラインチャット機能を使用すると、IDE のメインコーディングウィンドウから Amazon Q とチャットできます。インラインチャット機能を使用するには、提案するコードを強調表示し、小さな

入力画面で手順を指定します。Amazon Q は、メインコーディングウィンドウ内の差分に表示されるコードの生成に進みます。その後、変更を受け入れるか拒否するかを選択できます。

インラインチャットの利点は、チャットウィンドウとメインコーディングウィンドウ間を移動するときに発生するコンテキストの切り替えを排除できることです。

通常、インラインチャット機能は、コードの確認、ユニットテストの記述、またはコードベースの回答を必要とするその他のタスクの実行時に使用します。テキストベースの回答 (「このコードを説明する」への回答など) が必要な場合は、チャットウィンドウを使用することをお勧めします。

Amazon Q は、インラインチャットを通じてコードレコメンデーションを生成するときに、現在のファイルのコードを考慮します。他のファイルやプロジェクトのコードは表示されません。

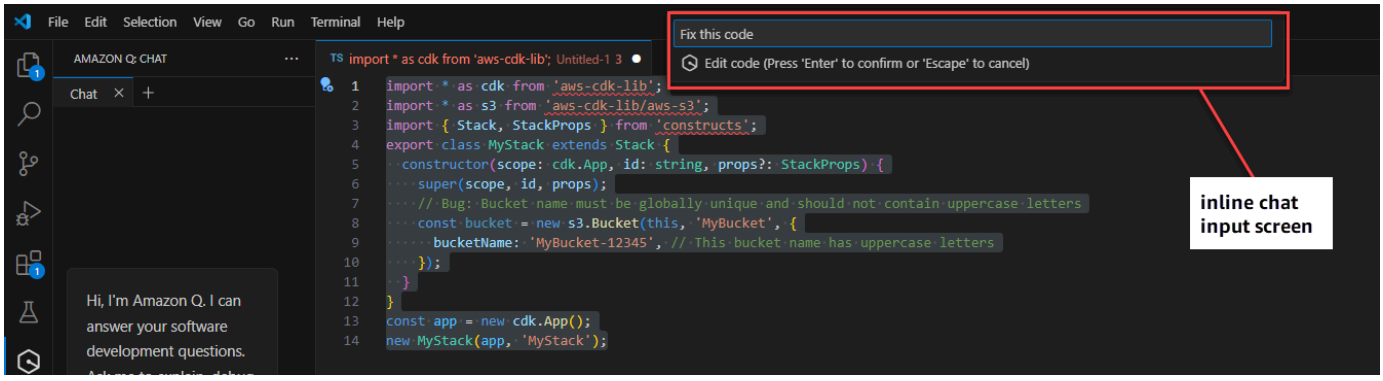
実行中の Amazon Q インラインチャット

インラインチャットセッションは次のように展開されます。

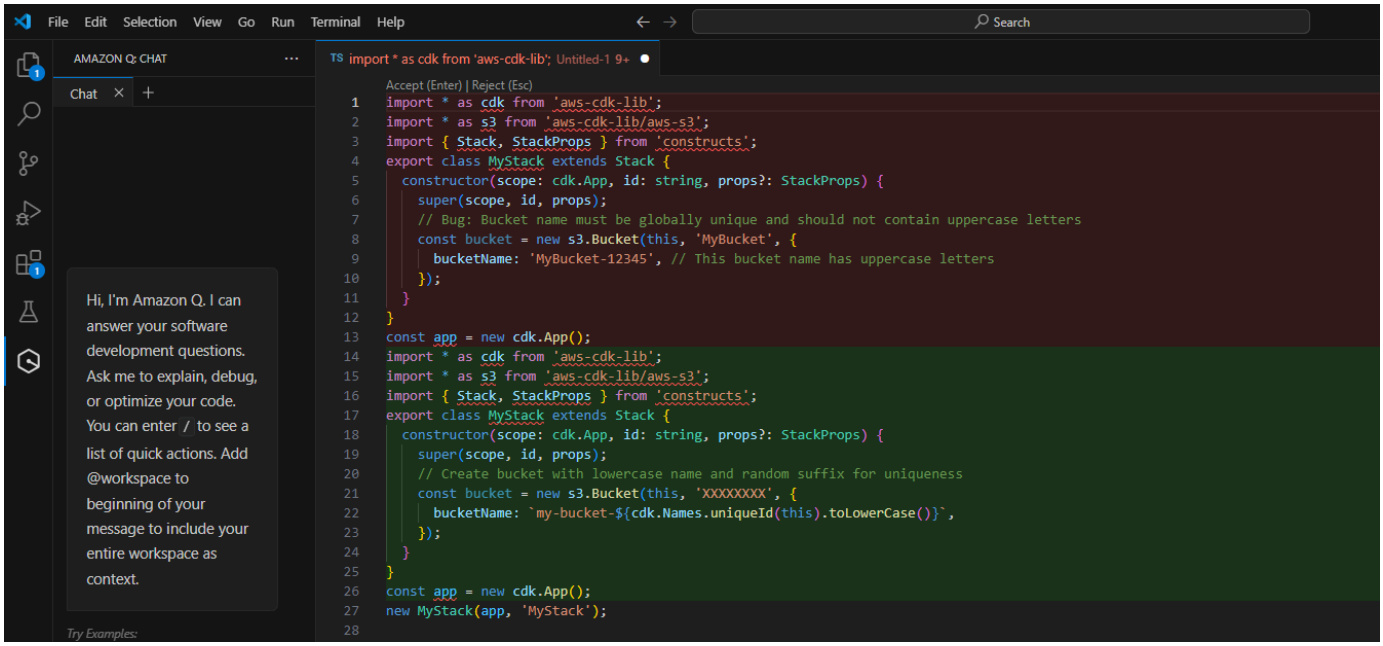
1. 提案するコードを強調表示し、IDE に基づいて次のオプションから選択します。

   - Visual Studio Code と でJetBrains、 #+I (Mac) または Ctrl+I (Windows) を押します。
   - でEclipse、 #+Shift+I (Mac) または Ctrl+Shift+I (Windows) を押します。
   - または、選択を右クリックして Amazon Q を選択し、インラインチャットを選択することもできます。

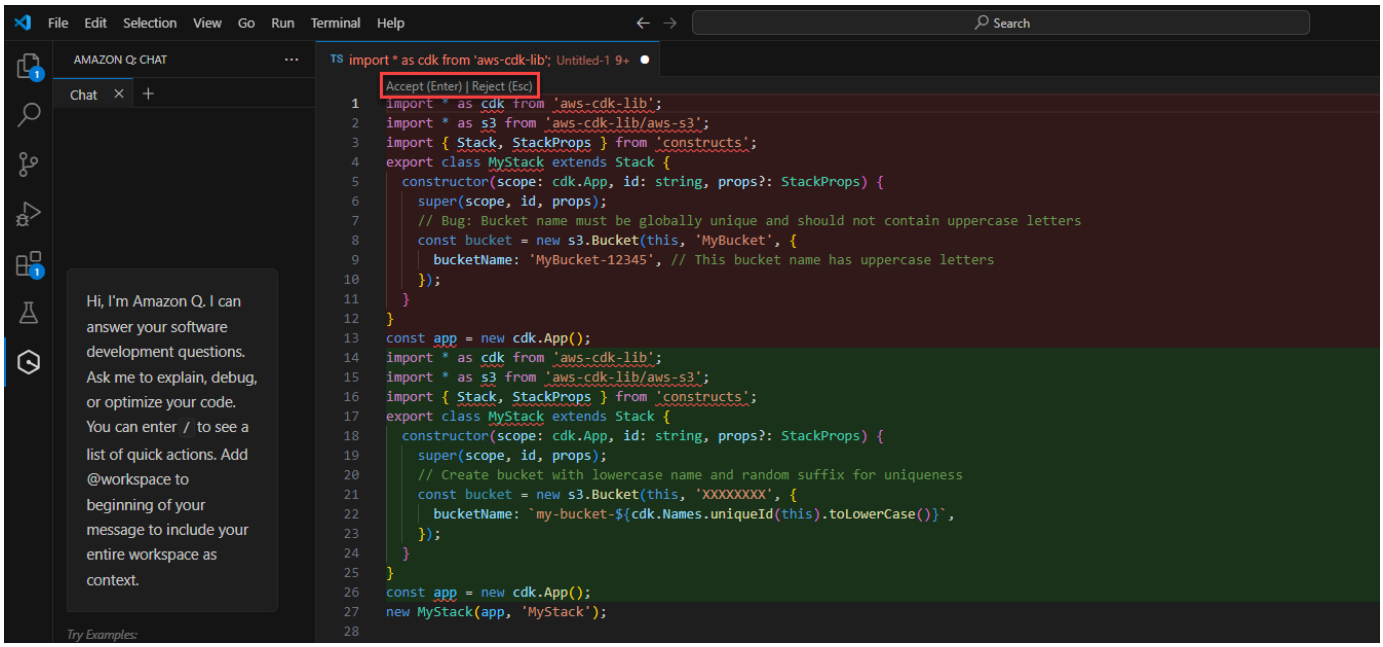
   これにより、メインコーディングウィンドウの上部に小さな入力画面が起動し、 などのプロンプトを入力できます**Fix this code**。



2. Amazon Q はコードを生成し、差分で表示します。



3. 変更を承諾または拒否するには、承諾または拒否を選択するか、キーボードの同等のもの (Enter または) を押しますEsc。



トピックおよび質問の例

インラインチャットは常にコードを回答として返すため、次のようなプロンプトを入力できます。

- このコードを文書化する
- このコードのリファクタリング

- この関数のユニットテストの書き込み

差分形式

インラインチャットでは、差分が複数のブロックで表示され、上部に既存のコード、下部に推奨コードが表示されます。side-by-side差分はサポートされていません。

IDE での Amazon Q Developer チャットへのコンテキストの追加

統合開発環境 (IDE) で Amazon Q とチャットするときは、Amazon Q が回答を調整して改善するために使用できるファイルやフォルダなどの追加のコンテキストを Amazon Q に提供できます。

Amazon Q にコンテキストを提供するには、次の 2 つの方法があります。

- 明示的 – コンテキストを明示的に指定するには、チャットウィンドウ**@**に と入力します。は、コンテキストとして含める項目を選択するコンテキストピッカーポップアップ**@**を起動します。または、「」と入力**@**して、ファイル、フォルダ、またはその他のコンテキストタイプの名前を入力して自動入力を開始することもできます。詳細については、「明示的なコンテキストタイプ」を参照してください。
- 自動 – コンテキストを自動的に提供するには、チャットの外部でコンテキストを個別にセットアップします。Amazon Q は、プロジェクトに取り組む開発者がチャットウィンドウに質問を入力するたびに、コンテキストを自動的に参照します。詳細については、「自動コンテキストタイプ」を参照してください。

Amazon Q が回答を生成すると、コンテキストドロップダウンリストでコンテキストとして使用したファイルが表示されます。これは、回答の先頭のすぐ上に表示されます。

明示的なコンテキストタイプ

チャット**@**に入力すると、次のコンテキストタイプから選択できます。

- @workspace – Amazon Q は、プロジェクトのワークスペースを回答のコンテキストとして使用します。@workspace オプションには設定が必要です。詳細については、「IDE の Amazon Q Developer チャットへのワークスペースコンテキストの追加」を参照してください。
- フォルダ – Amazon Q は、現在のプロジェクト内のフォルダのリストを表示し、選択したフォルダを回答のコンテキストとして使用します。
- ファイル – Amazon Q は、現在のプロジェクト内のファイルのリストを表示し、選択したファイルを回答のコンテキストとして使用します。

- コード – Amazon Q は、現在のプロジェクトのクラス、関数、グローバル変数のリストを表示し、選択内容を回答のコンテキストとして使用します。
- イメージ – Amazon Q では、イメージをプロンプトのコンテキストとして追加できます。UI モックアップやシーケンス図からコードを生成するなどのシナリオに役立ちます。イメージは JPEG、PNG、GIF、または WebP 形式で、最大サイズは 3.75 MB で、ディメンションは 8,000 x 8,000 ピクセルを超えない必要があります。コンテキストに固定されたイメージを含め、1 つのメッセージに最大 20 個のイメージを含めることができます。
- プロンプト – Amazon Q は、保存したプロンプトのリストを表示し、選択したプロンプトを回答のコンテキストとして使用します。プロンプトオプションには、何らかの設定が必要です。詳細については、「Amazon Q Developer チャットで使用するプロンプトをライブラリに保存する」を参照してください。

自動コンテキストタイプ

以下のタイプのコンテキストは、設定した場合、Amazon Q によって自動的に使用されます。

- プロジェクトルール – Amazon Q は、コンテキストとして定義した一連のプロジェクトルールを自動的に使用します。詳細については、「Amazon Q Developer チャットで使用するプロジェクトルールの作成」を参照してください。
- カスタマイズ – Amazon Q は、ソースコードのリポジトリをコンテキストとして自動的に使用します。カスタマイズの設定の詳細については、「」を参照してください提案のカスタマイズ。

IDE の Amazon Q Developer チャットへのワークスペースコンテキストの追加

統合開発環境 (IDE) で Amazon Q とチャットする場合、**`@workspace`** を質問に追加して、ワークスペースコードの最も関連性の高いチャンクをコンテキストとして自動的に含めることができます。Amazon Q Developer は、定期的に更新されるインデックスに基づいて関連性を判断します。

ワークスペースコンテキストにより、Amazon Q はファイルの検索、複数のファイル間でのコードの使用方法の理解、未開封のファイルを含む複数のファイルを活用するコードの生成などの機能を強化しました。

トピック

- セットアップ
- ワークスペースコンテキストを使用して質問する

セットアップ

続行する前に、最新バージョンの IDE がインストールされていることを確認してください。確認したら、次のステップを実行してセットアップを行います。

インデックス作成を有効にする

ワークスペースをコンテキストとして使用するために、Amazon Q は、コードファイル、設定ファイル、プロジェクト構造など、ワークスペースリポジトリのローカルインデックスを作成します。インデックス作成中、バイナリや、`.gitignore` ファイルで指定されたような不要なファイルは除外されます。

新しいワークスペースのインデックス作成には 5～20 分かかる場合があります。この間、IDE で CPU 使用率が高くなることが予想されます。最初のインデックス作成後、ワークスペースに変更を加えると、インデックスが変更分だけ更新されます。

ワークスペースコンテキストを初めて追加する際、IDE でインデックス作成を有効にする必要があります。それ以外の場合は、以下のステップを完了して、インデックス作成を有効にします。

1. Amazon Q チャットパネルの質問に **`@workspace`** を追加します。
2. Amazon Q により、インデックス作成を有効にするように求められます。[設定] を選択すると、IDE の Amazon Q 設定にリダイレクトされます。

   プロンプトが表示されない場合は、IDE の下部にある Amazon Q を選択して設定に移動できます。次に、開いた Amazon Q タスクバーから Open Settings を選択します。
3. Workspace Index の横にあるボックスを選択します。

インデックス作成を設定する (オプション)

インデックス作成プロセスには設定は必要ありませんが、インデックス作成専用のスレッド数を指定できます。使用するスレッドの数を増やすと、インデックス作成の完了が速くなり、CPU の使用量が増えます。インデックス作成設定を更新するには、Workspace Index ワーカースレッド設定のスレッド数を指定します。ワークスペースコンテキストに対してインデックス作成できるファイルの最大サイズを設定し、インデックス作成にグラフィック処理ユニット (GPU) を使用できるようにすることもできます。

ワークスペースコンテキストを使用して質問する

Amazon Q との会話にコンテキストとしてワークスペースを追加するには、質問対象のワークスペースを開き、チャットパネルで質問に **@workspace** を追加します。ワークスペースコンテキストを追加する質問には必ず @workspace を追加する必要があります。

別のワークスペースに関するチャットを開始する場合は、そのワークスペースを開き、新しいチャットタブを開きます。質問に **@workspace** を含め、新しいワークスペースをコンテキストとして追加します。

ワークスペース内の任意のファイルについて Amazon Q に質問できます。未開封のファイルも含まれます。Amazon Q は、既存の会話ベースのコーディング機能に加えて、ファイルの説明、コードの検索、ファイル間のコードの生成を行うことができます。

以下は、ワークスペースコンテキストを活用する Amazon Q へのチャットでの質問の例です。

- @workspace 認可を処理するコードはどこにありますか?
- @workspace このプロジェクトでアプリケーションロジックを含んでいる主なクラスは何ですか?
- @workspace main.py を説明して
- @workspace このプロジェクトに認証を追加して
- @workspace このプロジェクトでは、どのようなサードパーティーライブラリまたはパッケージが使用され、どのような目的で使用されていますか?
- @workspace 関数 ##### のユニットテストを追加して

Amazon Q Developer チャットで使用するプロンプトをライブラリに保存する

IDE で Amazon Q とチャットするときに使用できる一般的なプロンプトのライブラリを構築できます。これらのプロンプトをライブラリに保存することで、プロンプトを毎回再入力することなく、チャットに簡単に挿入できます。保存されたプロンプトは、複数の会話やプロジェクトで使用できます。

プロンプトは `~/.aws/amazonq/prompts`フォルダに保存されます。

プロンプトライブラリにプロンプトを保存するには

1. IDE で、Amazon Q チャットウィンドウを開きます。
2. と入力し**@**、プロンプトを選択します。

3. 「新しいプロンプトを作成する」を選択します。
4. プロンプト名に、 などのプロンプト名を入力し**Create sequence diagram**、Enter キーを押します。

   Amazon Q は、 `~/.aws/amazonq/prompts`フォルダ`Create sequence diagram.md`に というプロンプトファイルを作成し、IDE でファイルを開きます。
5. プロンプトファイルに、詳細なプロンプトを追加します。例:

   ```
   Create a sequence diagram using Mermaid that shows the sequence of
   calls between resources. Ignore supporting resources like IAM policies
   and security group rules.
   ```
6. プロンプトファイルを保存します。

保存されたプロンプトを使用するには

1. IDE で、Amazon Q チャットウィンドウを開きます。
2. と入力し**@**、プロンプトを選択します。
3. 保存したプロンプトを選択します。たとえば、シーケンス図を作成します。
4. (オプション) チャット入力ウィンドウで、必要に応じて詳細を追加します。さらにテキストを入力し、コンテキストタイプを追加できます。プロンプトの例は次のようになります。

   @Create a sequence diagram `using the files in the` **@lib** `folder`
5. プロンプトを送信し、Amazon Q が回答を生成するのを待ちます。

コンテキスト項目のピン留め

> **Note**
>
> コンテキストピン留めは現在、VS Code IDE でのみ使用できます。

コンテキストピン留めを使用すると、チャットセッション内のすべてのメッセージに追加されるコンテキスト項目を指定できます。コンテキスト項目を固定すると、現在の会話内のすべてのメッセージに自動的に含まれるため、**@workspace**、、 **@file**などのコマンドを繰り返し入力する必要がなくなります**@folder**。

ピン留めされた項目は 2 つのソースから取得できます。頻繁に参照する項目を手動でピン留めすることも、Amazon Q がレスポンスの品質を向上させるためにコンテキスト (現在のアクティブなファイルなど) を自動的に追加することもできます。ピン留めされたコンテキスト項目はチャットパネルのテキスト入力ボックスの上部に表示され、含めたくないコンテキストは削除できます。

明確なコンテキスト境界を維持するために、ピン留めされた項目は現在のチャットタブにのみ適用されます。新しいタブを開くと、アクティブなファイルなど、ピン留めされたデフォルトのコンテキストのみで新しく開始されます。

ピン留めされたコンテキストの使用

ピン留めされたコンテキスト項目を追加するには

1. IDE で、Amazon Q チャットパネルを開きます。
2. チャット**@prompt**で **@workspace**、**@file**、**@folder**、 などのコンテキストコマンドを使用したら、目的のコンテキストをクリックして固定します。

   または、「@ Pin Context」ボタンをクリックして使用可能なオプションを表示し、ピン留めするコンテキストを選択できます。
3. ピン留めされたコンテキストは、チャットパネルの上部にあるピン留めされたコンテキスト領域に表示されます。

コンテキスト項目を固定する方法

コンテキスト項目を固定するには、次の 3 つの方法があります。

1. @Pin コンテキストメニューの使用：
   - チャットパネルの「@Pin Context」ボタンをクリックします。
   - 使用可能なオプションから目的のコンテキスト項目を選択します。
2. コンテキストメニューとキーボードショートカットの使用：
   - チャット入力に「@」と入力して、コンテキストメニューを表示します。
   - 目的の項目に移動します。
   - Option/Alt + Enter を押して、選択した項目を固定します。
3. 入力プロンプトからのピン留め：
   - 入力にコンテキストコマンド (**@workspace**、、**@folder**、 など**@prompt**) **@file**を既に入力している場合は、入力のコンテキスト項目にカーソルを合わせます。
   - 項目をクリックして固定します。

ピン留め後、コンテキスト項目はチャットのテキスト入力ボックスの上部にあるピン留めされたコンテキスト領域に表示されます。

ピン留めされたコンテキスト項目を削除するには

- ピン留めされたコンテキスト項目を削除するには、ピルの左側にある X をクリックします。これは、ユーザーピン留めコンテキスト項目とシステム追加コンテキスト項目の両方で機能します。

Amazon Q Developer チャットで使用するプロジェクトルールの作成

IDE で Amazon Q とチャットするときに使用できるプロジェクトルールのライブラリを構築できます。これらのルールでは、チーム全体のコーディング標準とベストプラクティスについて説明します。たとえば、すべての Python コードがタイプヒントを使用する必要があること、またはすべての Java コードが Javadoc コメントを使用する必要があることを示すルールがあるとします。これらのルールをプロジェクトに保存することで、経験レベルに関係なく、開発者間の一貫性を確保できます。

プロジェクトルールは、プロジェクトの *project-root*`/.amazonq/rules`フォルダの Markdown ファイルで定義されます。

プロジェクトルールを作成すると、Amazon Q はデベロッパーがプロジェクト内で Amazon Q とチャットするたびに自動的にコンテキストとして使用し、回答を生成するときにそれらに従うようにします。チャットにコンテキストを追加する方法の詳細については、「」を参照してくださいIDE での Amazon Q Developer チャットへのコンテキストの追加。

プロジェクトルールは、ファイルシステムまたは Amazon Q チャットインターフェイスから直接作成できます。

Amazon Q チャットインターフェイスを使用してプロジェクトルールを作成するには

1. IDE で、Amazon Q チャットパネルを開きます。
2. チャット入力ボックスで、ルールボタンをクリックします。
3. 新しいルールの作成 を選択します。
4. 表示されるダイアログで、ルールの名前を入力します。

   これにより、プロジェクトの *project-root*`/.amazonq/rules`フォルダにその名前の Markdown ファイルが作成されます。

5. エディタにルールコンテンツを追加します。
6. ファイルを保存します。

ファイルシステムを使用してプロジェクトルールを作成するには

1. IDE で、プロジェクトのルートフォルダを開きます。
2. プロジェクトルートフォルダで、次のフォルダを作成します。

   project-root/.amazonq/rules

   このフォルダには、すべてのプロジェクトルールが格納されます。
3. で*project-root*/.amazonq/rules、プロジェクトルールファイルを作成します。Markdown ファイルである必要があります。例:

   cdk-rules.md
4. プロジェクトルールの Markdown ファイルを開きます。
5. ファイルに詳細なプロンプトを追加します。例:

```
All Amazon S3 buckets must have encryption enabled, enforce SSL, and block public
 access.
All Amazon DynamoDB Streams tables must have encryption enabled.
All Amazon SNS topics must have encryption enabled and enforce SSL.
All Amazon SNS queues must enforce SSL.
```

6. ファイルを保存します。
7. (オプション) プロジェクトルールのマークダウンファイルを追加します。

   これで、1 つ以上のプロジェクトルールが作成されました。Amazon Q は、開発者がプロジェクト内で Amazon Q とチャットするたびに、これらのルールを自動的にコンテキストとして使用します。

Amazon Q チャットインターフェイスでルールを管理するには

1. IDE で、Amazon Q チャットパネルを開きます。
2. チャット入力ボックスで、ルールボタンをクリックして、使用可能なすべてのルールを表示します。
3. ルールをクリックして、現在のチャットセッションのオンとオフを切り替えます。

- チェックマークが付いたルールはアクティブであり、会話に適用されます。
- チェックマークのないルールは、現在のセッションでは非アクティブです。

Amazon Q Developer でのチャット履歴の圧縮

IDE で Amazon Q Developer を操作すると、会話がチャット履歴に蓄積されます。この履歴は、Amazon Q がプロジェクトを理解し、より関連性の高いレスポンスを提供するのに役立つ重要なコンテキストを提供します。ただし、基盤となるモデルへの各リクエストに含めることができる会話履歴の量には制限があります。

コンテキストウィンドウの制限について

コンテキストウィンドウは、Amazon Q との 1 回のやり取りで処理できる情報の最大量を表します。これには、以下が含まれます。

- 現在の質問またはリクエスト
- 会話の以前のメッセージ
- 共有したコードスニペットとファイル
- プロジェクトに関するシステム情報

このコンテキストウィンドウが容量制限に近づくと、Amazon Q が会話の以前の部分を参照する能力に影響する可能性があります。

チャット履歴圧縮の仕組み

チャット履歴圧縮を使用すると、使用するコンテキストの量を減らしながら、会話の重要な情報を保持できます。圧縮が発生した場合：

1. Amazon Q が会話履歴を分析する
2. 重要なポイント、質問、決定事項の簡潔な概要を作成します。
3. この概要は、コンテキストウィンドウの詳細な会話履歴を置き換えます。
4. 完全な会話はチャットインターフェイスに表示されたままになります。

圧縮すると、重要なコンテキストを失うことなく会話を続けることができ、コンテキストウィンドウの制限に達したときにまったく新しいチャットを開始する必要がなくなります。

チャット履歴圧縮の使用

圧縮は 2 つの方法で使用できます。

手動圧縮

チャット履歴を手動で圧縮するには：

1. チャット入力フィールドに**/compact**「」と入力します。
2. Amazon Q はリクエストを処理し、圧縮された会話の概要を含む確認メッセージを表示します。

現在の会話を続けるが、応答時間が遅い、または関連性の低い回答に気付く場合は、手動圧縮を使用します。

自動圧縮ナッジ

コンテキストウィンドウが容量の約 80% に達すると、Amazon Q は圧縮を示す通知を表示します。この通知には以下が含まれます。

- 圧縮が推奨される理由の説明
- 圧縮をすぐにトリガーするボタン

圧縮後

圧縮後：

- 完全な会話履歴は、現在のセッションが終了するまでチャットインターフェイスに表示されます。
- Amazon Q は、圧縮された概要 (完全な履歴ではない) を使用してレスポンスを生成します。
- 圧縮された概要は、詳細な履歴ではなくコンテキストウィンドウに含まれます。
- IDE を再起動すると、詳細なチャット履歴がリセットされます。

関連コマンド

チャット履歴のクリア

圧縮の代わりに、 **/clear** コマンドを使用してチャット履歴を完全にクリアできます。

1. チャット入力フィールドに **/clear**と入力します。

2. Amazon Q は、表示ウィンドウとコンテキストウィンドウの両方から以前の会話履歴をすべて削除します。

圧縮履歴とクリア履歴を選択するタイミング

次の場合に圧縮を選択します。

- 現在の会話トピックを続行したい
- 以前のコンテキストは、現在のタスクに引き続き関連します。
- 会話の一般的な方向性と知識を維持したい

次の場合に履歴をクリアするには、 を選択します。

- まったく新しいタスクまたはトピックを開始する場合
- 以前の会話は関連しなくなりました
- 以前のコンテキストが新しいレスポンスに影響を与えないようにしたい
- 会話から機密情報を削除する場合

Amazon Q Developer の会話履歴の表示、削除、エクスポート

統合開発環境 (IDE) で Amazon Q とチャットすると、Amazon Q は各チャットタブを個別の会話として保存します。これらの会話を表示、検索、削除できます。それらを Markdown または HTML 形式のファイルにエクスポートすることもできます。

Amazon Q は、会話をローカルコンピュータのホームディレクトリに保存します。

Amazon Q はワークスペースごとに会話を個別に保存するので、会話履歴が表示されない場合は、ワークスペースが間違っている可能性があります。Amazon Q には、現在のワークスペースの会話履歴のみが表示されます。

次の手順を使用して、会話を表示、検索、削除、エクスポートします。

過去の会話を表示および検索するには

1. IDE で Amazon Q にサインインします。
2. Amazon Q チャットタブを開きます。
3. 次のいずれかを実行して、チャット履歴を開きます。

- チャットパネルの右上で、チャット履歴の表示ボタンを選択します。
- ctrl+F (Windows および Linux) または # F (Mac) を押します。

4. 次のいずれかを行います：

   - 表示する会話を選択します。会話は日付別に整理されます。
   - チャット履歴の上部にある検索バーを使用して会話を見つけます。Amazon Q は、入力したテキストと正確に一致する会話を見つけます。

1 つの会話を削除するには

1. IDE で Amazon Q にサインインします。
2. 次のいずれかを行います：

   - 開いているチャットセッションのチャットタブで、 と入力**/clear**してチャットタブの内容を削除します。
   - Amazon Q チャットタブを開き、次のいずれかを実行してチャット履歴を開きます。
     - チャットパネルの右上で、チャット履歴の表示ボタンを選択します。
     - ctrl+F (Windows および Linux) または # F (Mac) を押します。

     削除する会話で、縦の省略記号 (⋮) を選択し、削除を選択します。

会話を Markdown または HTML にエクスポートするには

1. IDE で Amazon Q にサインインします。
2. 次のいずれかを行います：

   - チャットセッションがすでに開始されている状態で、チャットパネルの右上でエクスポートボタンを選択して、タブに表示される会話をエクスポートします。
   - Amazon Q チャットタブを開き、次のいずれかを実行してチャット履歴を開きます。
     - チャットパネルの右上で、チャット履歴の表示ボタンを選択します。
     - ctrl+F (Windows および Linux) または # F (Mac) を押します。

     エクスポートする会話で、縦の省略記号 (⋮) を選択し、エクスポートを選択して会話を Markdown または HTML 形式ファイルにエクスポートします。

デフォルトでは、Amazon Q はファイルに名前を付けq-dev-chat-*yyyy*-*mm*-*dd*.*md*|*html*、プロジェクトのルートに保存します。

Amazon Q Developer とのチャットでのショートカットキーの使用

Amazon Q には、エージェントチャットインターフェイスを効率的に操作するためのキーボードショートカットが用意されています。キーボードショートカットを使用すると、キーボードとマウスを切り替えることなくワークフローを維持できます。

> **Note**
>
> キーボードショートカットは、現在 IDE Visual Studio Code でのみ使用できます。

キーボードショートカット

Amazon Q チャットのキーボードショートカット

アクション	Visual Studio Code ショートカット
コマンドを実行する	Shift + Cmd + Enter (Mac) Shift + Ctrl + Enter (Windows) Shift + Meta + Enter (Linux)
コマンドを拒否する	Shift + コマンド + R (Mac) Shift + Ctrl + R (Windows) Shift + Meta + R (Linux)
生成を停止する	Shift + コマンド + バックスペース (Mac) Shift + Ctrl + バックスペース (Windows) Shift + Meta + Backspace (Linux)

キーボードショートカットのカスタマイズ

ショートカットは、IDE の標準のキーボードショートカットカスタマイズインターフェイスを使用して変更できます。Amazon Q は各 IDE のネイティブショートカットシステムを使用するため、行った変更は Amazon Q インターフェイスに反映されます。ツールヒントと UI 要素に表示されるショートカットは、カスタマイズされたキーマッピングと一致するように自動的に更新されます。

ショートカット検出可能性

Amazon Q には、キーボードショートカットを検出するいくつかの方法があります。

- ツールヒント: コマンドの実行ボタン、コマンド拒否ボタン、または停止ボタンにカーソルを合わせると、現在のキーボードショートカットが表示されます。
- 設定: IDE のキーボードショートカットカスタマイズインターフェイスですべてのショートカットを表示する

ショートカットの動作

- エージェントチャットのキーボードショートカットは、焦点が Amazon Q チャットパネルにある場合にのみアクティブになります。
- これにより、既存の IDE キー割り当てとの競合を防ぐことができます。
- ショートカットを変更する場合は、変更をツールヒントに反映するために IDE を更新する必要がある場合があります。

IDEs での Amazon Q チャットのモデルの選択

IDE でのチャット中に Amazon Q で使用するモデルを選択できます。選択したモデルは、以降のすべてのチャットセッションで保持されます。

次の表は、IDE で Amazon Q チャットで使用できるモデルとそのコンテキストウィンドウを示しています。

モデル	コンテキストウィンドウ
Claude Sonnet 3.7	200,000
Claude Sonnet 4 (デフォルト)	200,000

IDE でチャットに使用するモデルを選択する

IDE でチャットするときに Amazon Q が使用するモデルを選択するには：

1. IDE で Amazon Q チャットパネルを開きます。
2. テキストボックスの下部で、モデルメニュードロップダウンを選択します。使用可能なオプションから、使用するモデルを選択します。

   選択したモデルは、変更するまで保持されます。

コンテキストウィンドウ

コンテキストウィンドウは、Amazon Q がリクエストを処理して応答するために使用できる、会話履歴や明示的または自動的なコンテキストを含むコンテキストの量です。コンテキストは、テキストとコードを含むトークンで測定されます。

コンテキストの詳細については、「」を参照してくださいチャットへのコンテキストの追加。

Amazon Q Developer によるインライン提案の生成

Amazon Q は、リアルタイムでコードを提案できます。コードの記述中、Amazon Q は既存のコードとコメントに基づいて、自動的に提案コードを生成します。パーソナライズされた推奨事項のサイズと範囲は、単一行のコードから、完全に形成された関数にいたるまで、多岐にわたります。

単一行のコードやコメントを入力し始めると、Amazon Q は現在と以前の入力に基づいて提案を行います。ファイル名も考慮されます。

Amazon Q 拡張機能をダウンロードすると、インライン提案が自動的に有効になります。開始するには、コードの作成を開始します。Amazon Q がコード提案の生成を開始します。

Amazon Q がソフトウェア開発チームの内部ライブラリ、独自のアルゴリズム手法、エンタープライズコードスタイルに対して生成する提案をカスタマイズすることもできます。提案のカスタマイズの詳細については、「提案のカスタマイズ」を参照してください。

トピック

- Amazon Q での提案の一時停止
- Amazon Q のコード補完の仕組み

- AWS コーディング環境でインライン提案を生成する
- ショートカットキーの使用
- コードリファレンスの使用
- コードの例

Amazon Q での提案の一時停止

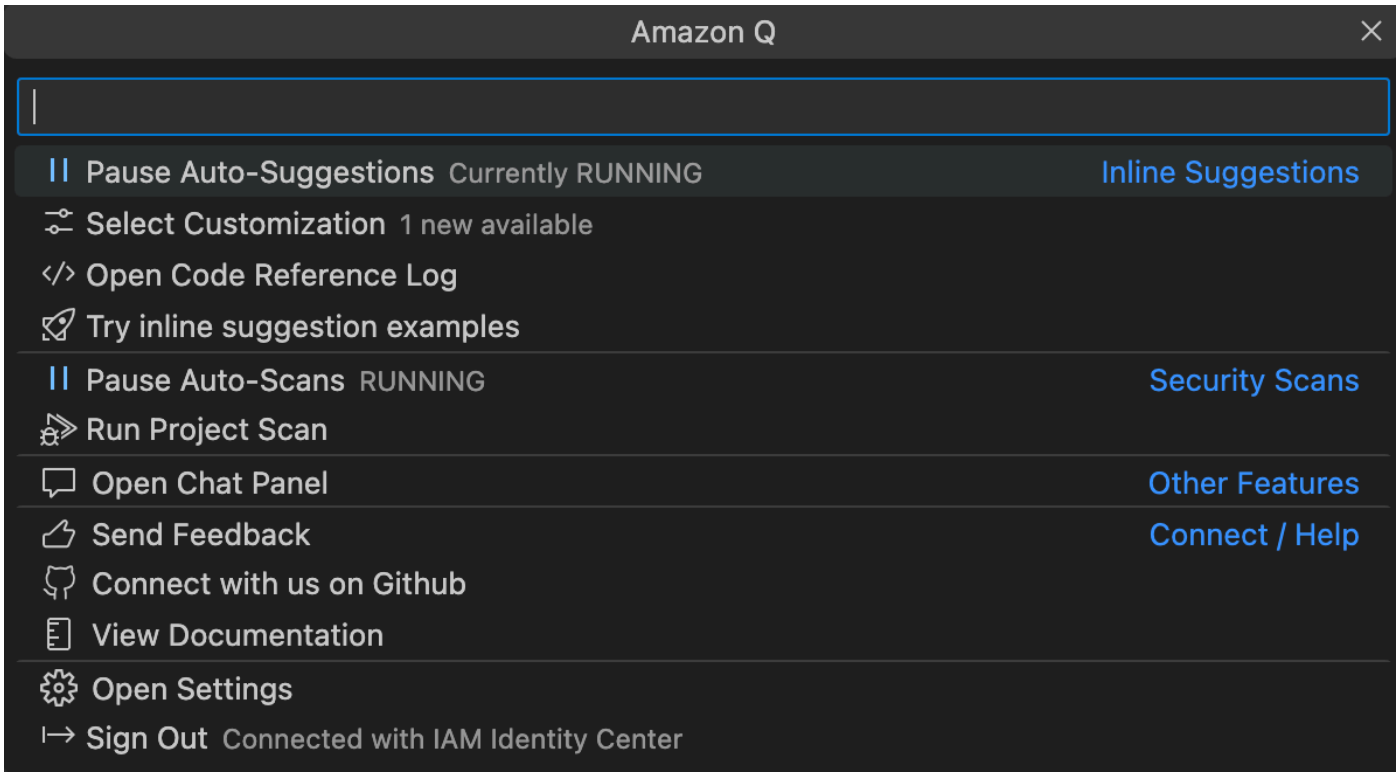
IDE を選択すると、Amazon Q でインラインコードの提案を一時停止および再開する手順を確認できます。

Visual Studio Code

1. VS Code で、IDE ウィンドウの下部にあるコンポーネントトレイから [Amazon Q] を選択します。

   Amazon Q タスクバーが IDE ウィンドウの上部に表示されます。
2. [自動提案の一時停止] か [自動提案の再開] を選択します。

次の図は、VS Code の Amazon Q タスクバーを示しています。

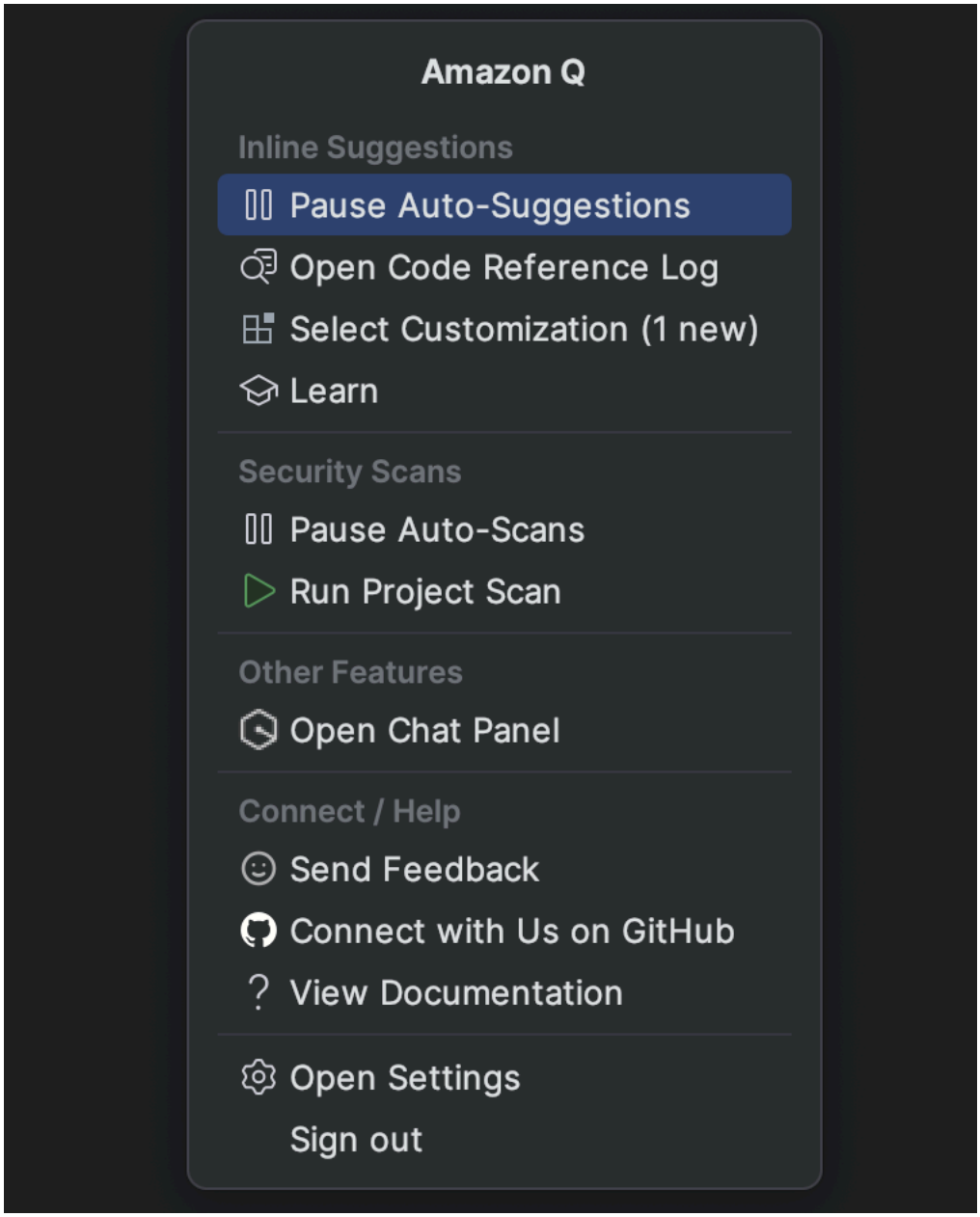


JetBrains

1. IDE で、IDE JetBrains ウィンドウの下部にあるステータスバーから Amazon Q を選択します。

   Amazon Q タスクバーがステータスバーの上に開きます。
2. [自動提案の一時停止] か [自動提案の再開] を選択します。

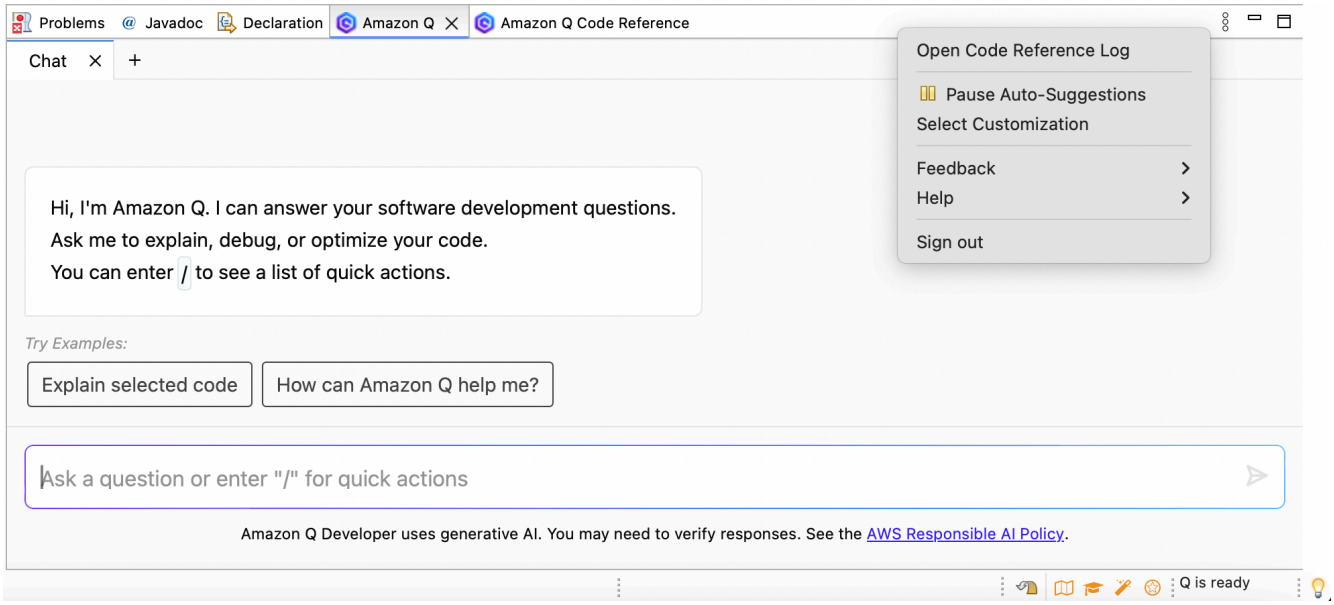
次の図は、IDE の Amazon Q JetBrains タスクバーを示しています。



Eclipse

1. IDE で、IDE Eclipse の右上隅にある Amazon Q アイコンを選択します。
2. Amazon Q チャットタブを開いた状態で、タブの右上隅にある省略記号アイコンを選択します。Amazon Q タスクバーが開きます。

   次の図は、IDE の Amazon Q Eclipse タスクバーを示しています。

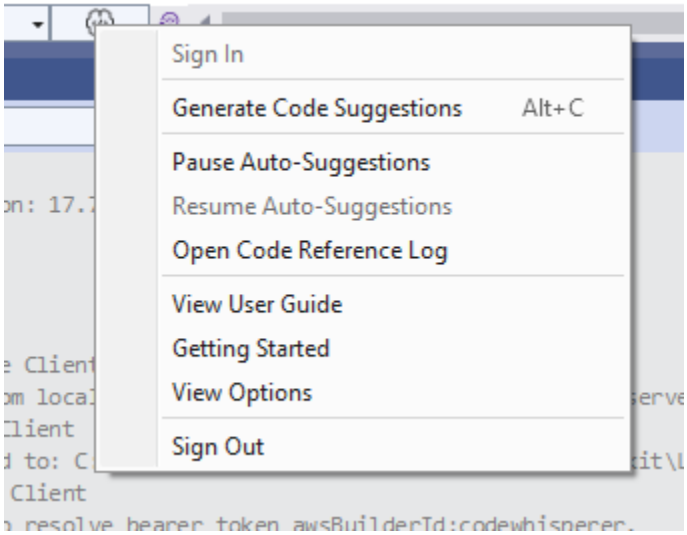


3. [自動提案の一時停止] か [自動提案の再開] を選択します。

Visual Studio

1. ウィンドウの端で Amazon Q アイコンをクリックします。
2. [自動提案の一時停止] か [自動提案の再開] を選択します。

次の図は、Visual Studio の Amazon Q タスクバーを示しています。



AWS Cloud9

Amazon Q は、 AWS Cloud9での提案のオンとオフの切り替えをサポートしていません。

Amazon Q の提案の受信を停止するには AWS Cloud9、アクセスに使用しているロールまたはユーザー AWS Cloud9 から へのアクセスを Amazon Q に許可する IAM ポリシーを削除します AWS Cloud9。

AWS Lambda

Lambda で Amazon Q のコード提案を非アクティブ化または再アクティブ化するには:

1. Lambda コンソールで、特定の Lambda 関数の画面を開きます。
2. [コードソース] セクションのツールバーから [ツール] を選択します。
3. ドロップダウンメニューから、[Amazon Q のコード提案] を選択します。

Amazon SageMaker AI Studio

1. SageMaker AI Studio コンソールで、ウィンドウの下部から Amazon Q を選択します。

   [Amazon Q] パネルが開きます。
2. [自動提案の一時停止] か [自動提案の再開] を選択します。

JupyterLab

1. JupyterLab コンソールで、ウィンドウの下部で Amazon Q を選択します。

[Amazon Q] パネルが開きます。

2. [自動提案の一時停止] か [自動提案の再開] を選択します。

AWS Glue Studio Notebook

1. AWS Glue Studio Notebook コンソールで、ウィンドウの下部から Amazon Q を選択します。

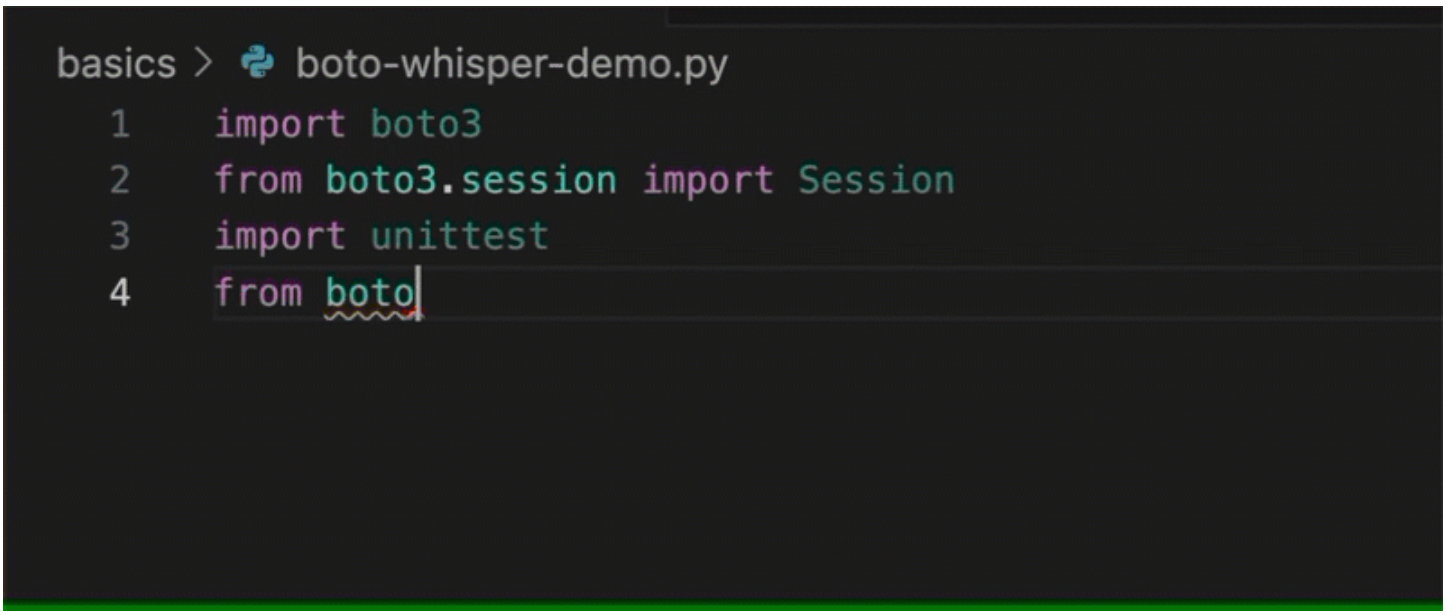
   [Amazon Q] パネルが開きます。

2. [自動提案の一時停止] か [自動提案の再開] を選択します。

Amazon Q のコード補完の仕組み

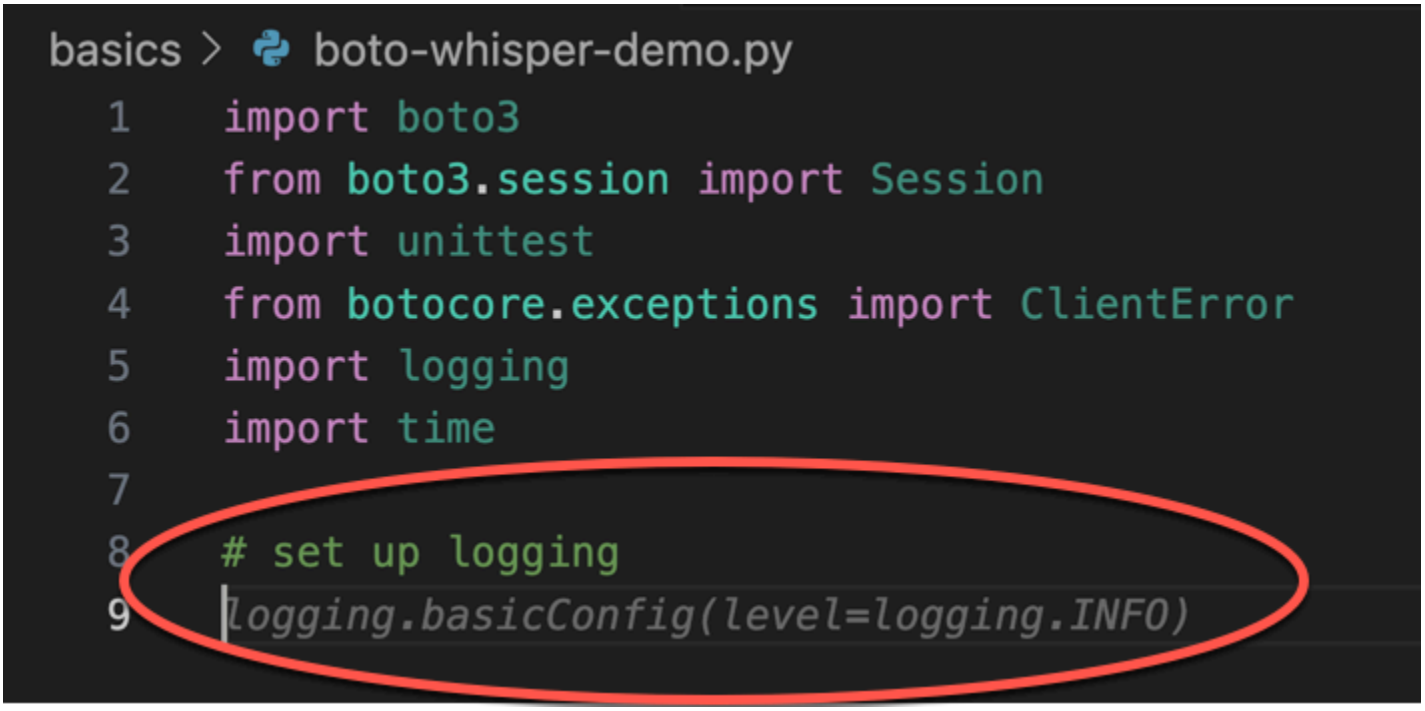
このセクションでは、Amazon Q が完全なアプリケーションの作成にどのように役立つかを実際に見ていきます。このアプリケーションは、Amazon S3 バケットと Amazon DynamoDB テーブルに加えて、両方のタスクを検証するユニットテストを作成します。

ここでは、Amazon Q は、インポートするライブラリの選択を手助けします。矢印キーを使用して、デベロッパーは複数の提案を切り替えます。



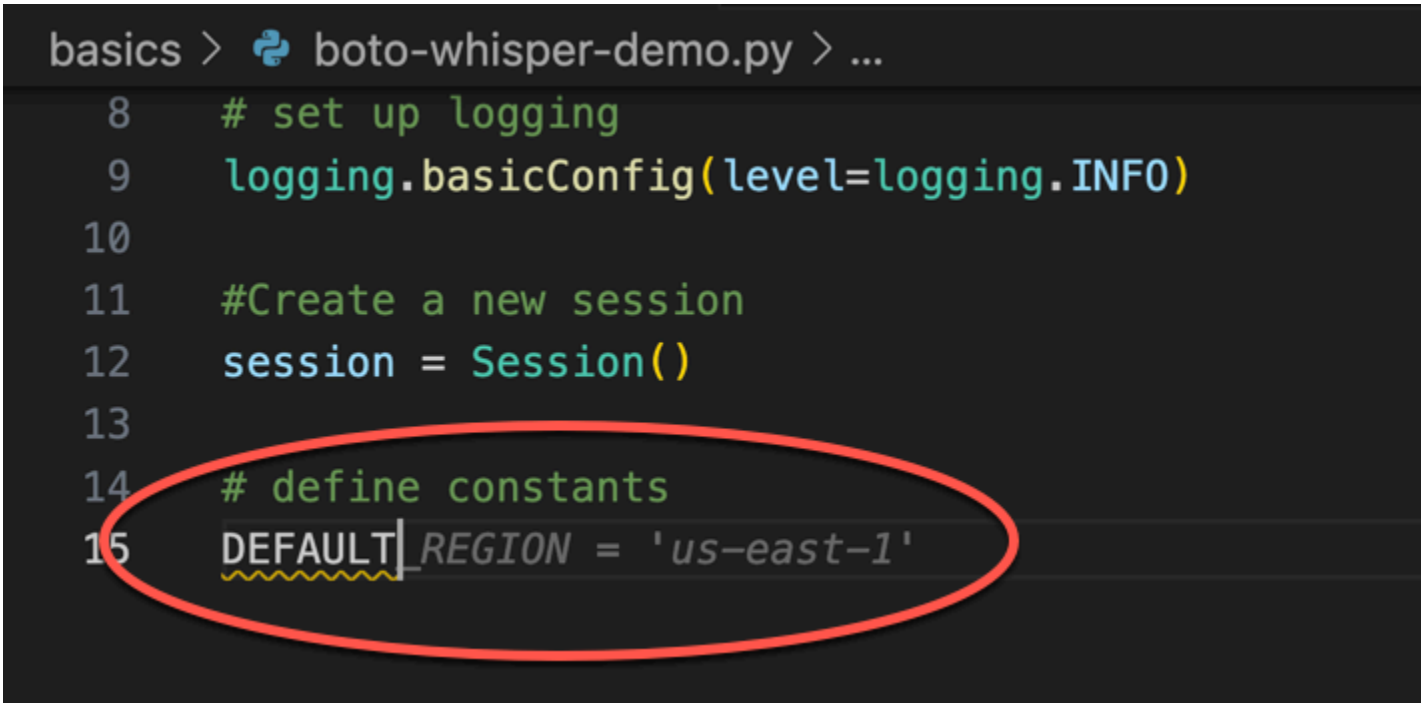
ここでは、デベロッパーが次の行に書き込むコードについて説明するコメントを入力します。

Amazon Q は、呼び出されるメソッドを正しく予測します。デベロッパーは、タブキーを使用して提案を受け入れることができます。



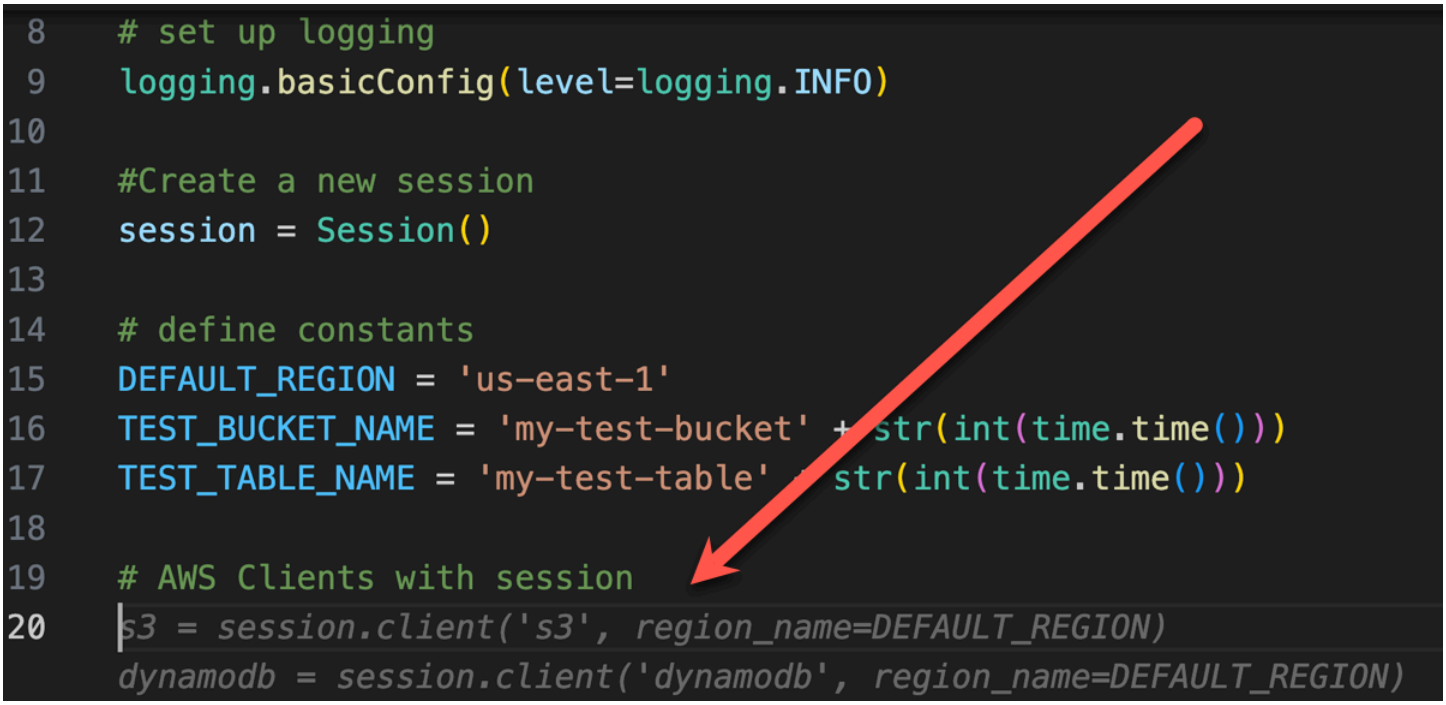
ここでは、デベロッパーは定数を定義する準備をします。

Amazon Q は、最初の定数が REGION になり、その値がデフォルトである us-east-1 になると正しく予測します。



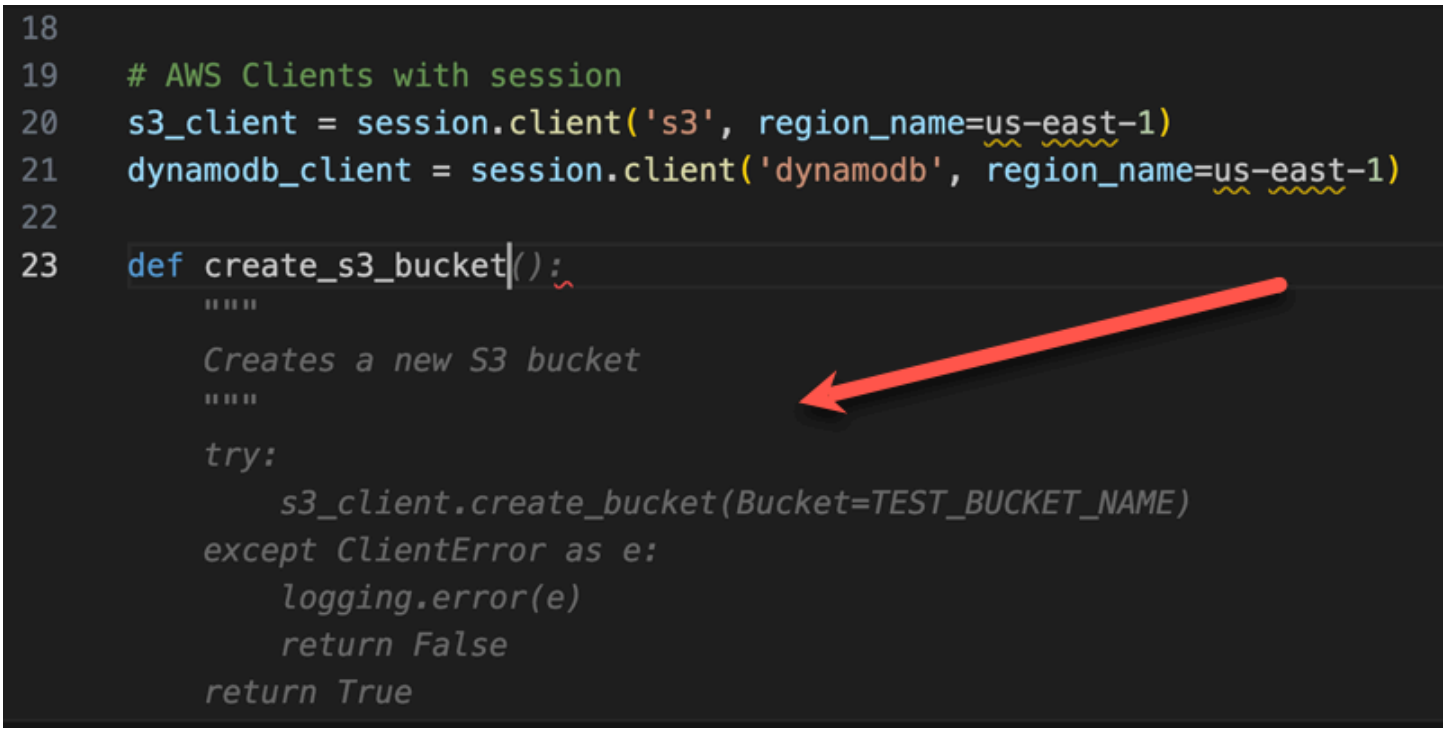
ここでは、デベロッパーは、ユーザーと Amazon S3 と DynamoDB の両方の間でセッションを開くコードを作成する準備をします。

Amazon Q は、 AWS APIsと SDKs、正しい形式を提案します。



デベロッパーは、バケットを作成する関数の名前を入力しただけです。ただし、その情報 (およびコンテキスト) に基づいて、Amazon Q は try/except 句を含む完全な関数を提供します。

TEST_BUCKET_NAME, which is a constant declared earlier in the same file. の使用に注目してください。これは、同じファイルで先に宣言された定数です。



デベロッパーは、DynamoDB テーブルを作成する関数の名前を入力し始めたばかりです。しかし、Amazon Q はこれがどこに向かっているかがわかっています。

提案は、先に作成した DynamoDB セッションを考慮しており、コメントでもそれに言及しています。

デベロッパーがユニットテストクラスの名前を記述するだけで、Amazon Q によりユニットテストクラスの補完が提案されました。

同じファイルで前に作成した 2 つの関数への組み込みリファレンスに注目してください。

デベロッパーは、DynamoDB テーブルを作成する関数の名前を入力し始めたばかりです。しかし、Amazon Q はこれがどこに向かっているかがわかっています。

提案は、先に作成した DynamoDB セッションを考慮しており、コメントでもそれに言及しています。

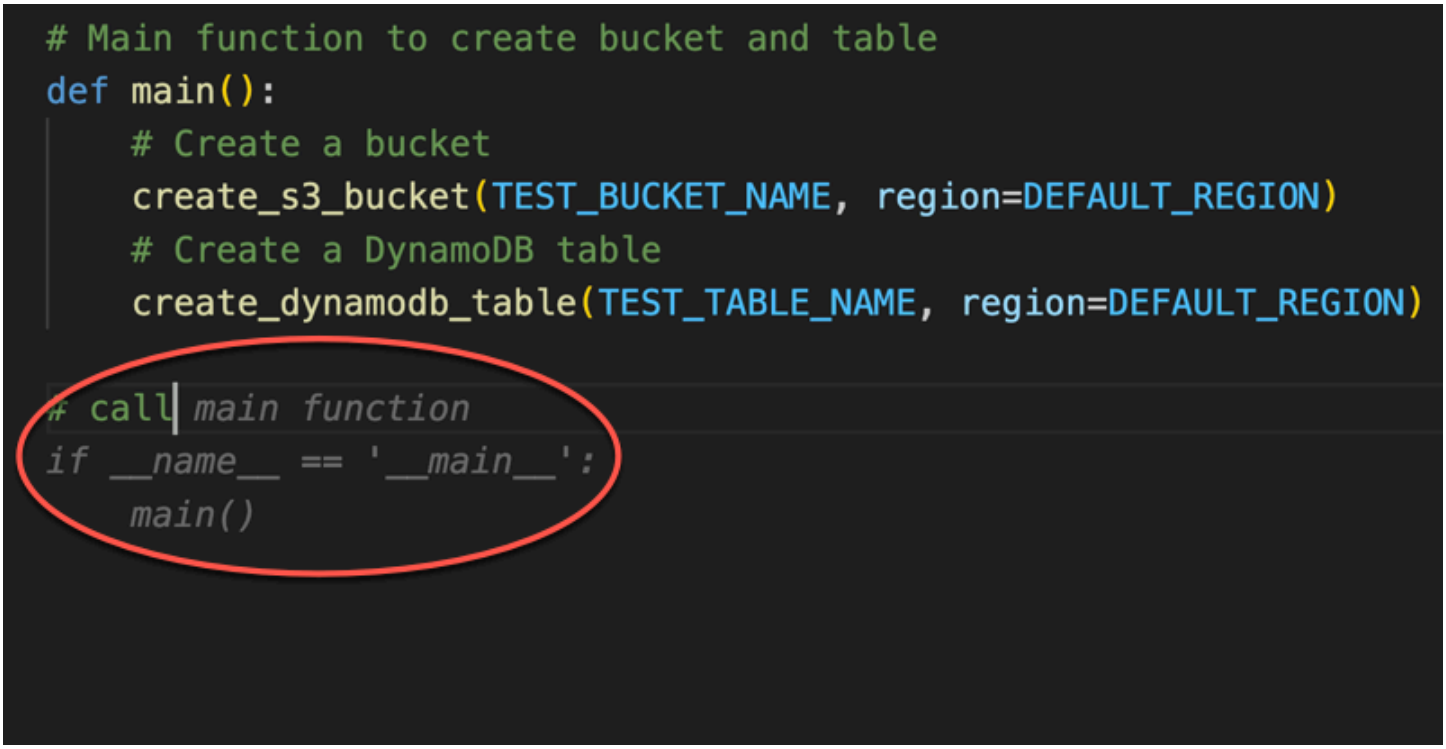
```
69  # Unit test class
70  class TestBotoWhisper(unittest.TestCase):
71      def setUp(self):
            self.s3 = session.client('s3', region_name=DEFAULT_REGION)
            self.dynamodb = session.client('dynamodb', region_name=DEFAULT_REGION)
            self.s3_resource = session.resource('s3', region_name=DEFAULT_REGION)
            self.dynamodb_resource = session.resource('dynamodb', region_name=DEFAULT_

        def tearDown(self):
            self.s3.delete_bucket(Bucket=TEST_BUCKET_NAME)
            self.dynamodb.delete_table(TableName=TEST_TABLE_NAME)

        def test_create_s3_bucket(self):
            self.assertTrue(create_s3_bucket(TEST_BUCKET_NAME, DEFAULT_REGION))

        def test_create_dynamodb_table(self):
            self.assertTrue(create_dynamodb_table(TEST_TABLE_NAME, DEFAULT_REGION))
```

コメントとコンテキストのみに基づいて、Amazon Q はメイン関数全体を生成します。

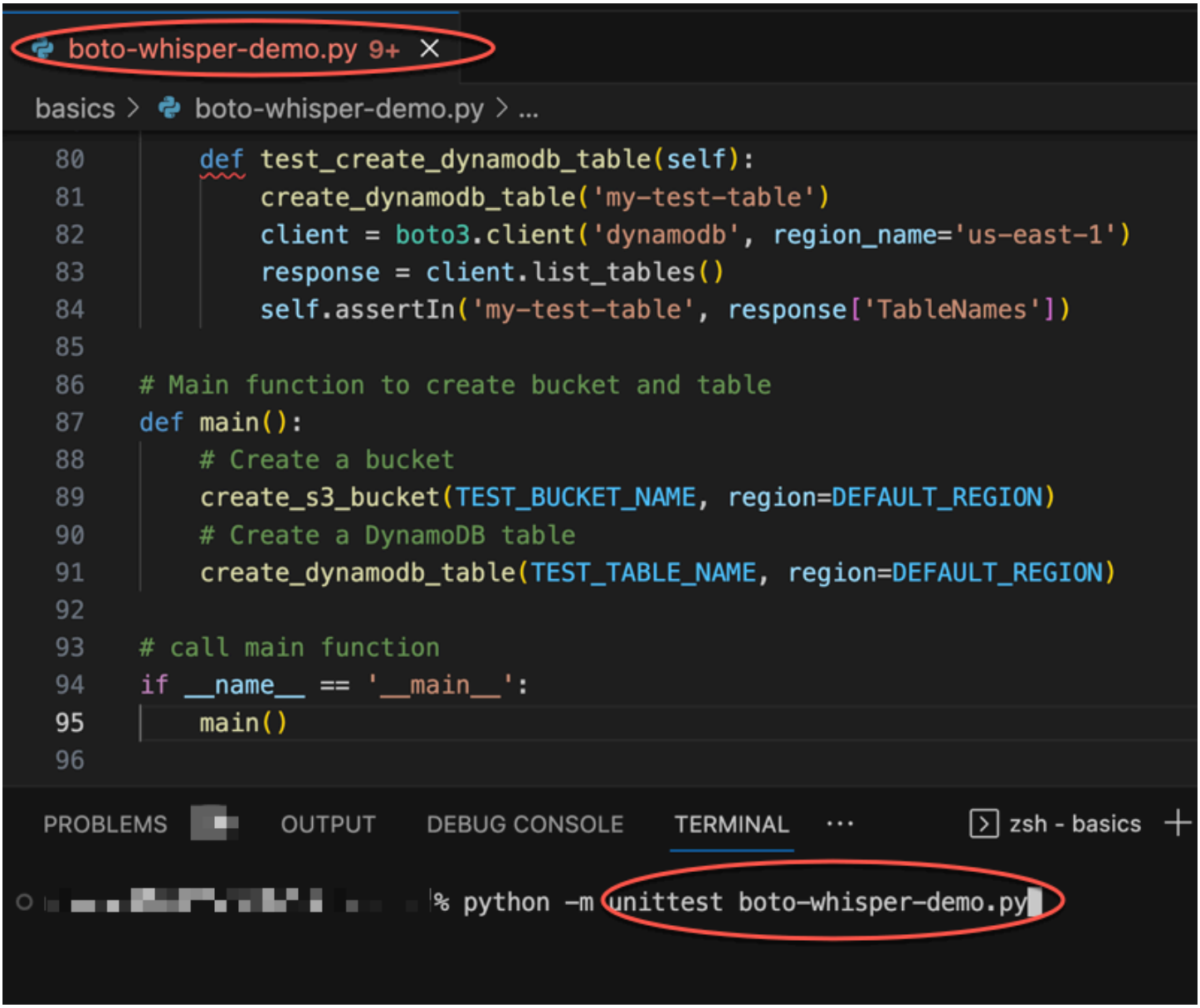
```
basics > boto-whisper-demo.py > ...
80      def test_create_dynamodb_table(self):
81          create_dynamodb_table('my-test-table')
82          client = boto3.client('dynamodb', region_name='us-east-1')
83          response = client.list_tables()
84          self.assertIn('my-test-table', response['TableNames'])
85
86  # Main function to create bucket and table
87  def main():
        create_s3_bucket(TEST_BUCKET_NAME, region='us-east-1')
        create_dynamodb_table(TEST_TABLE_NAME, region='us-east-1')
88
```

残っているのはメインガードだけで、Amazon Q はそのことを理解しています。

コメントとコンテキストのみに基づいて、Amazon Q はメイン関数全体を生成します。



最後に、デベロッパーはコーディングが行われたのと同じ IDE のターミナルからユニットテストを実行します。



AWS コーディング環境でインライン提案を生成する

Amazon Q Developer は、サードパーティーの IDE に加えて、独自のコーディング環境を提供する AWS のサービス内でインライン提案を生成できます。

以下のセクションでは、統合された AWS のサービス内で Amazon Q のインラインコードの提案を設定する方法について説明します。

> **Note**
>
> エンタープライズの一部として Amazon Q を使用している場合は、Amazon Q Developer Pro を使用することになります。この場合、組織の管理者は、コーディングを開始する前に追加のステップを完了する必要があります。詳細については、「Amazon Q Developer の使用開始」を参照してください。

トピック

- Amazon SageMaker AI Studio での Amazon Q Developer の使用
- JupyterLab での Amazon Q Developer の使用
- Amazon EMR Studio での Amazon Q Developer の使用
- AWS Glue Studio での Amazon Q Developer の使用
- での Amazon Q Developer の使用 AWS Lambda
- Amazon Q Developer を他のサービスと併用する

Amazon SageMaker AI Studio での Amazon Q Developer の使用

Amazon SageMaker AI Studio 内で Amazon Q とチャットできます。また、コードを書くときに、コードの提案を自動的に行うこともできます。

Amazon SageMaker AI Studio で Amazon Q Developer を使用するには、SageMaker AI 実行ロールに Amazon Q アクセス許可を追加する必要があります。アクセス許可の設定方法は、Amazon Q Developer 無料利用枠と Pro 利用枠のどちらを使用しているかによって異なります。

Amazon Q for Amazon SageMaker AI Studio をセットアップしてアクティブ化するには、「Amazon SageMaker AI ユーザーガイド」の「ユーザーの Amazon Q Developer のセットアップ」を参照してください。 Amazon SageMaker

JupyterLab での Amazon Q Developer の使用

このページでは、JupyterLab 用の Amazon Q Developer をセットアップしてアクティブ化する方法について説明します。アクティブ化すると、Amazon Q はコードの作成時に自動的にコードの提案を行うことができます。

> **Note**
>
> Python は、Amazon Q が JupyterLab でサポートする唯一のプログラミング言語です。

JupyterLab のインストール

コンピュータに JupyterLab をインストールします。JupyterLab が既にインストールされている場合は、次のコマンドを実行してバージョンを確認します。

```
pip show jupyterlab
```

レスポンスのバージョンをメモし、以下のセクションのいずれかで対応する指示に従います。

Jupyter Lab バージョン 4.0 以降での pip を使用したインストール

次のコマンドを使用して、JupyterLab 4 の Amazon Q 拡張機能をインストールして有効化できます。

```
# JupyterLab 4
pip install amazon-q-developer-jupyterlab-ext
```

Jupyter Lab バージョン 3.6 以降 4.0 未満での pip を使用したインストール

次のコマンドを使用して、JupyterLab 3 の Amazon Q 拡張機能をインストールして有効化できます。

```
# JupyterLab 3
pip install amazon-q-developer-jupyterlab-ext~=3.0
jupyter server extension enable amazon-q-developer-jupyterlab-ext
```

による認証 AWS ビルダー ID

次の手順では、Amazon Q を有効化する際の認証に使用する Builder ID を設定します。

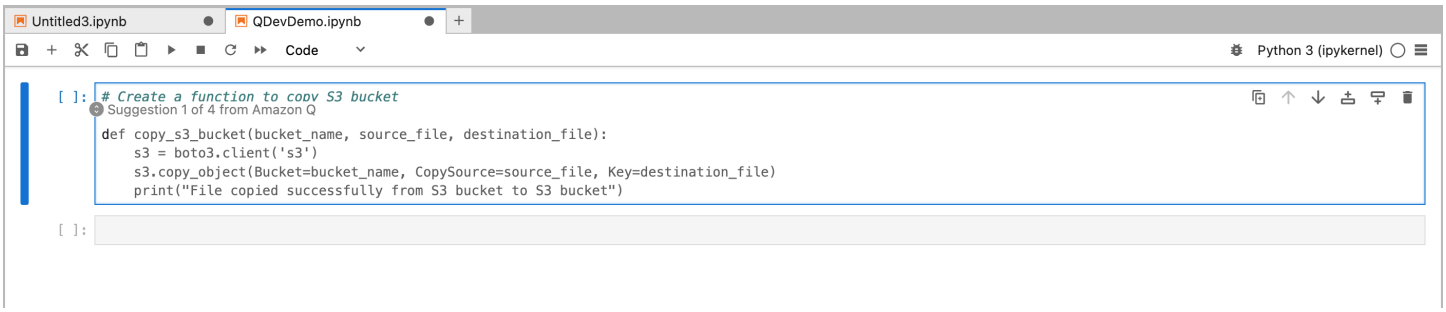
1. JupyterLab を使用しているブラウザタブを更新します。
2. ウィンドウ下部の Amazon Q パネルから、[Amazon Q を開始] を選択します。
3. ポップアップウィンドウで、[コードをコピーして進む] を選択します。

4. [AWS ビルダー IDを作成] ページで、Builder ID を持っていない場合は、個人用 E メールアドレスを入力して [次へ] を選択します。

   Builder ID を既に持っている場合は、[リクエストを承認] ページに関するステップに進みます。
5. 次の [AWS ビルダー IDを作成] ページで、名前を入力し、[次へ] を選択します。
6. E メールの確認コードを受け取ったら、空白のフィールドに入力し、[検証] を選択します。
7. 次の画面で、パスワードを選択して確認してから、作成 AWS ビルダー ID を選択します。
8. 次のページで、Amazon Q がデータにアクセスすることを許可する [許可] を選択します。

これで、Builder ID を使用して JupyterLab の Amazon Q にログインしているはずです。

コーディングを開始するには、「ショートカットキーの使用」を参照してください。



Amazon EMR Studio での Amazon Q Developer の使用

このページでは、Amazon EMR Studio 用の Amazon Q Developer をセットアップしてアクティブ化する方法について説明します。アクティブ化すると、Amazon Q は ETL コードの作成時に自動的にコードの提案を行うことができます。

> **Note**
>
> Amazon Q は Python をサポートしています。Python は、Amazon EMR Studio の Spark ジョブの ETL スクリプトのコーディングに使用できます。

Amazon Q と連携するように Amazon EMR Studio を設定する手順は次のとおりです。

1. Amazon EMR Studio ノートブックを設定します。
2. Amazon EMR Studio ノートブックの IAM ユーザーロールに次のポリシーをアタッチします。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonQDeveloperPermissions",
            "Effect": "Allow",
            "Action": [
                "codewhisperer:GenerateRecommendations"
            ],
            "Resource": "*"
        }
    ]
}
```

3. Amazon EMR コンソールを開きます。
4. Amazon EMR Studio で、[Workspaces (ノートブック)] を選択します。
5. WorkSpace を選択し、[削除] を選択します。

AWS Glue Studio での Amazon Q Developer の使用

このページでは、AWS Glue Studio ノートブックで Amazon Q Developer をセットアップしてアクティブ化する方法について説明します。アクティブ化すると、Amazon Q は ETL コードの作成時に自動的にコードの提案を行うことができます。

> **Note**
>
> Amazon Q は、Python と Scala の両方をサポートしています。これは、 AWS Glue Studio の Spark ジョブの ETL スクリプトのコーディングに使用される 2 つの言語です。

次の手順では、Amazon Q と連携 AWS Glue するように を設定します。

1. AWS Glue Studio Notebook をセットアップします。
2. Glue Studio ノートブックの IAM ロールに次のポリシーをアタッチします。

   > **Note**
   >
   > `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Sid": "AmazonQDeveloperPermissions",
               "Effect": "Allow",
               "Action": [
                   "codewhisperer:GenerateRecommendations"
               ],
               "Resource": "*"
           }
       ]
   }
   ```

3. Glue コンソールを開きます。
4. [ETL ジョブ] で、[ノートブック] を選択します。
5. [Jupyter Notebook] が選択されていることを確認します。[作成] を選択します。
6. [Job name] (ジョブ名) を入力します。

7. IAM ロールの場合は、Amazon Q とやり取りするように設定したロールを選択します。
8. [ノートブックの開始] を選択します。

での Amazon Q Developer の使用 AWS Lambda

このドキュメントでは、Lambda コンソールで Amazon Q Developer をアクティブ化する方法を説明します。アクティブ化されると、Amazon Q により、Lambda コードエディタで関数を開発する際に、推奨コードの提案がオンデマンドで行われます。

> **Note**
>
> Lambda コンソールでは、Amazon Q は Python と Node.js ランタイムを使用する関数のみをサポートします。

AWS Identity and Access Management Lambda の アクセス許可

Lambda コンソールで Amazon Q が提案を行えるようにするには、IAM ユーザーまたは IAM ロールに対して適切な IAM アクセス許可を有効にする必要があります。次の IAM ポリシーの例で示すように、`codewhisperer:GenerateRecommendations` アクセス許可を追加する必要があります。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AmazonQDeveloperPermissions",
      "Effect": "Allow",
      "Action": ["codewhisperer:GenerateRecommendations"],
      "Resource": "*"
    }
```

```
  ]
}
```

IAM ポリシーを使用して IAM プリンシパルに制限的なアクセス許可をグラントするのがベストプラクティスです。での IAM の使用の詳細については AWS Lambda、「 AWS Lambda デベロッパーガイド」の「 での ID とアクセスの管理 AWS Lambda」を参照してください。

Lambda での Amazon Q Developer のアクティブ化

Lambda コンソールのコードエディタで Amazon Q を有効にするには、次の手順を実行します。

1. Lambda コンソールの関数ページを開いて、編集する関数を選択します。
2. コードエディタに入力すると、Amazon Q からの自動コード提案がデフォルトで有効になります。提案を一時停止するには、コードソースパネルの左下隅にある Amazon Q を選択します。コマンドパレットがコードソースパネルの上部に表示されます。そこから、自動提案を一時停止を選択します。

ショートカットキーについては、「ショートカットキーの使用」を参照してください。

Amazon Q Developer を他のサービスと併用する

AWS Identity and Access Management 他の サービスの アクセス許可

他のサービスのコンテキストで Amazon Q から提案を行うには、IAM ユーザーまたは IAM ロールに対して正しい IAM アクセス許可を有効にする必要があります。次の IAM ポリシーの例で示すように、`codewhisperer:GenerateRecommendations` アクセス許可を追加する必要があります。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

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

```
  "Statement": [
    {
      "Sid": "AmazonQDeveloperPermissions",
      "Effect": "Allow",
      "Action": ["codewhisperer:GenerateRecommendations"],
      "Resource": "*"
    }
  ]
}
```

IAM ポリシーを使用して IAM プリンシパルに制限的なアクセス許可をグラントするのがベストプラクティスです。詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

ショートカットキーの使用

Amazon Q からインラインの提案を取得するときは、Amazon Q の起動や提案の受け入れなど、一般的なアクションにキーボードショートカットを使用できます。

コードを作成している統合開発環境 (IDE) を選択すると、IDE のキーボードショートカットを確認できます。

Visual Studio Code

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	右矢印
前の提案	左矢印
提案を拒否する	ESC、バックスペース、または入力を続けて文字が一致しなくなり次第、提案は表示されなくなります。

[Action] (アクション)	キーボードショートカット
次の単語を受け入れる	オプション + 右矢印

VS Code のキーバインドを変更するには、VS Code ウェブサイトの「Key Bindings for Visual Studio Code 」を参照してください。

> **Note**
>
> VS Code のインライン提案ツールバーは、デフォルトで無効になっています。詳細については、VS Code ウェブサイトの「Redesigned inline suggestions toolbar」を参照してください。

JetBrains

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	右矢印
前の提案	左矢印
提案を拒否する	ESC、バックスペース、または入力を続けて文字が一致しなくなり次第、提案は表示されなくなります。

IntelliJ のキーバインドを変更するには、JetBrains ウェブサイトの「IntelliJ IDEA keyboard shortcuts」を参照してください。

Eclipse

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	MacOS: オプション +] Windows: Alt +]
前の提案	MacOS: オプション + [Windows: Alt + [
提案を拒否する	ESC、バックスペース、または入力を続けて文字が一致しなくなり次第、提案は表示されなくなります。

でキーバインディングを変更するにはEclipse、 Eclipseドキュメントの「キーバインディングの変更」を参照してください。

Toolkit for Visual Studio

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する キーバインドの `AWSToolkit.CodeWhisperer.GetSuggestion`	Windows: Alt + C
提案を受け入れる	タブ
次の提案 キーバインドの `Edit.NextSuggestion`	Windows: Alt + .
前の提案	Windows: Alt + ,

[Action] (アクション)	キーボードショートカット
キーバインドの `Edit.PreviousSuggestion`	
提案を拒否する	ESC、バックスペース、または入力を続けて文字が一致しなくなり次第、提案は表示されなくなります。

Microsoft の Visual Studio のデフォルトキーボードショートカットも参照してください。

Visual Studio でキーバインドを変更するには、[ツール] -> [オプション] -> [キーボード] を使用します。

Amazon SageMaker AI

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	下矢印
前の提案	上矢印
提案を拒否する	ESC

JupyterLab

[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ

[Action] (アクション)	キーボードショートカット
次の提案	下矢印
前の提案	上矢印
提案を拒否する	ESC

AWS Glue Studio Notebook

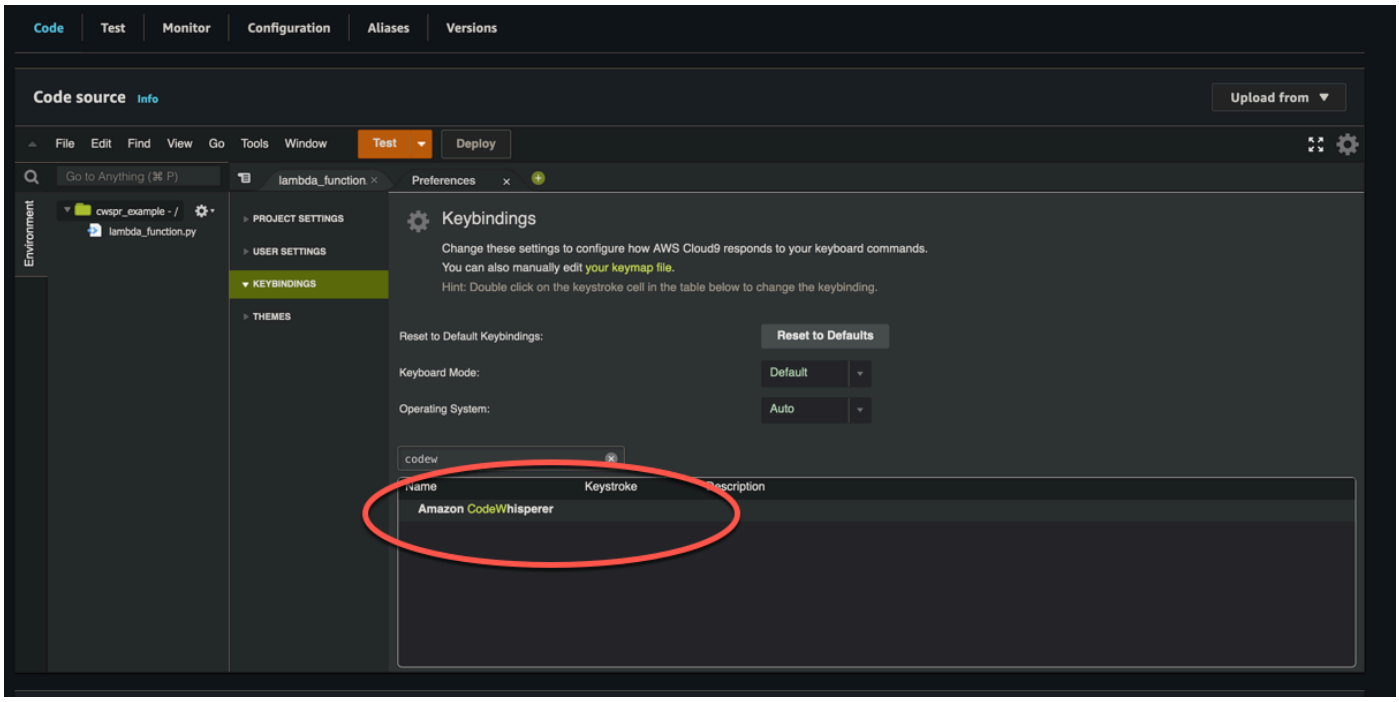
[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	下矢印
前の提案	上矢印
提案を拒否する	ESC

AWS Lambda

[Action] (アクション)	キーボードショートカット
コード提案を手動で取得	MacOS: Option + C Windows: Alt + C
提案の受け入れ	タブ
提案の拒否	ESC、バックスペース、任意の方向へのスクロール、入力の続行により、提案は自動的に消えます。

キーバインドを変更する手順は次のとおりです。

1. 特定の関数を表示しながら、歯車アイコンをクリックして、[環境設定] タブを開きます。
2. [設定] タブで、[キー割り当て] を選択します。
3. キーバインド検索ボックスに「Amazon Q」と入力します。

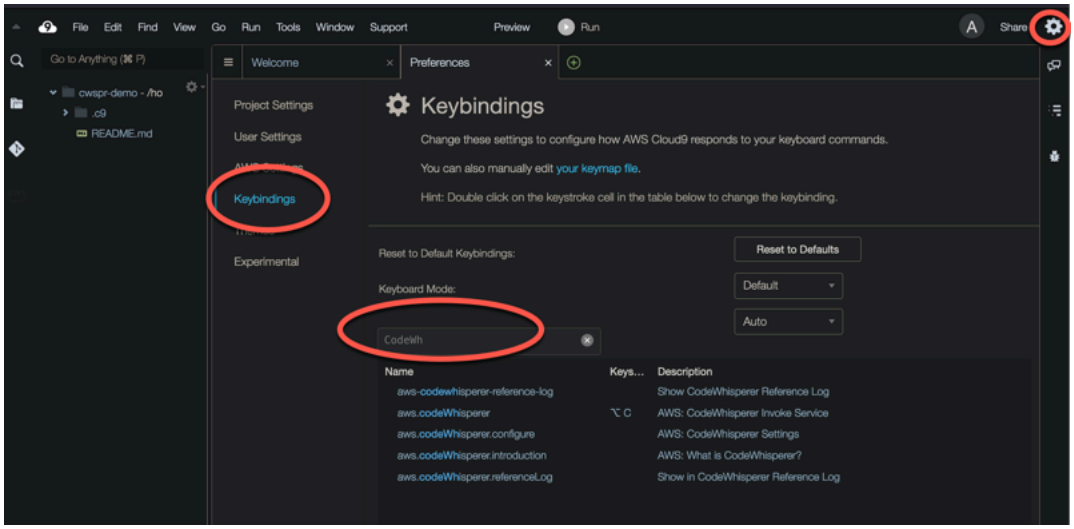


AWS Cloud9

[Action] (アクション)	キーボードショートカット
コード提案を手動で取得	MacOS: Option + C Windows: Alt + C
提案の受け入れ	タブ
提案の拒否	ESC、バックスペース、任意の方向へのスクロール、入力の続行により、提案は自動的に消えます。

1. 特定の環境を表示しながら、歯車アイコンをクリックして、[環境設定] タブを開きます。

2. [設定] タブで、[キー割り当て] を選択します。
3. キーバインド検索ボックスに「Amazon Q」と入力します。
4. Keystroke 列で、関心のある関数に対応するスペースをダブルクリックします。
5. 関数を割り当てるキーを入力します。



コードリファレンスの使用

Amazon Q は、オープンソースプロジェクトからも、ある程度学習します。提案が公開されているコードに似ている場合があります。コードリファレンスには、提案の生成に使用されるソース Amazon Q に関する情報が含まれます。

トピック

- コードリファレンスの表示と更新
- コードリファレンスのオンとオフの切り替え
- リファレンスを含むコードからオプトアウトする

コードリファレンスの表示と更新

リファレンスログを使用すると、公開されているコードに似たコードレコメンデーションへのリファレンスを表示できます。Amazon Q によって出されたコード提案は、更新することも編集することもできます。

IDE を選択すると、コードリファレンスを表示および更新する手順を確認できます。

Visual Studio Code

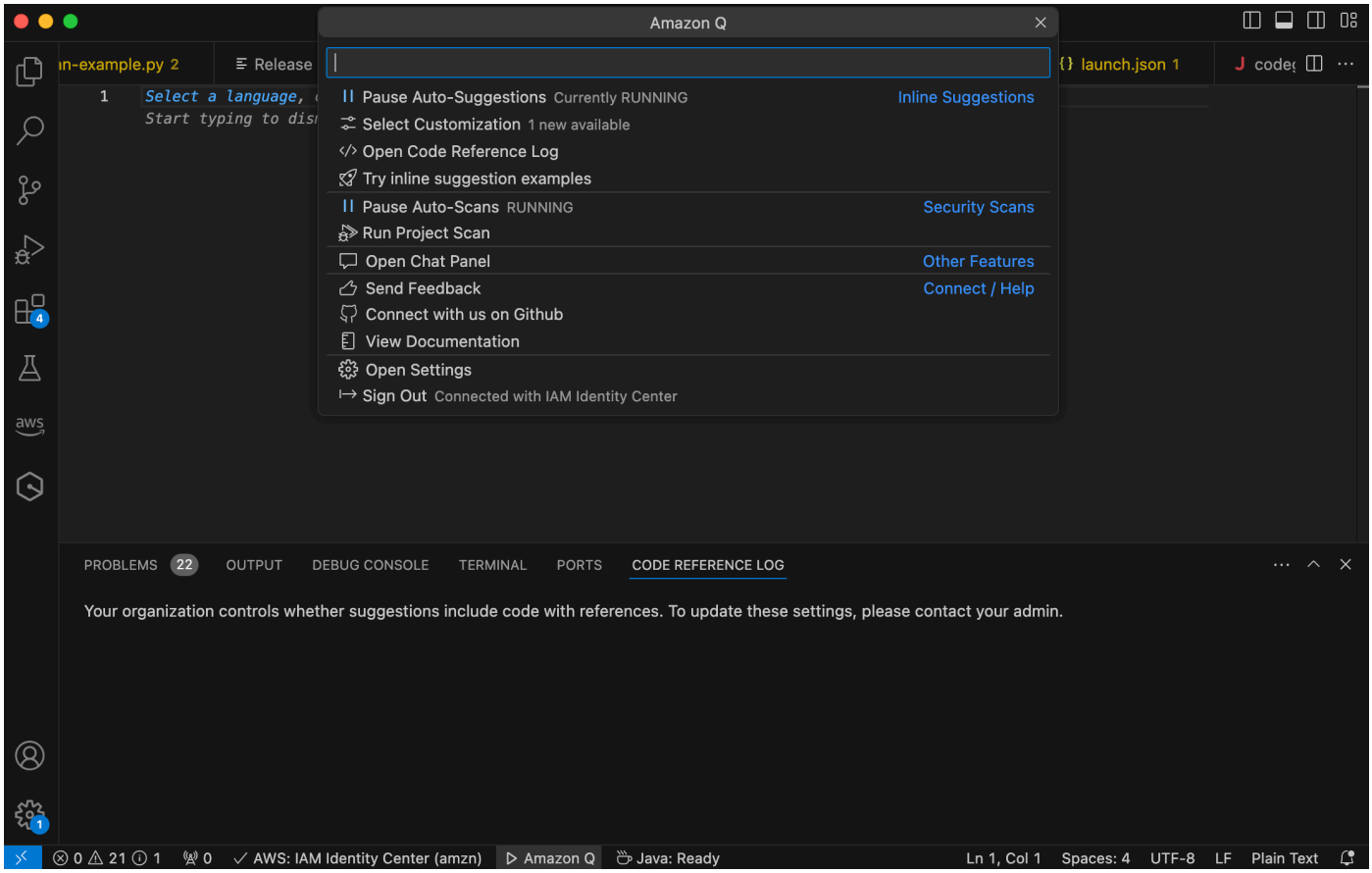
VS Code で Amazon Q リファレンスログを表示する手順は次のとおりです。

1. VS Code と Amazon Q 拡張機能の両方の最新バージョンを使用していることを確認してください。
2. VS Code で、IDE ウィンドウの下部にあるコンポーネントトレイから [Amazon Q] を選択します。

   Amazon Q タスクバーが IDE ウィンドウの上部に表示されます。
3. [コードリファレンスログを開く] を選択します。

   コードリファレンスログのタブが開きます。コード提案へのリファレンスが一覧表示されます。

次の画像は、Amazon Q タスクバーとコードリファレンスログのタブが開いている様子を示しています。



JetBrains

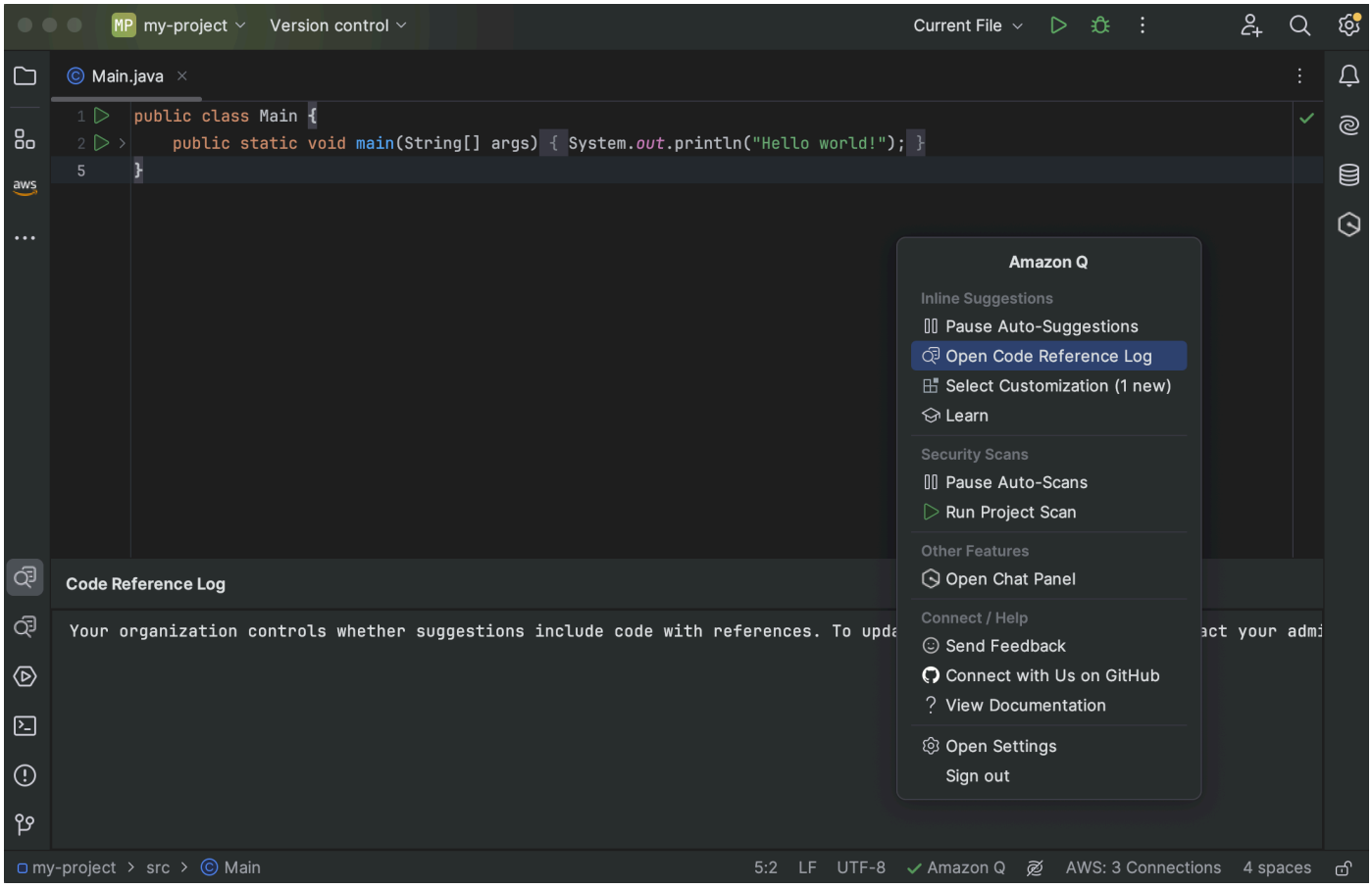
JetBrains IDE で Amazon Q リファレンスログを表示する手順は次のとおりです。

1. JetBrains IDE と Amazon Q プラグインの両方の最新バージョンを使用していることを確認してください。
2. JetBrains で、IDE ウィンドウの下部にあるステータスバーから [Amazon Q] を選択します。

   Amazon Q タスクバーがステータスバーの上に開きます。
3. [コードリファレンスログを開く] を選択します。

   コードリファレンスログのタブが開きます。コード提案へのリファレンスが一覧表示されます。

次の画像は、Amazon Q タスクバーとコードリファレンスログのタブが開いている様子を示しています。

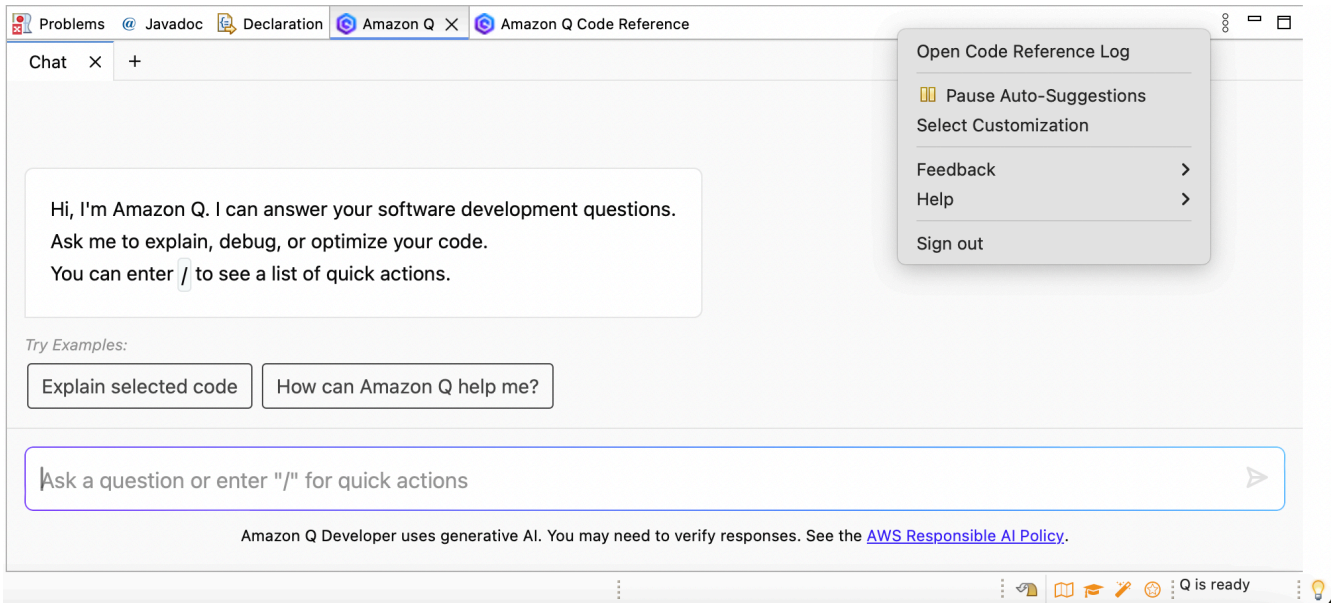


Eclipse

Eclipse IDE で Amazon Q リファレンスログを表示する手順は次のとおりです。

1. IDE と Amazon Q Eclipse プラグインの両方の最新バージョンを使用していることを確認してください。
2. Eclipse IDE で、IDE の右上隅にある Amazon Q アイコンを選択します。
3. Amazon Q チャットタブを開いた状態で、タブの右上隅にある省略記号アイコンを選択します。Amazon Q タスクバーが開きます。

   次の図は、IDE の Amazon Q Eclipse タスクバーを示しています。

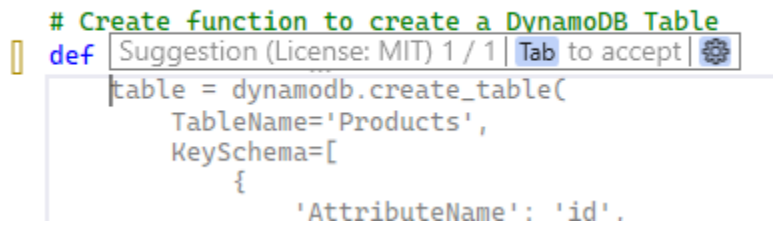


4. [コードリファレンスログを開く] を選択します。

   コードリファレンスログのタブが開きます。コード提案へのリファレンスが一覧表示されます。

Toolkit for Visual Studio

Amazon Q が Toolkit for Visual Studio のリファレンスを含むコードを提案すると、リファレンスタイプが提案の説明に表示されます。

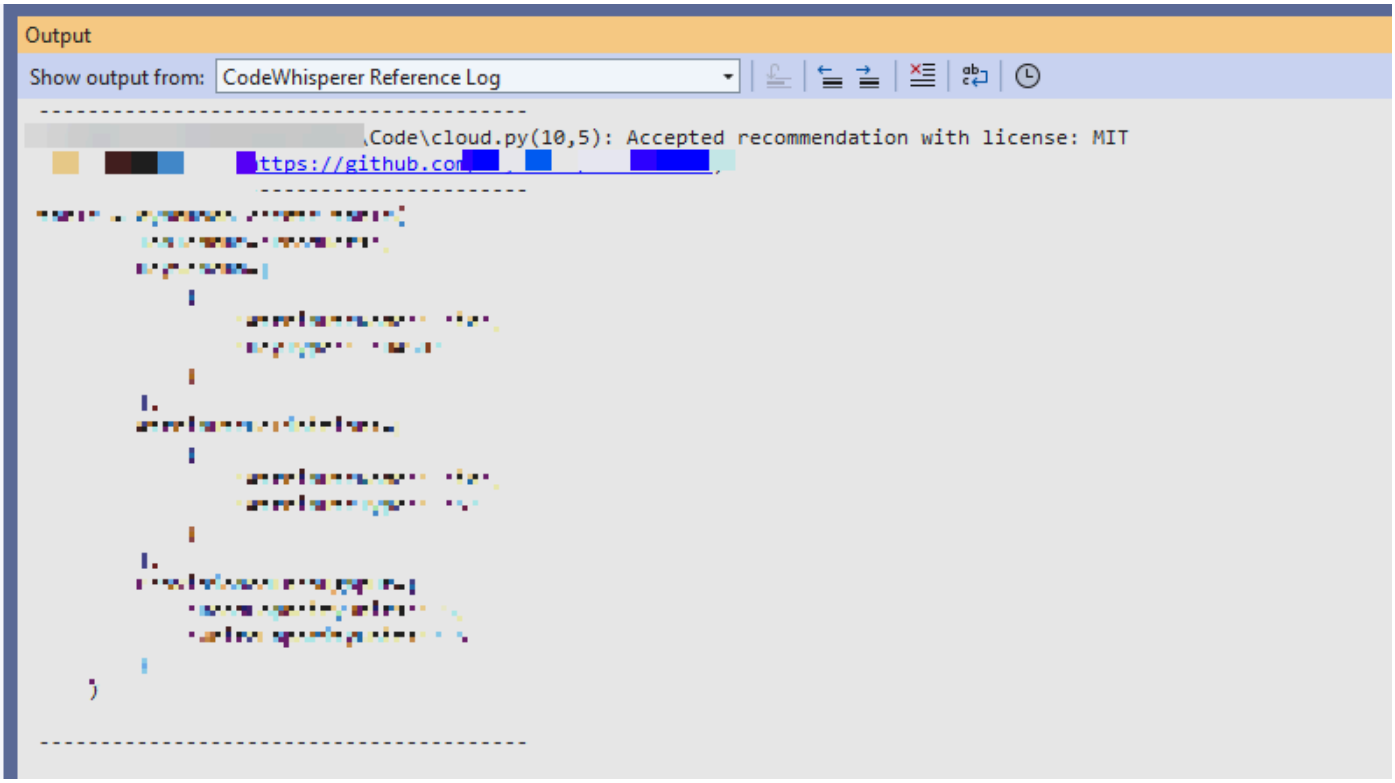


受け入れられた提案で、リファレンスを含むものはすべてリファレンスログに記録されます。

リファレンスログにアクセスするには、 AWS アイコンを選択し、Open Code Reference Log を選択します。

受け入れられた提案で、リファレンスを含むものが一覧表示されます。この一覧には以下が含まれます。

- 提案が受け入れられた場所。これをダブルクリックすると、コード内のその場所に移動します。
- 関連ライセンス
- 参照元ソースコード
- リファレンスに帰属するコードの断片



AWS Cloud 9

AWS Cloud 9 で Amazon Q を使用する場合、コード参照はデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. AWS Cloud 9 コンソールの左上隅で、 AWS Cloud 9 ロゴを選択します。
2. ドロップダウンメニューから [環境設定] を選択します。

   コンソールの右側で、[環境設定] タブが開きます。
3. [環境設定] タブの [プロジェクト設定] にある [拡張機能] で、[AWS Toolkit] を選択します。
4. [Amazon Q: コードリファレンスのある提案を含める] を選択するか、選択解除します。

Lambda

Lambda の Amazon Q はコードリファレンスをサポートしていません。Amazon Q を Lambda で使用すると、リファレンスを含むコード提案は省略されます。

SageMaker AI Studio

SageMaker AI Studio で Amazon Q リファレンスログを表示するには、次の手順を使用します。

1. SageMaker AI Studio ウィンドウの下部で、Amazon Q パネルを開きます。
2. [コードリファレンスログを開く] を選択します。

JupyterLab

JupyterLab で Amazon Q リファレンスログを表示する手順は次のとおりです。

1. JupyterLab ウィンドウの下部で、Amazon Q パネルを開きます。
2. [コードリファレンスログを開く] を選択します。

AWS Glue Studio Notebook

AWS Glue Studio Notebook で Amazon Q リファレンスログを表示するには、次の手順を使用します。

1. AWS Glue Studio Notebook ウィンドウの下部で、Amazon Q パネルを開きます。

2. [コードリファレンスログを開く] を選択します。

コードリファレンスのオンとオフの切り替え

ほとんどの IDE で、コードリファレンスはデフォルトでオンになっています。IDE を選択すると、コードリファレンスをオンまたはオフにする手順を確認できます。

Visual Studio Code

VS Code で Amazon Q を使用すると、コードリファレンスはデフォルトでオンになります。

オフにする、または後でオンに戻す手順は次のとおりです。

1. VS Code と Amazon Q 拡張機能の両方の最新バージョンを使用していることを確認してください。
2. VS Code で、IDE ウィンドウの下部にあるコンポーネントトレイから [Amazon Q] を選択します。

   Amazon Q タスクバーが IDE ウィンドウの上部に表示されます。
3. [設定を開く] をクリックします。設定タブが開き、Amazon Q に関連するオプションが表示されます。
4. リファレンスでコードを表示するの横にあるボックスを選択または選択解除します。

JetBrains

JetBrains IDE で Amazon Q を使用する場合、コードリファレンスはデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. JetBrains IDE と Amazon Q プラグインの両方の最新バージョンを使用していることを確認してください。
2. JetBrains で、IDE ウィンドウの下部にあるステータスバーから [Amazon Q] を選択します。

   Amazon Q タスクバーがステータスバーの上に開きます。
3. [設定を開く] をクリックします。設定ウィンドウが開き、Amazon Q に関連するオプションが表示されます。
4. リファレンスでコードを表示するの横にあるボックスを選択または選択解除します。

Eclipse

Eclipse で Amazon Q を使用すると、コードリファレンスはデフォルトでオンになります。

オフにする、または後でオンに戻す手順は次のとおりです。

1. IDE と Amazon Q Eclipse プラグインの両方の最新バージョンを使用していることを確認してください。
2. IDE Eclipse で設定を開きます。
3. 左側のナビゲーションバーから Amazon Q を選択します。
4. リファレンスでコードを表示するの横にあるボックスを選択または選択解除します。
5. [Apply (適用)] を選択して変更を保存します。

Toolkit for Visual Studio

Toolkit for Visual Studio で Amazon Q を使用する場合、コードリファレンスはデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. Toolkit for Visual Studio の最新バージョンを使用していることを確認します。
2. Visual Studio でオプションを開きます。
3. 左側のナビゲーションバーから AWS Toolkit を選択し、Amazon Q を選択します。
4. リファレンスで提案を含めるの横にあるドロップダウンから、True または False を選択します。
5. OK を選択して変更を保存します。

AWS Cloud 9

AWS Cloud 9 で Amazon Q を使用する場合、コード参照はデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. AWS Cloud 9 コンソールの左上隅で、 AWS Cloud 9 ロゴを選択します。
2. ドロップダウンメニューから [環境設定] を選択します。

   コンソールの右側で、[環境設定] タブが開きます。

3. [環境設定] タブの [プロジェクト設定] にある [拡張機能] で、[AWS Toolkit] を選択します。
4. [Amazon Q: コードリファレンスのある提案を含める] を選択するか、選択解除します。

Lambda

Lambda の Amazon Q はコードリファレンスをサポートしていません。Amazon Q を Lambda で使用すると、リファレンスを含むコード提案は省略されます。

SageMaker AI Studio

SageMaker AI Studio で Amazon Q を使用する場合、コード参照はデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. SageMaker AI Studio ウィンドウの上部から設定を選択します。
2. [設定] ドロップダウンから、[詳細設定エディタ] を選択します。
3. Amazon Q ドロップダウンで、[コードリファレンスを含む提案を有効にする] の横にあるチェックボックスをオンまたはオフにします。

JupyterLab

JupyterLab で Amazon Q を使用する場合、コードリファレンスはデフォルトでオンになっています。

オフにする、または後でオンに戻す手順は次のとおりです。

1. JupyterLab ウィンドウの上部から、[設定] を選択します。
2. [設定] ドロップダウンから、[詳細設定エディタ] を選択します。
3. Amazon Q ドロップダウンで、[コードリファレンスを含む提案を有効にする] の横にあるチェックボックスをオンまたはオフにします。

AWS Glue Studio Notebook

1. AWS Glue Studio Notebook ウィンドウの下部から Amazon Q を選択します。
2. ポップアップメニューから、[リファレンスを含むコード] の横にあるスイッチを切り替えます。

> **Note**
>
> コード参照の一時停止は、現在の AWS Glue Studio ノートブックの期間中のみ有効です。

リファレンスを含むコードからオプトアウトする

一部の IDE では、管理者レベルでリファレンスを含む提案の受け取りからオプトアウトできます。

IDE を選択すると、管理者としてオプトアウトする手順を確認できます。

Visual Studio Code

エンタープライズ管理者の場合は、コードリファレンスを含む提案から組織全体でオプトアウトできます。これを行うと、組織内の個々のデベロッパーは IDE を通じてオプトインできなくなります。これらのデベロッパーは、前のセクションで説明したチェックボックスをオンまたはオフにできますが、エンタープライズレベルでオプトアウトしている場合は効果がありません。

エンタープライズレベルでリファレンスを含む提案をオプトアウトする手順は次のとおりです。

1. Amazon Q Developer コンソールで、[設定] を選択します。
2. [Amazon Q Developer アカウントの詳細] ペインで、[編集] を選択します。
3. [詳細の編集] ページの [詳細設定] ペインで、[コードリファレンスを含む提案] のチェックボックスをオフにします。
4. [Save changes] (変更の保存) をクリックします。

JetBrains

エンタープライズ管理者の場合は、コードリファレンスを含む提案から組織全体でオプトアウトできます。これを行うと、組織内の個々のデベロッパーは IDE を通じてオプトインできなくなります。これらのデベロッパーは、前のセクションで説明したチェックボックスをオンまたはオフにできますが、エンタープライズレベルでオプトアウトしている場合は効果がありません。

エンタープライズレベルでリファレンスを含む提案をオプトアウトする手順は次のとおりです。

1. Amazon Q Developer コンソールで、[設定] を選択します。
2. [Amazon Q Developer アカウントの詳細] ペインで、[編集] を選択します。

3. [詳細の編集] ページの [詳細設定] ペインで、[コードリファレンスを含む提案] のチェックボックスをオフにします。
4. [Save changes] (変更の保存) をクリックします。

Eclipse

エンタープライズ管理者の場合は、コードリファレンスを含む提案から組織全体でオプトアウトできます。これを行うと、組織内の個々のデベロッパーは IDE を通じてオプトインできなくなります。これらのデベロッパーは、前のセクションで説明したチェックボックスをオンまたはオフにできますが、エンタープライズレベルでオプトアウトしている場合は効果がありません。

エンタープライズレベルでリファレンスを含む提案をオプトアウトする手順は次のとおりです。

1. Amazon Q Developer コンソールで、[設定] を選択します。
2. [Amazon Q Developer アカウントの詳細] ペインで、[編集] を選択します。
3. [詳細の編集] ページの [詳細設定] ペインで、[コードリファレンスを含む提案] のチェックボックスをオフにします。
4. [Save changes] (変更の保存) をクリックします。

Toolkit for Visual Studio

エンタープライズレベルでリファレンスを含む提案をオプトアウトする手順は次のとおりです。

1. コードリファレンスの設定には、次の 2 つの方法のいずれかでアクセスできます。
    a. ウィンドウの端にある Amazon Q アイコンをクリックし、[オプション...] を選択します。
    b. [ツール] -> [AWS Toolkit] -> [Amazon Q] に移動します。
2. リファレンスを含む提案を含めるかどうかに応じて、トグルを [True] または [False] に変更します。

AWS Cloud 9

Amazon Q in AWS Cloud 9 は、エンタープライズレベルでのリファレンスを使用したコード提案のオプトアウトをサポートしていません。

個々のデベロッパーレベルでオプトアウトするには、「コードリファレンスのオンとオフの切り替え」を参照してください。

Lambda

Lambda の Amazon Q はコードリファレンスをサポートしていません。Amazon Q を Lambda で使用すると、リファレンスを含むコード提案は省略されます。

SageMaker AI Studio

Amazon Q は、SageMaker AI Studio のエンタープライズレベルでのリファレンスによるコード提案のオプトアウトをサポートしていません。

JupyterLab

Amazon Q では、JupyterLab で、リファレンスを含むコード提案からのエンタープライズレベルでのオプトアウトをサポートしていません。

AWS Glue Studio Notebook

Amazon Q は、 AWS Glue Studio Notebook のリファレンスを使用したコード提案のオプトアウトをサポートしていません。

コードの例

Amazon Q は、さまざまなシナリオでコードを提案できます。任意のプログラミング言語でコードを記述する際にどのように役立つかを理解するには、次のコード例を参照してください。

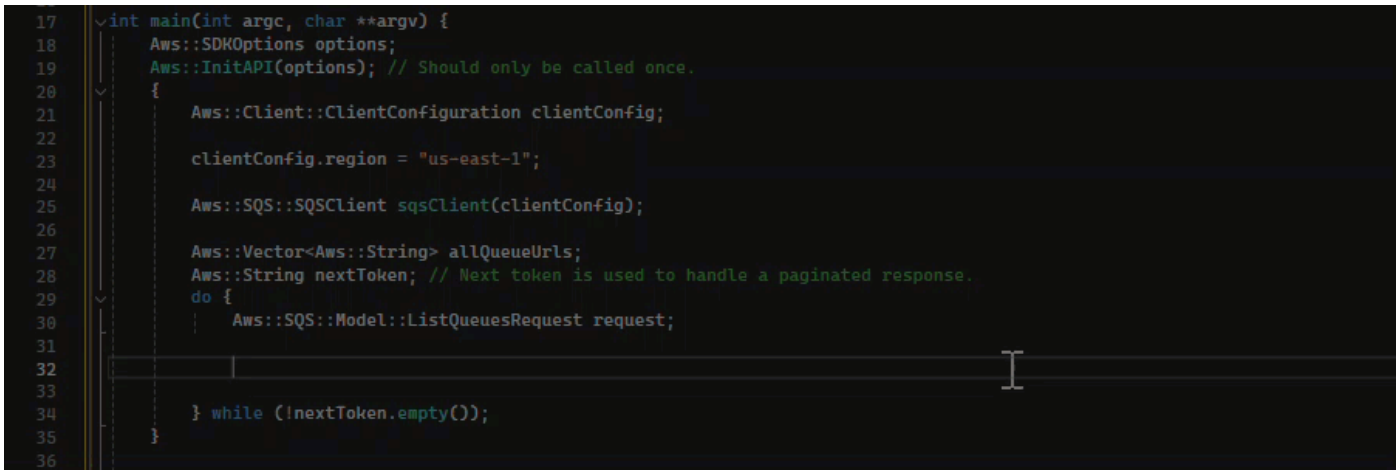
トピック

- Amazon Q Developer を使用して単一行コードを補完する
- Amazon Q Developer を使用して完全な関数を生成する
- Amazon Q Developer を使用してブロックを補完する
- Amazon Q Developer を使用して docstring、JSDoc、Javadoc を補完する
- Amazon Q Developer を使用して行ごとの提案を行う

Amazon Q Developer を使用して単一行コードを補完する

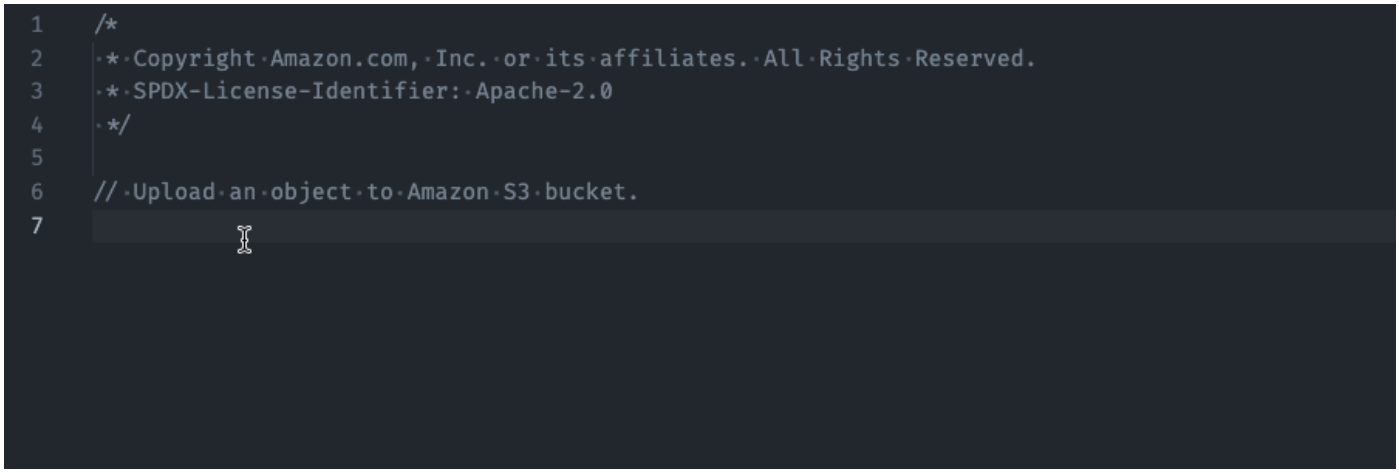
単一行のコードを入力し始めると、Amazon Q は現在と以前の入力に基づいて提案を行います。

C++



JavaScript

この例では、Amazon Q はデベロッパーが開始したコード行を補完します。

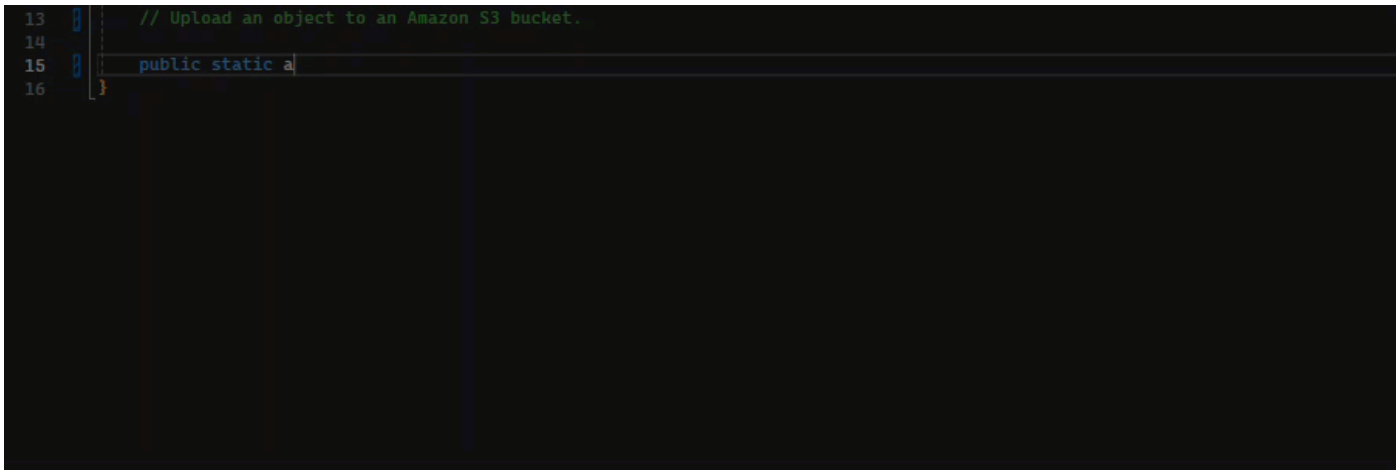


TypeScript

この例では、ユーザーが完全なコメントを入力し、Amazon Q がそれに従ってコードを提供します。

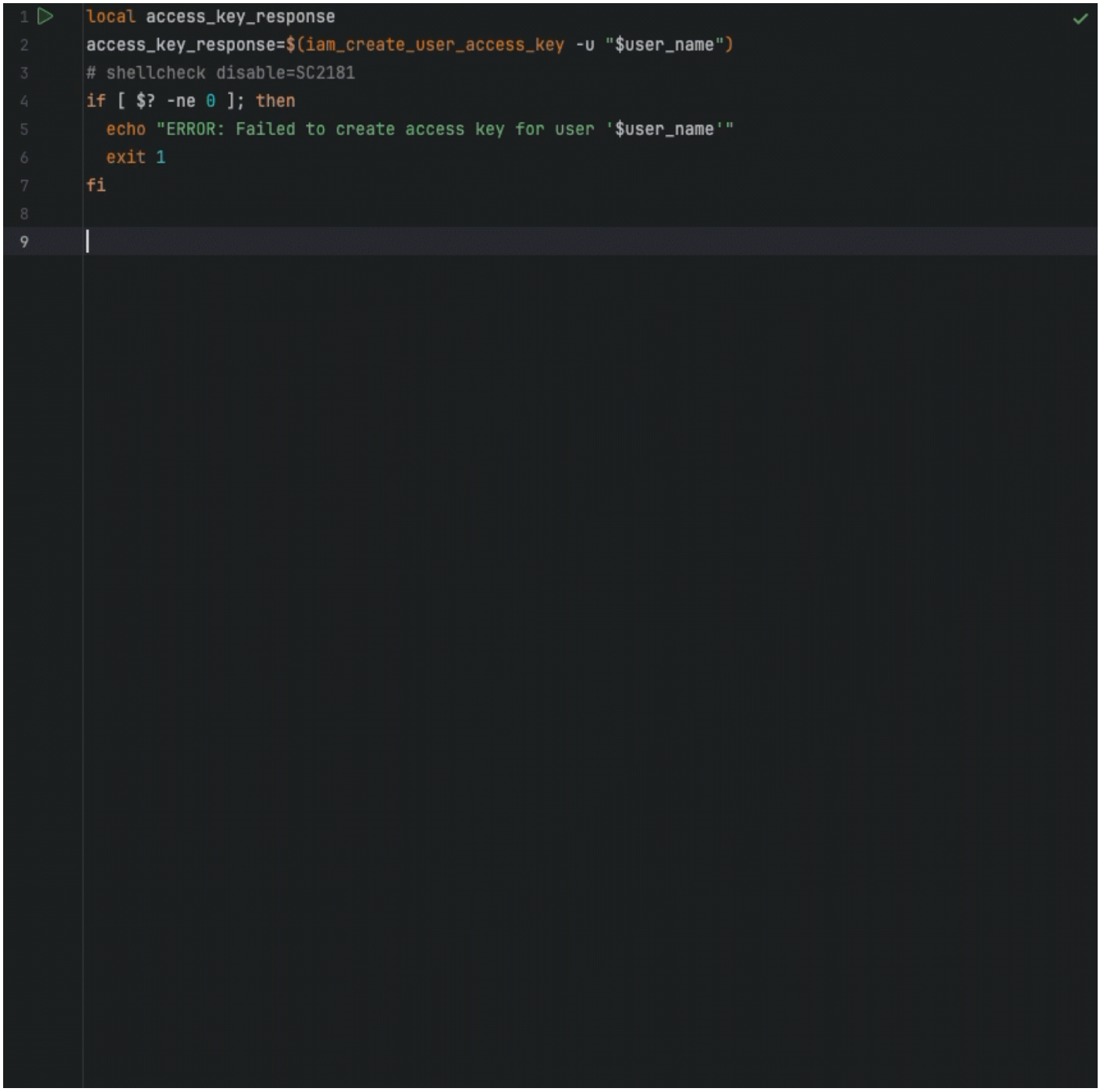
C#

この例では、Amazon Q はコメントに基づいて 1 行のコードを提案します。



Shell

下の画像では、Amazon Q が 1 行のコードを補完する方法を提案しています。

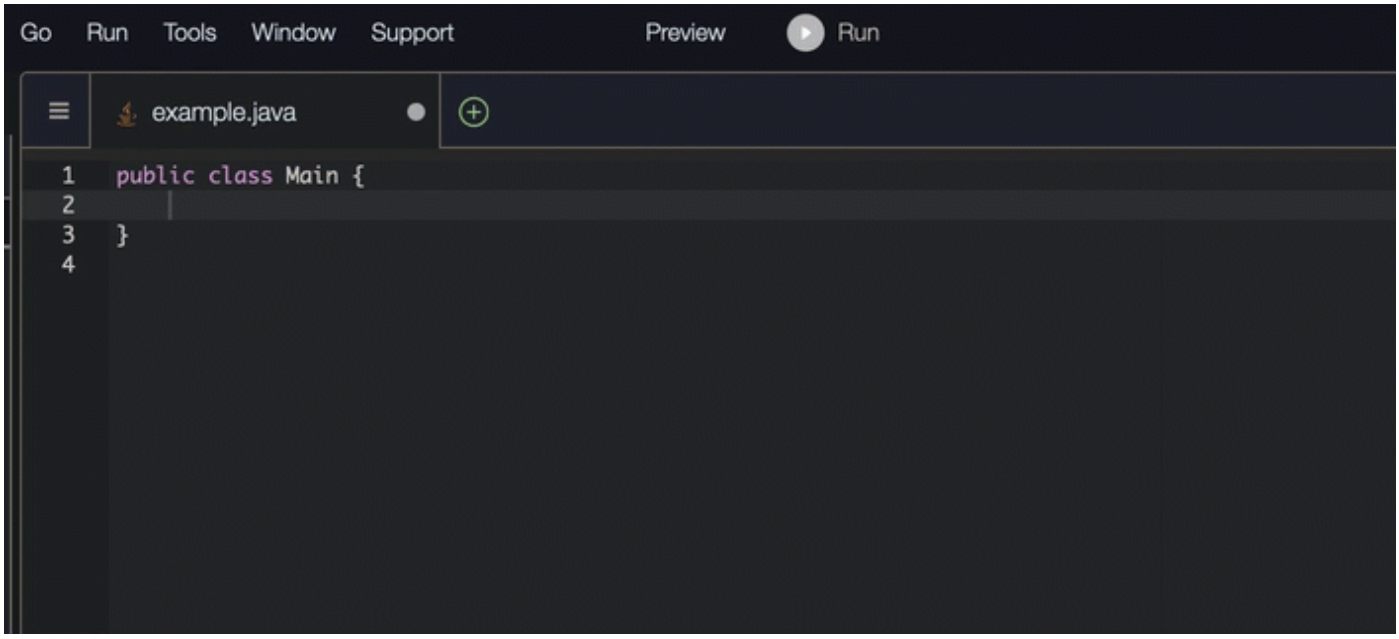


Java

単一行のコードを入力し始めると、Amazon Q は現在と以前の入力に基づいて提案を行います。

以下の Java での例では、ユーザーは既存のクラスに `public` の文字列を入力します。

入力に基づき、Amazon Q はメインメソッドの署名の提案を生成します。



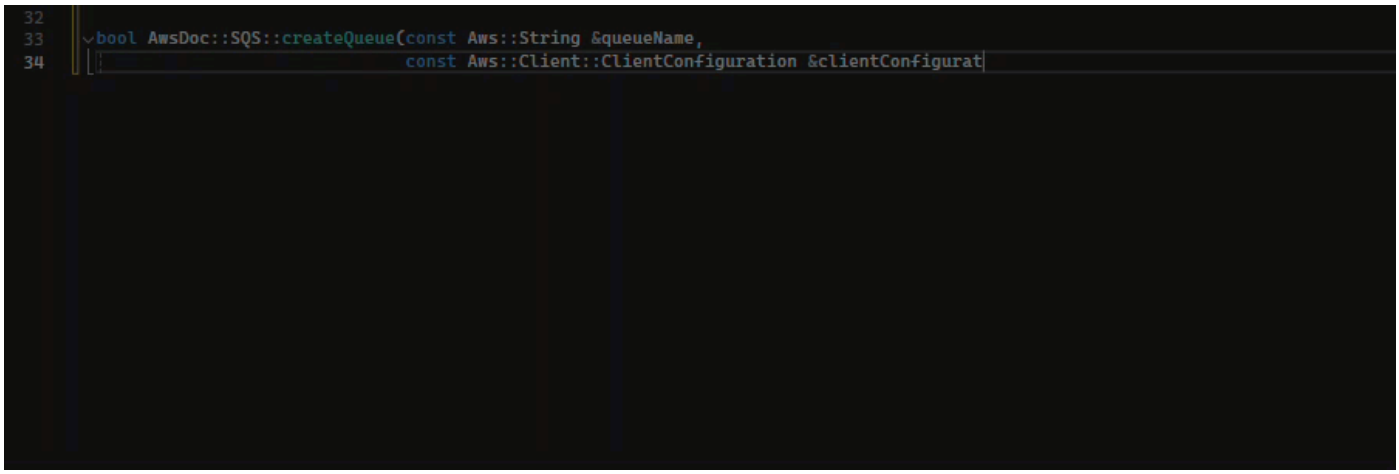
Python

この例では、Amazon Q はデベロッパーのコメントに基づいて 1 行のコードを提案します。

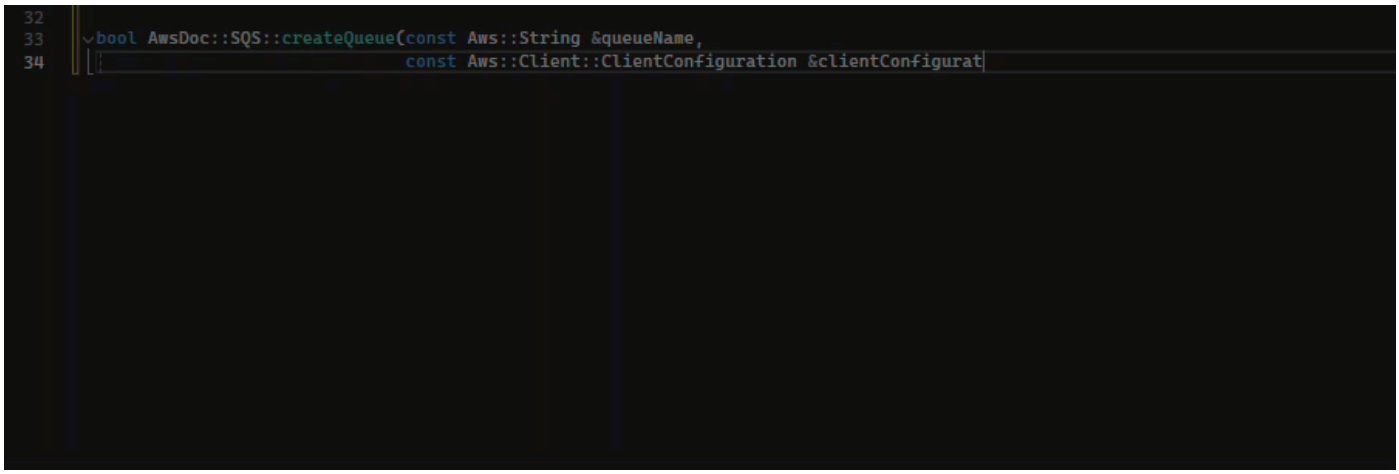
Amazon Q Developer を使用して完全な関数を生成する

Amazon Q は、入力されたコメントに基づいて関数全体を生成できます。コメントを完成させると、Amazon Q は関数の署名を提案します。提案を受け入れると、Amazon Q は自動的にカーソルを関数の次の部分に進めて提案を行います。提案の合間であっても、追加のコメントやコード行を入力すると、Amazon Q は入力に基づいてリファクタリングします。

C

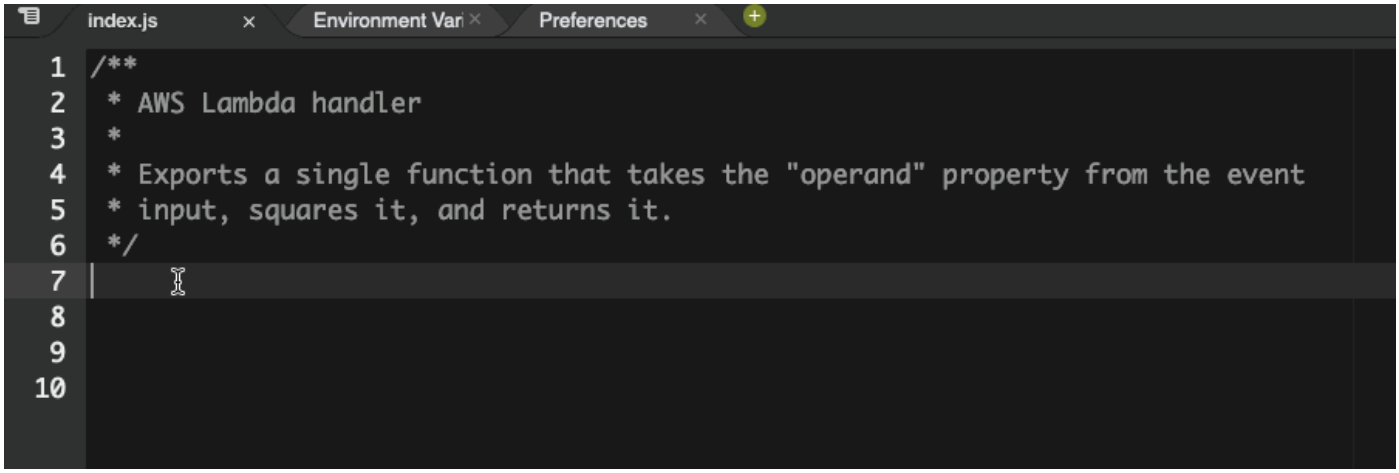


C++

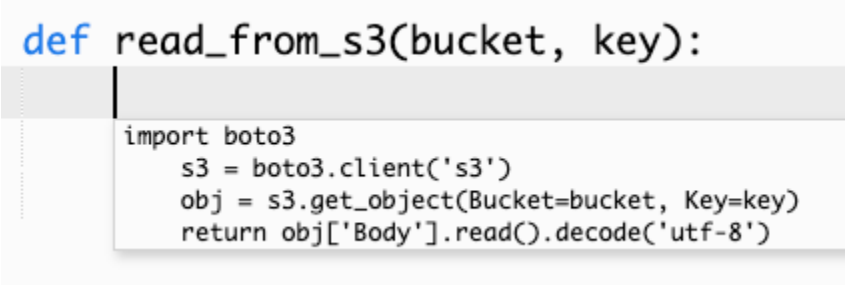


JavaScript

次の例では、ユーザーが一連のコメントに基づいて完全な関数を生成し、編集します。

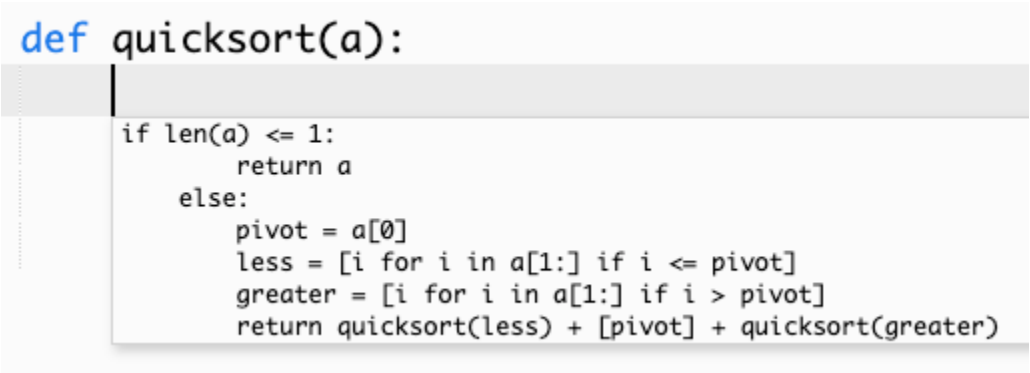


次の画像では、ユーザーが Amazon S3 からファイルを読み取るための関数署名を書いています。その後、Amazon Q は `read_from_s3` メソッドの完全実装を提案します。



> **Note**
>
> 前述の例のように、時々 Amazon Q は提案の一部として `import` ステートメントを含めることがあります。ベストプラクティスとして、手動でこれらの 「`import`」 ステートメントをファイルの先頭に移動します。

別の例として、次の画像では、ユーザーが関数署名を書いています。その後、Amazon Q は `quicksort` メソッドの完全実装を提案します。



Amazon Q は、提案する際に過去のコードスニペットを考慮します。次の画像では、前の例のユーザーが、提案された上記の 「`quicksort`」 の実装を受け入れました。次に、ユーザーはジェネリック 「`sort`」 方法の別の関数署名を書きます。次に、Amazon Q は既に書かれている内容に基づいて実装を提案します。

```
def quicksort(a):
    if len(a) <= 1:
        return a
    else:
        pivot = a[0]
        less = [i for i in a[1:] if i <= pivot]
        greater = [i for i in a[1:] if i > pivot]
        return quicksort(less) + [pivot] + quicksort(greater)

def sort(a):
    return quicksort(a)
```

次の画像では、ユーザーがコメントを書いています。このコメントに基づいて、Amazon Q は関数署名を提案します。

```
# Binary search function
def binary_search(arr, l, r, x):
```

次の画像では、前の例のユーザーが、提案された関数署名を受け入れました。Amazon Q は、その後 `binary_search` 関数の完全な実装を提案できます。

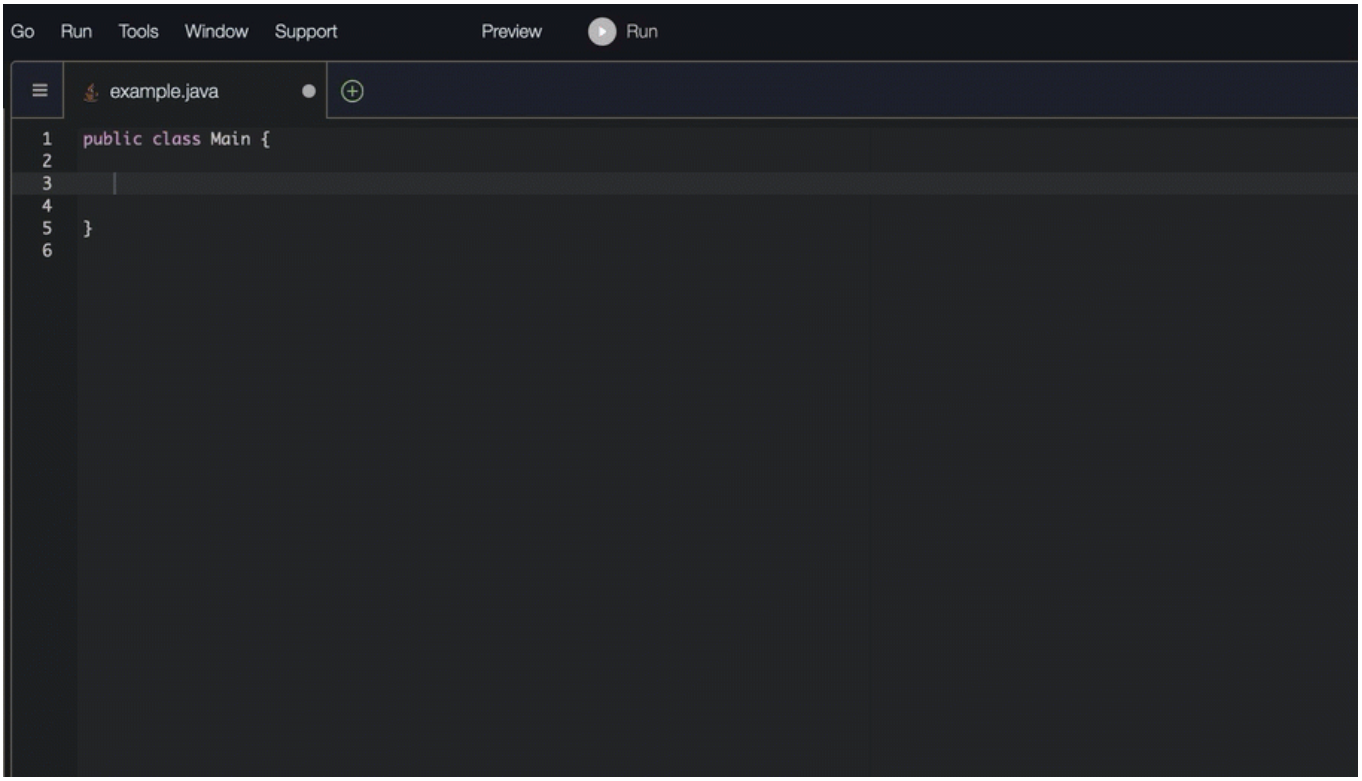
```
# Binary search function
def binary_search(arr, l, r, x):
    while l <= r:
        mid = l + (r - l) // 2
        if arr[mid] == x:
            return mid
        elif arr[mid] < x:
            l = mid + 1
        else:
            r = mid - 1
```

Java

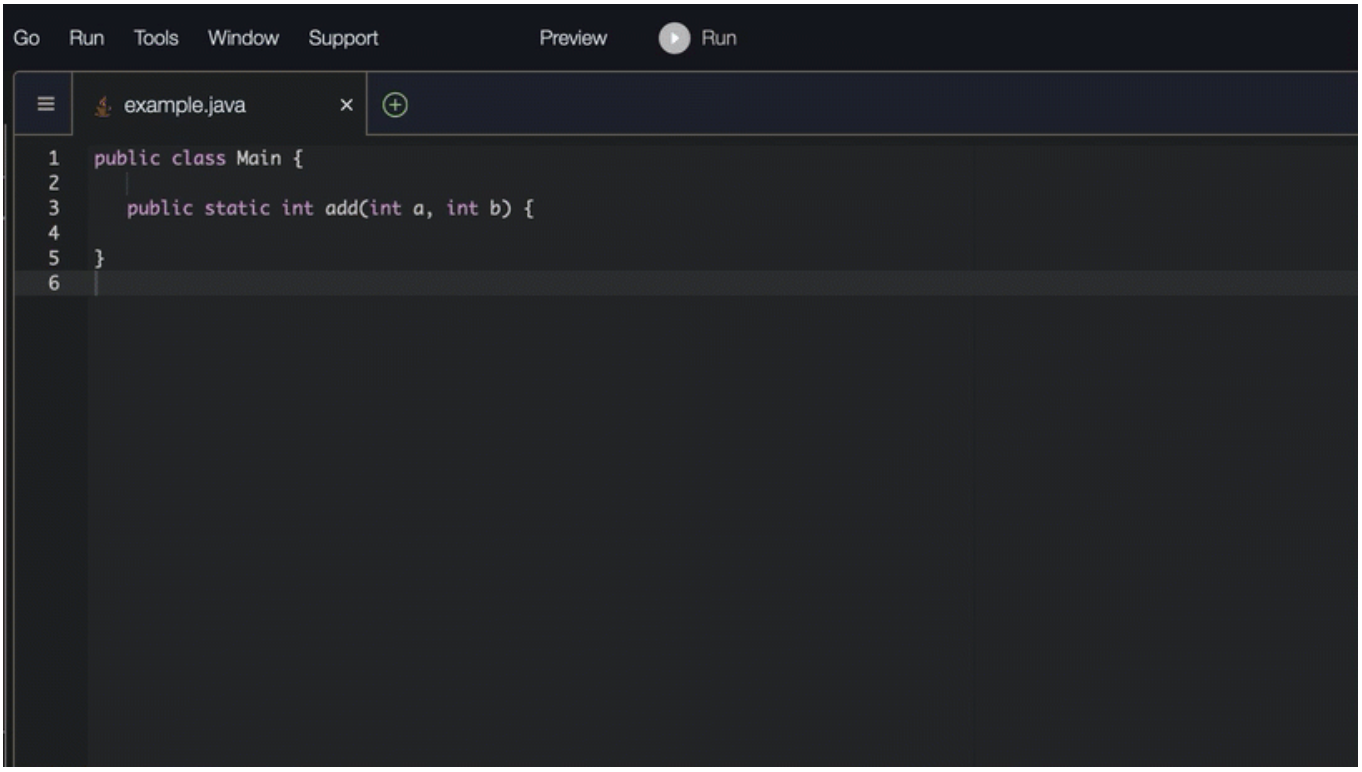
以下のリストでは、Amazon Q がどのように提案を行い、関数を作成するプロセス全体を進めるかについての例を挙げます。

1. 次の例では、ユーザーがコメントを入力します。Amazon Q は関数署名を提案します。

   ユーザーがその提案を受け入れると、Amazon Q は関数本文を提案します。

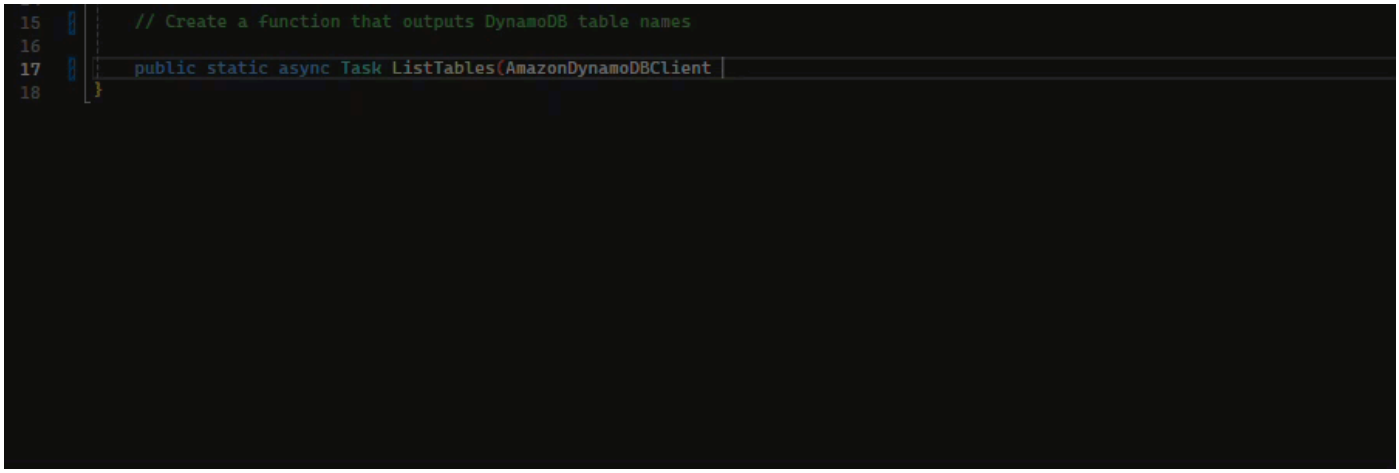


2. 次の図では、Amazon Q からの提案を受け入れる前に、ユーザーが関数の本文にコメントを入力します。次の行で、Amazon Q がコメントに基づいて提案を生成します。



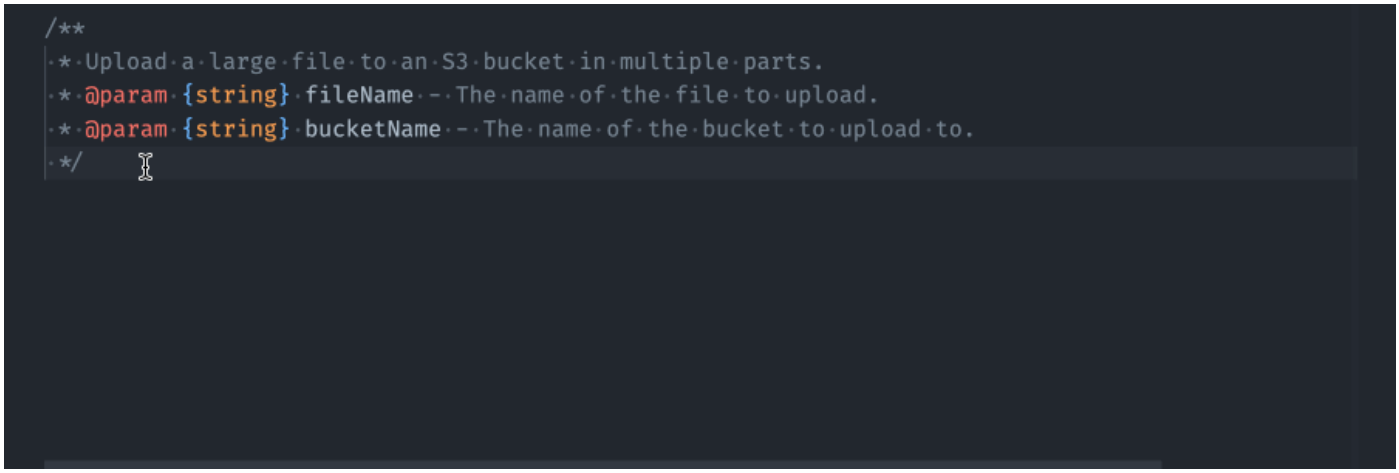
C#

次の例では、Amazon Q は完全な関数を提案しています。



TypeScript

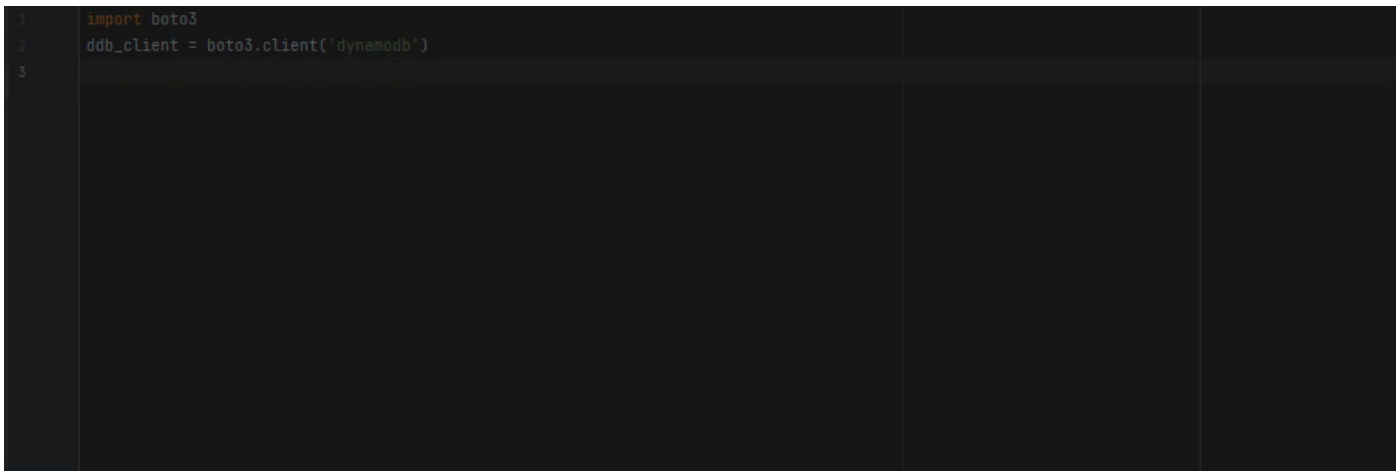
次の例では、Amazon Q はユーザーの docstring に基づいて関数を生成します。



Python

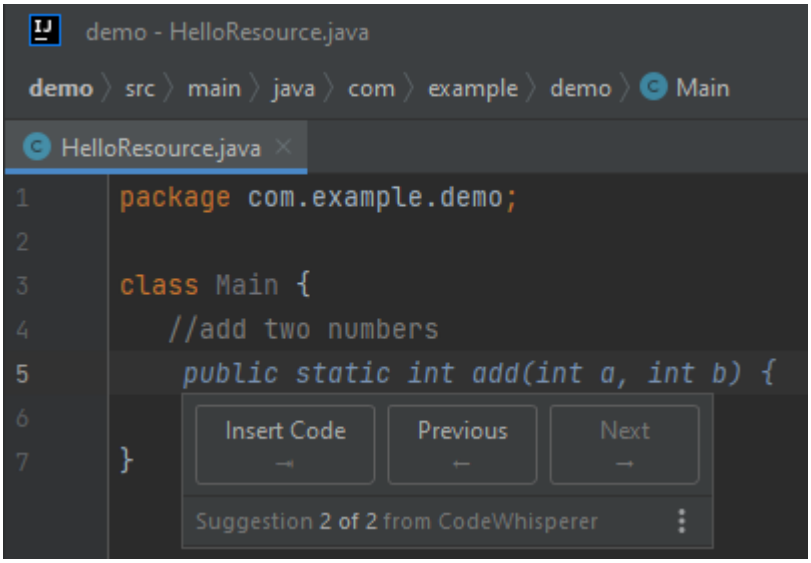
Amazon Q は、入力されたコメントに基づいて関数全体を生成できます。コメントを完成させると、Amazon Q は関数の署名を提案します。提案を受け入れると、Amazon Q は自動的にカーソルを関数の次の部分に進めて提案を行います。提案の合間であっても、追加のコメントやコード行を入力すると、Amazon Q は入力に基づいてリファクタリングします。

次の例では、Amazon Q は完全な関数と対応するユニットテストの両方を生成します。



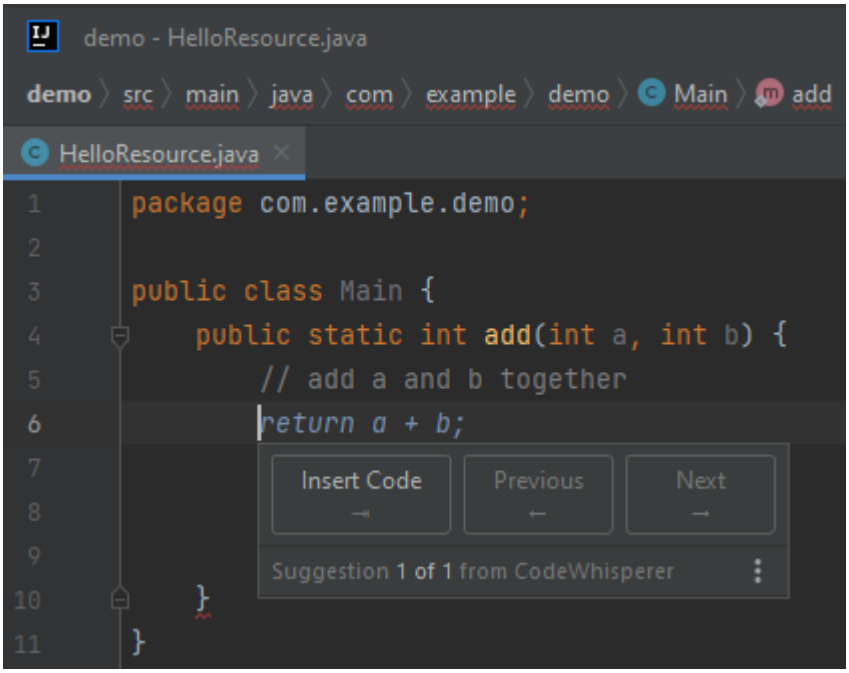
以下のリストでは、Amazon Q がどのように提案を行い、関数を作成するプロセス全体を進めるかについての例を挙げます。

1. 次の画像では、ユーザーがコメントを入力します。コメントの下にある関数署名は、Amazon Q からの提案です。

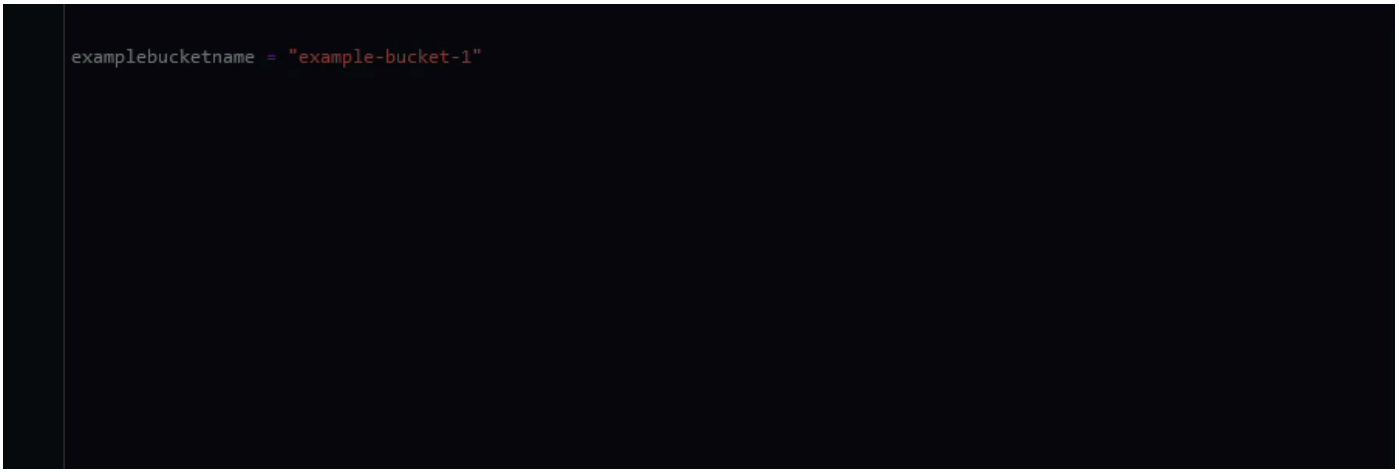


2. 次の画像では、ユーザーが関数署名の Amazon Q 提案を受け入れています。提案を受け入れるとカーソルが自動的に進み、Amazon Q が関数本文について新しい提案を行います。

3. 下の画像では、Amazon Q からの提案を受け入れる前に、ユーザーが関数の本文にコメントを入力します。次の行で、Amazon Q がコメントの内容に基づいて新しい提案を生成しています。



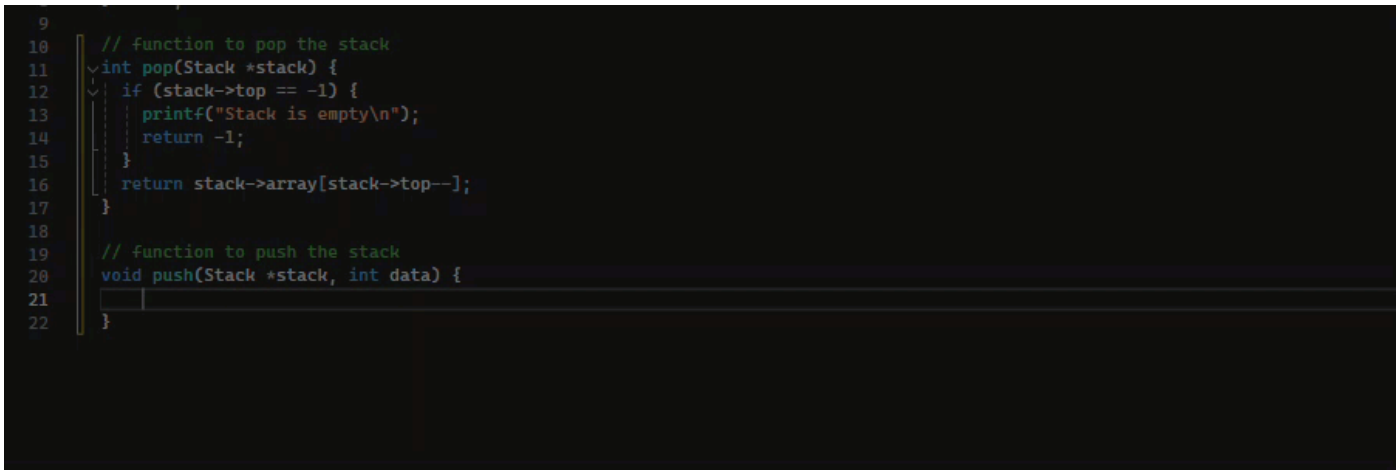
この例では、ユーザーが署名の一部を入力した後、Amazon Q が完全な関数を提案します。



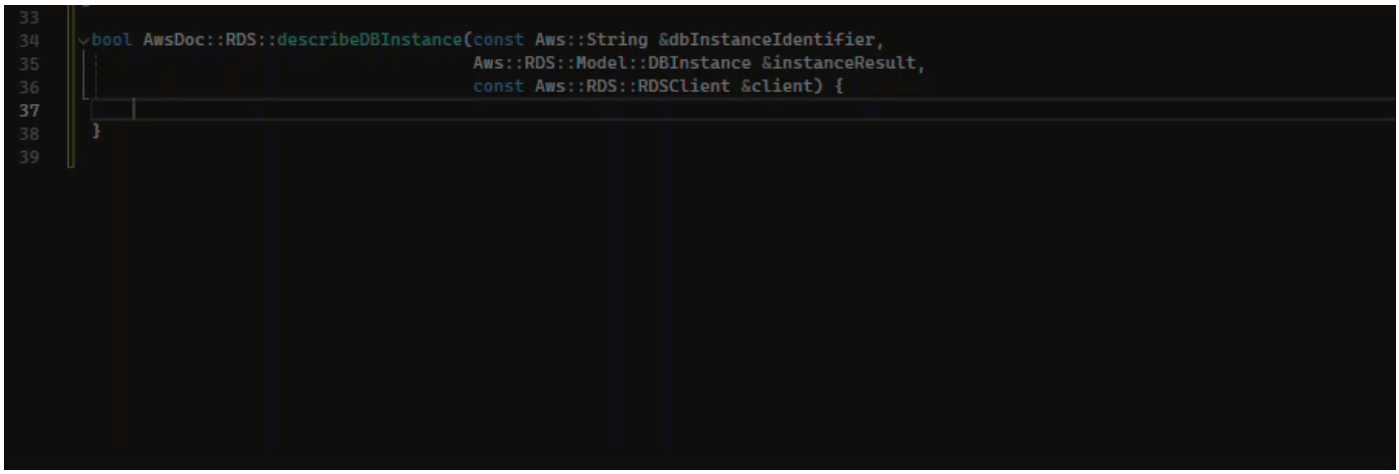
Amazon Q Developer を使用してブロックを補完する

ブロック補完は、`if/for/while/try` コードブロックの補完に使用されます。

C

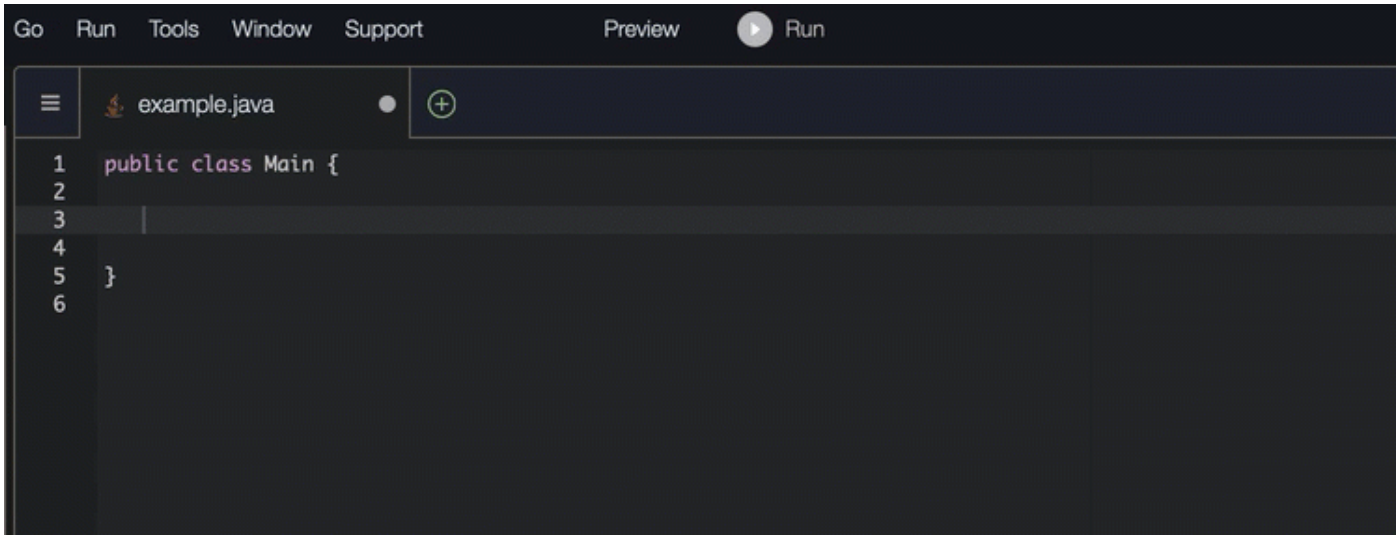


C++



Java

下の例では、ユーザーが `if` ステートメントの署名を入力します。ステートメントの本文は Amazon Q からの提案です。



C#

下の画像では、Amazon Q が関数を完了する方法を提案しています。

TypeScript

下の画像では、Amazon Q が関数を完了する方法を提案しています。

Python

この例では、Amazon Q がコンテキストに基づいてコードのブロックを提案します。

Amazon Q Developer を使用して docstring、JSDoc、Javadoc を補完する

Amazon Q は、コード内でドキュメントを生成または補完するのに役立ちます。

C++

```cpp
/// <summary>
/// This example shows how to attach a policy to an IAM role.
/// </summary>
/// <param name="roleName
bool AwsDoc::IAM::putRolePolicy(
        const Aws::String &roleName,
        const Aws::String &policyName,
        const Aws::String &policyDocument,
        const Aws::Client::ClientConfiguration &clientConfig) {
    Aws::IAM::IAMClient iamClient(clientConfig);
    Aws::IAM::Model::PutRolePolicyRequest request;

    request.SetRoleName(roleName);
    request.SetPolicyName(policyName);
    request.SetPolicyDocument(policyDocument);

    Aws::IAM::Model::PutRolePolicyOutcome outcome = iamClient.PutRolePolicy(request);
    if (!outcome.IsSuccess()) {
        std::cerr << "Error putting policy on role. " <<
                  outcome.GetError().GetMessage() << std::endl;
```

Javascript

この例では、Amazon Q は既存の定数に基づいて JSDoc パラメータを入力します。

```javascript
import {PutObjectCommand, S3Client} from "@aws-sdk/client-s3";

const client = new S3Client({});

/**
 *
 */
export const putObject = async (bucketName, key, body) => {
  const params = {
    Bucket: bucketName,
    Key: key,
    Body: body,
  };
  return client.send(new PutObjectCommand(params));
```

C#

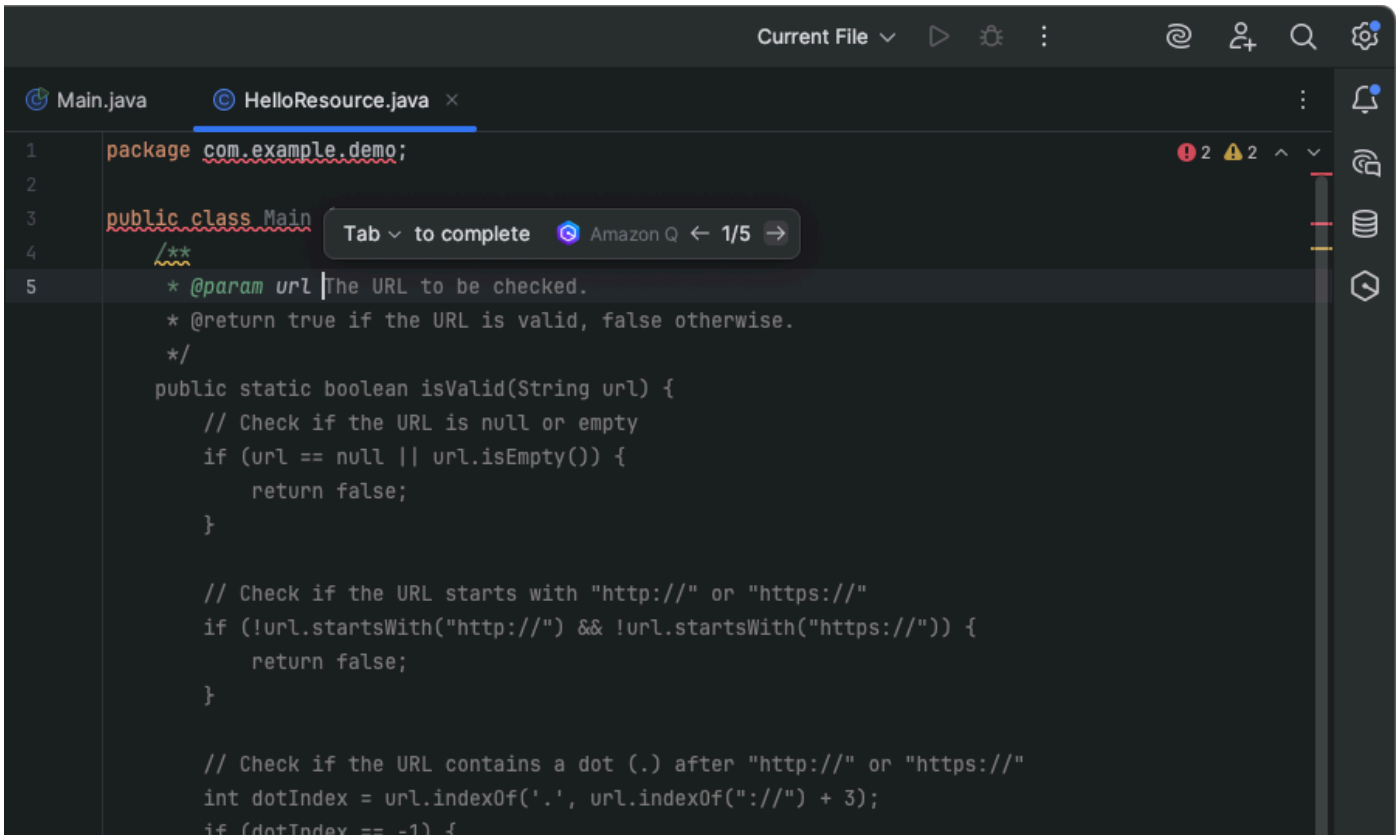
この例では、Amazon Q は既存の定数に基づいて JSDoc パラメータを入力します。

```
    /// <summary>
    /// Shows how to create a new Amazon S3 bucket.
    /// </summary>
    public static async Task<bool> CreateBucketAsync(IAmazonS3 client, string bucketName)
    {
        try
        {
            var request = new PutBucketRequest
            {
                BucketName = bucketName,
                UseClientRegion = true,
            };

            var response = await client.PutBucketAsync(request);
            return response.HttpStatusCode == System.Net.HttpStatusCode.OK;
        }
        catch (AmazonS3Exception ex)
```

Java

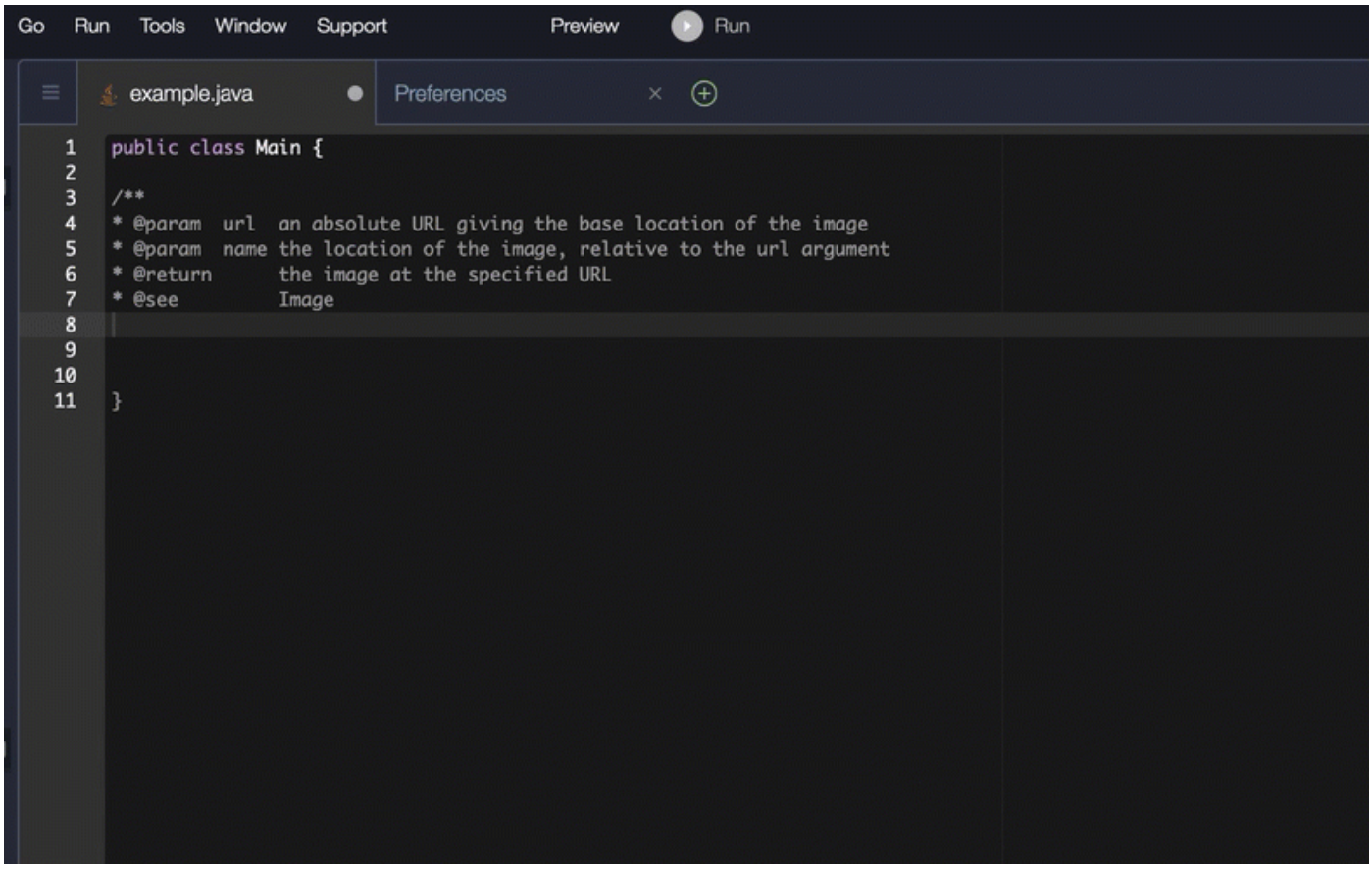
以下の例は、Oracle のウェブサイトの例から引用しています。

以下の図では、ユーザーは docstring の入力を開始しています。Amazon Q は、Docstring に追加する単語を提案しました。



以下の例は、Oracle のウェブサイトの例から引用しています。

以下の Java での例では、ユーザーがドキュメント文字列を入力します。Amazon Q が docstring を処理する関数を提案します。



Python

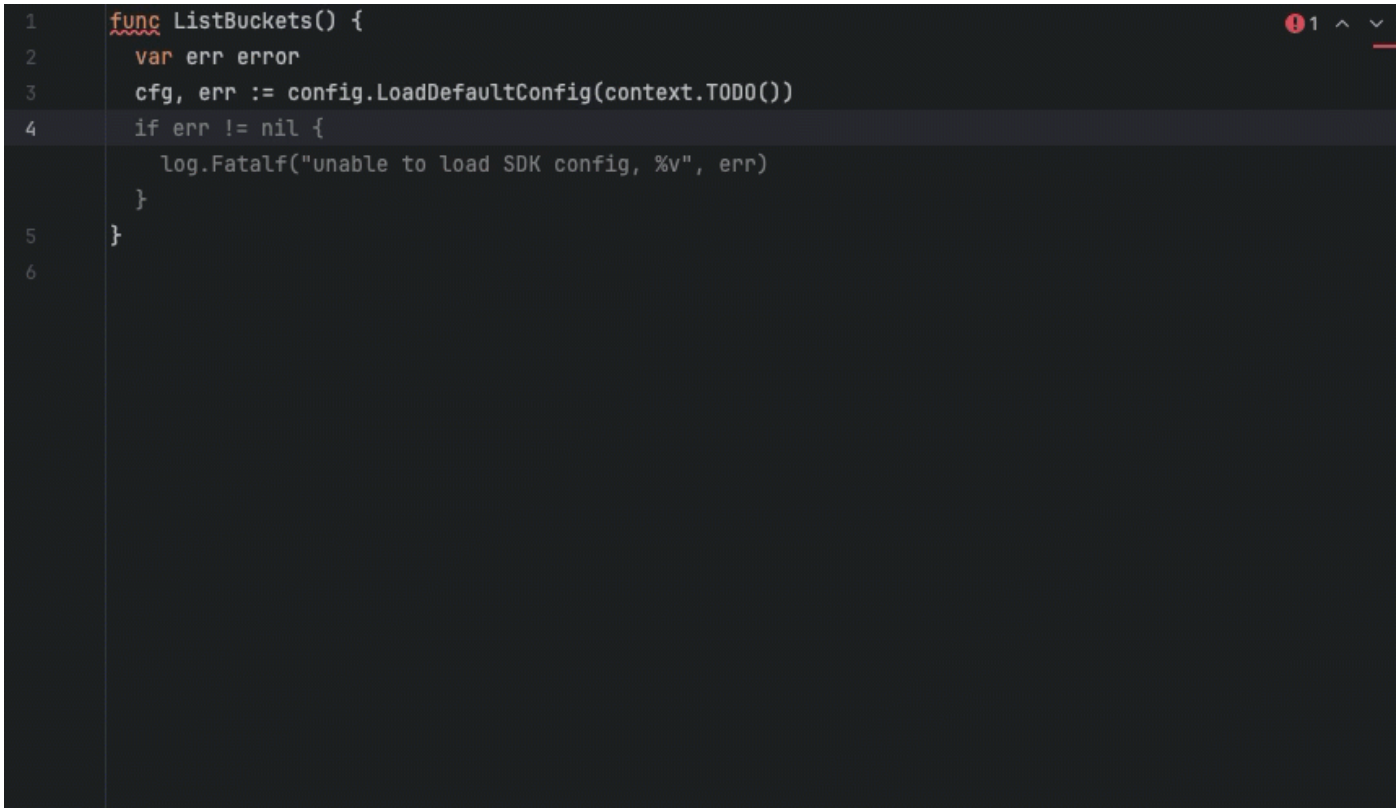
この例では、Amazon Q は周囲のコンテキストに基づいて docstring を提案します。

Amazon Q Developer を使用して行ごとの提案を行う

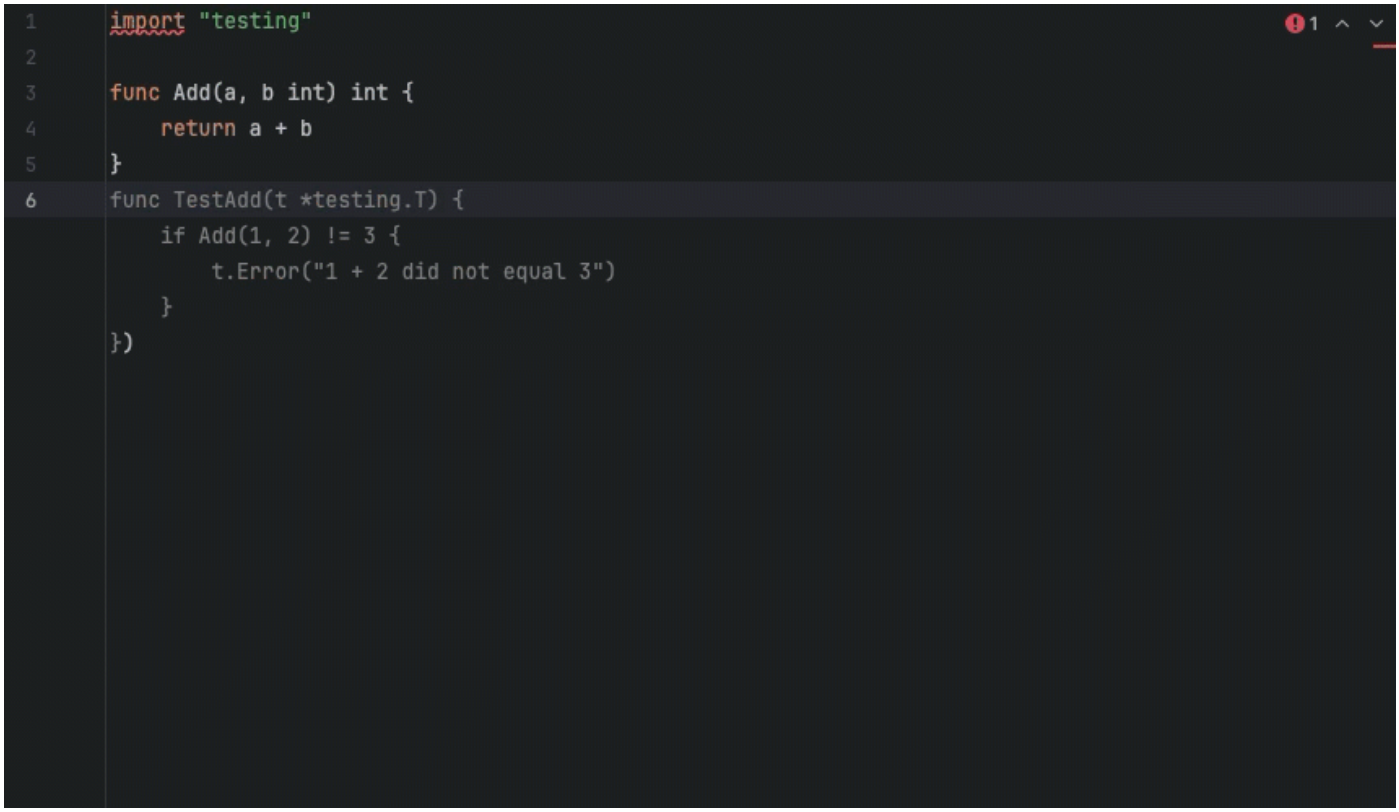
ユースケースにより、Amazon Q は 1 つの提案で関数ブロック全体を生成できない場合があります。そのような場合でも、Amazon Q は行ごとの提案を行うことができます。

Go and GoLand

この例では、Amazon Q は行ごとの提案を行います。

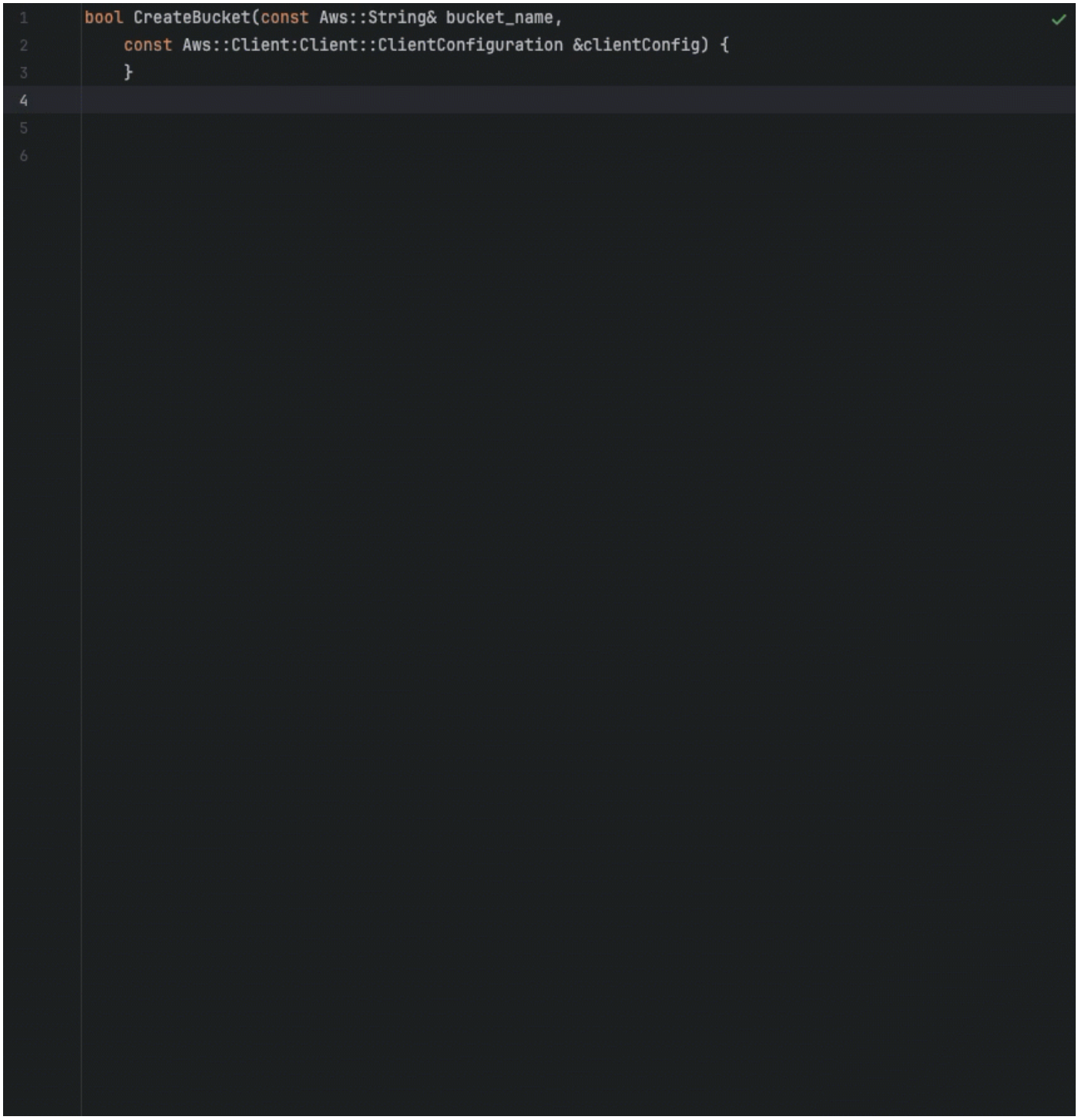


行ごとの提案の別の例を示します。今回はユニットテスト付きです。



C++ and CLion

この例では、Amazon Q は行ごとの提案を行います。



Python

次の画像では、顧客が Amazon CloudWatch Logs グループにメッセージを発行することを示す最初のコメントを書いています。このコンテキストを考えると、次の画像で示すように、Amazon Q は最初の推奨でクライアント初期化コードを提案することしかできません。

```
# Publish a message to a CloudWatch Logs Group

client = boto3.client('logs')
```

ただし、ユーザーが行ごとの提案を要求し続けると、Amazon Q も既に書かれている内容に基づいてコード行を提案し続けます。

```
# Publish a message to a CloudWatch Logs Group
client = boto3.client('logs')
response = client.put_log_events(
    
    logGroupName='VPCFlowLogs',
```

> **Note**
>
> 上記の例では、「VPCFlowLogs」 は正しい定数値ではない可能性があります。Amazon Q が提案を行う際には、必要に応じて必ず定数の名前を変更してください。

次の画像で示すように、Amazon Q は最終的にコードブロック全体を完成させることができます。

```
# Publish a message to a CloudWatch Logs Group
client = boto3.client('logs')
response = client.put_log_events(
    logGroupName='VPCFlowLogs',
    logStreamName='VPCFlowLogs',
    logEvents=[
        {
            'timestamp': int(round(time.time() * 1000)),
            'message': json.dumps(event)
        }
    ]
)
No recommendations
```

この例では、Amazon Q は一度に 1 行ずつ提案を行います。

Amazon Q Developer を使用した IDE でのコードの変換

Amazon Q Developer は、言語およびオペレーティングシステム (OS) レベルの自動アップグレードと変換を実行することで、統合開発環境 (IDEs) でコードを変換できます。変換するコードを指定すると、Amazon Q はファイルを確認して適用できる変更を生成します。

開始するには、変換をサポートする IDE に Amazon Q をインストールします。次に、Amazon Q で実行する変換のタイプに関するトピックを参照してください。

変換をサポートする IDEs「」を参照してくださいIDE での Amazon Q Developer の使用。

トピック

- Amazon Q Developer を使用した Java アプリケーションの変換
- Amazon Q Developer を使用した .NET アプリケーションの変換

Amazon Q Developer を使用した Java アプリケーションの変換

Amazon Q は、Java アプリケーションの次のタイプの変換をサポートしています。

- Java 言語と依存関係のバージョンのアップグレード
- Oracle から PostgreSQL データベースへの移行用の埋め込み SQL 変換

開始するには、実行する変換のタイプに関するトピックを参照してください。

トピック

- クォータ

- Amazon Q Developer を使用した Java バージョンのアップグレード
- Amazon Q Developer を使用した Java アプリケーションの埋め込み SQL の変換
- Amazon Q Developer によるコマンドラインでのコードの変換
- 変換ジョブ履歴の表示
- Java 変換に関する問題のトラブルシューティング

クォータ

IDE およびコマンドラインでの Amazon Q を使用した Java アプリケーション変換では、次のクォータが維持されます。

- ジョブあたりのコード行 – Amazon Q が特定の変換ジョブで変換できるコード行の最大数。
- 1 か月あたりのコード行数 — Amazon Q が 1 か月に変換できるコード行の最大数。
- 同時ジョブ – 同時に実行できる変換ジョブの最大数。このクォータは、Visual Studio の .NET 変換を含む、IDE 内のすべての変換に適用されます。
- 1 か月あたりのジョブ – 1 か月に実行できる変換ジョブの最大数。

リソース	クォータ
ジョブあたりのコード行	無料利用枠: 1000 行のコード
1 か月あたりのコード行	無料利用枠: 2000 行のコード
同時ジョブ	ユーザーあたり 1 ジョブ AWS アカウントあたり 25 ジョブ
1 か月あたりのジョブ数	Pro 階層: 1000 ジョブ 無料利用枠: 100 ジョブ

Amazon Q Developer を使用した Java バージョンのアップグレード

Amazon Q Developer は、統合開発環境 (IDE) で Java アプリケーションを新しい言語バージョンにアップグレードできます。Amazon Q がコードをアップグレードするために実行できる変更には、廃

止されたコードコンポーネントと APIs の更新、コード内のライブラリ、フレームワーク、その他の依存関係のアップグレードが含まれます。

コードを変換するために、Amazon Q はまずソース言語バージョンでコードを構築し、変換の実行に必要な情報があることを確認します。Amazon Q がコードを正常に変換したら、IDE で変更を検証して受け入れます。Amazon Q Developer は、アップグレードされたコードをターゲット JDK と互換性を持たせるために必要な最小限の変更を行うため、プロジェクトのライブラリと依存関係をアップグレードするには追加の変換が必要です。Amazon Q がコードを変換する方法の詳細については、「」を参照してくださいAmazon Q Developer が Java 言語アップグレードのコードを変換する方法。

トピック

- サポートされている Java のアップグレードと IDEs
- ステップ 1: 前提条件
- ステップ 2: プロジェクトを設定する
- ステップ 3: 依存関係のアップグレードファイルを作成する (オプション)
- ステップ 4: コードを変換する
- Amazon Q Developer が Java 言語アップグレードのコードを変換する方法

サポートされている Java のアップグレードと IDEs

Amazon Q は現在、変換用に次の Java ソースコードバージョンとターゲットバージョンをサポートしています。コードを同じ Java バージョンに変換するには、ソースコードバージョンのライブラリやその他の依存関係のアップグレードが含まれます。

サポートされている Java のアップグレード

ソースコードのバージョン	サポートされているターゲットバージョン
Java 8	Java 17 および Java 21
Java 11	Java 17 および Java 21
Java 17	Java 17 および Java 21
Java 21	Java 21

Amazon Q は、次の IDEs で Java アップグレードをサポートしています。

- IDEs JetBrains のモジュール
- のプロジェクトとワークスペース Visual Studio Code

ステップ 1: 前提条件

続行する前に、「IDE で Amazon Q のセットアップ」の手順が完了していることを確認してください。

コード変換ジョブを開始する前に、以下の前提条件を満たしていることを確認します。

- プロジェクトはサポートされているJavaバージョンで記述され、 上に構築されますMaven。
- プロジェクトは IDE Mavenで を使用して正常に構築されます。 Maven 3.8 以降は現在サポートされています。
- プロジェクトソース JDK がローカルで利用可能で、ソースコードのバージョンである。例えば、Java 8 コードを変換する場合、ローカルの JDK インストールは JDK 8 である必要があります。
- プロジェクトのビルドが 55 分以内に完了する。
- プロジェクトが正しく設定され、正しい JDK バージョンが指定されている。詳細については、「ステップ 2: プロジェクトを設定する」を参照してください。
- プロジェクトが仮想プライベートクラウド (VPC) やオンプレミスネットワークなど、プライベートネットワーク上のリソースへのアクセスを必要としない。例えば、プロジェクトにネットワーク内のデータベースに接続するユニットテストが含まれている場合、変換は失敗します。
- プロジェクトが Java プロジェクトで Java 以外の言語をパッケージ化するプラグインを使用していない。例えば、Java ソースコードに加えて、フロントエンド JavaScript コードを実行するために frontend-maven-plugin を使用している場合、変換は失敗します。
- ローカルネットワークで、Amazon Q がコードの変換に使用する Amazon S3 バケットへのアップロードが許可されている。詳細については、「Allow access to Amazon S3 buckets in data perimeters」を参照してください。

ステップ 2: プロジェクトを設定する

プロジェクトを設定するには、使用している IDE に応じて次の情報を使用します。

JetBrains でプロジェクトを設定する

JetBrains でプロジェクトを設定するには、次のプロジェクト設定とモジュール設定を指定する必要があります。

モジュールがプロジェクトと同じ JDK と言語レベルを使用している場合は、モジュール設定を更新する必要はありません。

- Project SDK – プロジェクトのコンパイルに使用される JDK。
- Project language level – プロジェクトで使用される Java バージョン。
- Module SDK – モジュールをコンパイルするために使用される JDK。
- Module language level – モジュールで使用される Java バージョン。
- Maven Runner JRE – モジュールのビルドに使用する JDK。

プロジェクト設定とモジュールの設定を更新する

プロジェクトまたはモジュールの SDK と言語レベルの設定を更新する手順は次のとおりです。

1. JetBrains IDE で、[File] を選択し、次に [Project Structure] を選択します。
2. [Project Structure] ウィンドウが開きます。[Project Settings] で [Modules] を選択します。
    a. プロジェクト JDK を更新するには、[SDK] の横にあるドロップダウンリストから選択します。
    b. プロジェクト言語を更新するには、[Language level] の横にあるドロップダウンから選択します。
3. [Project Settings] で [Modules] を選択します。
    a. モジュール JDK を更新するには、[SDK] の横にあるドロップダウンリストから選択します。
    b. モジュール言語を更新するには、[Language level] の横にあるドロップダウンから選択します。

詳細については、JetBrains ドキュメントの「Project structure settings」と「Module structure settings」を参照してください。

Maven 設定を更新する

Maven Runner JRE を更新するには、次の手順を実行します。

1. JetBrains IDE で歯車アイコンをクリックし、表示されるメニューで [Settings] を選択します。
2. [Settings] ウィンドウで、[Build, Execution, Deployment]、[Build Tools]、[Maven]、[Runner] の順に選択します。

3. JRE フィールドで、変換するモジュールの構築に使用する JDK を選択します。

VS Code でプロジェクトを設定する

VS Code でプロジェクトを設定するには、プロジェクトに以下が含まれている必要があります。

- プロジェクトルートフォルダの `pom.xml` ファイル
- プロジェクトディレクトリの `.java` ファイル

プロジェクトに Maven ラッパー実行可能ファイル (macOS の場合は `mvnw`、Windows の場合は `mvnw.cmd`) が含まれている場合は、それがプロジェクトのルートにあることを確認してください。Amazon Q はこのラッパーを使用します。他の Maven の設定は必要ありません。

Maven ラッパーを使用しない場合は、Maven をインストールします。詳細については、Apache Maven ドキュメントの「Installing Apache Maven」を参照してください。

Maven をインストールしたら、PATH 変数に追加します。詳細については、「Maven を PATH に追加するにはどうすればよいですか? Your Java `runtime` variable should also be pointing to a JDK and not to a JRE」を参照してください。設定が正しいことを確認するには、`mvn -v` を実行します。出力には、Maven のバージョンと、JDK へのパスを指す `runtime` 変数が表示されます。

ステップ 3: 依存関係のアップグレードファイルを作成する (オプション)

Amazon Q には、依存関係のアップグレードファイル、プロジェクトの依存関係を一覧表示する YAML ファイル、および変換中にアップグレードするバージョンを提供できます。依存関係のアップグレードファイルを提供することで、Amazon Q がアップグレードについて認識していないサードパーティーとサードパーティーの依存関係を指定できます。

ファーストパーティの依存関係とは、組織が維持し、ローカルまたは組織のプライベートネットワークでのみ利用できるライブラリ、プラグイン、フレームワークを指します。Amazon Q は、ローカル環境でビルドを実行するときに、ファーストパーティの依存関係にアクセスできます。詳細については、「ローカル環境でのコードの構築」を参照してください。サードパーティーの依存関係は、公開されているか、組織に固有ではないオープンソースの依存関係です。

YAML ファイルでアップグレードするファーストパーティの依存関係を指定できます。Amazon Q は JDK アップグレード中にそれらをアップグレードします (Java 8 から 17 など) 。最初の JDK アップグレード後に別の変換 (17 から 17 または 21 から 21) を開始して、サードパーティーの依存関係をアップグレードできます。

Amazon Q が最小限の JDK アップグレードを実行したら、別の変換を開始して、すべてのサードパーティーの依存関係をアップグレードできます。または、YAML ファイルでサードパーティーの依存関係とそのバージョンを指定して、ライブラリのアップグレード変換中にのみそれらの依存関係をアップグレードすることもできます。

Amazon Q では、変換中に依存関係のアップグレードファイルを指定するように求められます。指定する場合は、まず ファイルが正しく設定されていることを確認してください。YAML ファイルには次のフィールドが必要です。

- name - 依存関係のアップグレードファイルの名前。
- description (オプション) - 依存関係のアップグレードファイルと変換の説明。
- dependencyManagement - アップグレードする依存関係とプラグインのリストが含まれます。
- dependencies - アップグレードするライブラリの名前とバージョンが含まれます。
- plugins - アップグレードするプラグインの名前とバージョンが含まれます。
- identifier - ライブラリ、プラグイン、またはその他の依存関係の名前。
- targetVersion - アップグレード先の依存関係のバージョン。
- versionProperty (オプション) - アプリケーションの `pom.xml` ファイルの `properties` タグで設定されている、定義する依存関係のバージョン。
- originType - FIRST_PARTY または Third_PARTY のいずれかで指定された、依存関係がファーストパーティかサードパーティーか。

以下は、依存関係のアップグレード YAML ファイルの例と、Amazon Q が解析するために必要な設定です。

```
name: dependency-upgrade

description: "Custom dependency version management for Java migration from JDK 8/11/17
 to JDK 17/21"


dependencyManagement:

  dependencies:

    - identifier: "com.example:library1"

      targetVersion: "2.1.0"
```

```
      versionProperty: "library1.version"  # Optional
      originType: "FIRST_PARTY"
    - identifier: "com.example:library2"
      targetVersion: "3.0.0"
      originType: "THIRD_PARTY"
  plugins:
    - identifier: "com.example.plugin"
      targetVersion: "1.2.0"
      versionProperty: "plugin.version"  # Optional
      originType: "THIRD_PARTY"
```

ステップ 4: コードを変換する

独自のコードを変換する前に、サンプルプロジェクトを変換して IDE が正しくセットアップされていることをテストすることをおすすめします。コード変換に使える GitHub プロジェクトの例が https://github.com/aws-samples/aws-appconfig-java-sample にあります。

IDE のセットアップをテストするには、サンプルプロジェクトをダウンロードして解凍し、IDE の次の手順を実行します。提案された変更と変換の概要を表示できる場合は、独自のコードプロジェクトを変換する準備ができています。変換が失敗するということは、IDE が正しく設定されていません。設定の問題に対処するには、「ステップ 2: プロジェクトを設定する」と「トラブルシューティング」を確認します。

> **Note**
>
> コード変換中は、ローカルマシンをオフにしたり、閉じたり、スリープ状態にしたりしないでください。初期ビルドと検証ビルドでは、安定したネットワーク接続を必要とするクライアント側の環境を使用します。

コードプロジェクトまたはコードモジュールの言語バージョンをアップグレードするには、以下のステップを実行します。

JetBrains

1. JetBrains でアップグレードするモジュールを開きます。IDE でプロジェクトが正常にビルドされていることを確認します。
2. Amazon Q ロゴを選択し、開いた Amazon Q チャットパネルに **/transform** と入力します。
3. [アプリケーションの変換] ポップアップが表示されます。ドロップダウンリストからアップグレードするプロジェクトを選択し、[変換] を選択します。
4. Amazon Q では、アップグレード依存関係ファイルを指定するように求められます。アップグレード先の依存関係とバージョンで YAML を設定している場合は、 ファイルを追加します。Amazon Q はファイルを検証して、正しく設定されていることを確認します。エラーが発生した場合は、「」で説明されている形式と必須フィールドを確認してくださいステップ 3: 依存関係のアップグレードファイルを作成する (オプション)。
5. Amazon Q が変換を開始します。進捗状況は、[変換の詳細] タブで確認できます。
6. 変換が完了したら、プロジェクトを更新する前にアップグレードされたコードを確認できます。新しいコードを表示するには、[変換の詳細] タブに移動し、[差分の表示] を選択します。表示される [パッチを適用] ウィンドウでファイルを選択して、ソースコードとアップグレードされたコードを含む差分ビューを開きます。
7. Amazon Q が加えた変更を受け入れるには、[差分の表示] を選択して [パッチを適用] ウィンドウを開きます。更新されたファイルをすべて選択し、[OK] を選択してプロジェクトを更新します。
8. コードのアップグレード方法の詳細と次のステップの提案を確認するには、[変換の詳細] タブで [変換の概要を表示] を選択します。

Visual Studio Code

1. VS Code でアップグレードするプロジェクトまたはワークスペースを開きます。プロジェクトが IDE で正常にビルドされていることを確認します。
2. Amazon Q ロゴを選択し、開いた Amazon Q チャットパネルに **/transform** と入力します。
3. IDE の上部にある検索バーからアップグレードするプロジェクトを選択します。
4. Amazon Q でソースコードのバージョンが見つからない場合は、コードバージョンを選択するように求められます。ソースコードが書き込まれるバージョンを選択し、ポップアップで [変換] を選択して続行します。

5. プロンプトが表示されたら、JDK への JAVA_HOME パスを入力します。プロジェクトの設定の詳細については、「VS Code プロジェクトを設定する」を参照してください。
6. Amazon Q では、アップグレード依存関係ファイルを指定するように求められます。アップグレード先の依存関係とバージョンで YAML を設定している場合は、 ファイルを追加します。Amazon Q はファイルを検証して、正しく設定されていることを確認します。エラーが発生した場合は、「」で説明されている形式と必須フィールドを確認してくださいステップ 3: 依存関係のアップグレードファイルを作成する (オプション)。
7. Amazon Q が変換を開始します。[変換ハブ] タブで進捗状況を確認できます。
8. 変換が完了すると、[変更案] タブが開きます。プロジェクトを更新する前にアップグレードされたコードを確認するには、[変更案をダウンロード] を選択します。ファイルを選択して、ソースコードとアップグレードされたコードを含む差分ビューを開きます。
9. Amazon Q の変更を受け入れるには、[変更案] タブに移動し、[受け入れる] を選択します。
10. コードのアップグレード方法の詳細と次のステップの提案を確認するには、[変換ハブ] で [ビューおよびその他のアクション] 省略記号ボタンをクリックし、[変換の概要を表示] を選択します。

Amazon Q Developer が Java 言語アップグレードのコードを変換する方法

コードを変換するために、Amazon Q Developer はプロジェクトのコード言語バージョンのアップグレードに使用する変換プランを生成します。コードを変換すると、変換の概要とファイルの差分が表示され、変更を受け入れる前に内容を確認できます。Amazon Q Developer は、アップグレードされたコードをターゲット JDK と互換性を持たせるために必要な最小限の変更を行うため、プロジェクトのライブラリと依存関係をアップグレードするには追加の変換が必要です。以下のセクションでは、Amazon Q が変換を実行する方法について詳しく説明します。

コードをビルドし、変換プランを作成する

コードの変換を開始するために、Amazon Q はプロジェクトをローカルでビルドし、ソースコード、プロジェクトの依存関係、ビルドログを含むビルドアーティファクトを生成します。

ビルドアーティファクトを生成した後、Amazon Q は安全なビルド環境にコードをビルドし、アップグレードするプロジェクトまたはモジュールに合わせてカスタマイズされた変換プランを作成します。変換プランでは、Amazon Q が行おうとしている特定の変更点の概要が説明されます。これには、新しい依存関係バージョン、主要なコード変更、廃止されたコードの代替案が含まれます。こうした変更は、コードの予備ビルドに基づいており、変換中に変更される可能性があります。

コードを変換する

コードを変換するために、Amazon Q は変換プランで提案された変更に基づいてコードをターゲット Java バージョンにアップグレードしようとします。変更を行う際には、ソースコード内の既存のユニットテストを再ビルドして実行し、発生したエラーを繰り返し修正します。JDK のアップグレードは、次のソースコードバージョンからターゲットバージョンに行うことができます。

- Java 8 ~ 17
- Java 8 ~ 21
- Java 11 ~ 17
- Java 11 ~ 21
- Java 17 ~ 21

Amazon Q は、コードをターゲットの Java バージョンと互換性を持たせるために必要な最小限の変更を行います。Java 17 または 21 に正常にアップグレードした後。Amazon Q が最小限の JDK アップグレードを実行したら、別の変換を開始して、すべてのサードパーティーの依存関係をアップグレードできます。または、YAML ファイルでサードパーティーの依存関係とそのバージョンを指定して、ライブラリのアップグレード変換中にのみそれらの依存関係をアップグレードすることもできます。

Amazon Q は、コードをアップグレードする際に次の変更を試みます。

- ターゲット Java バージョンの推奨事項に従って、非推奨のコードコンポーネントを更新する
- 一般的なライブラリとフレームワークを、ターゲットの Java バージョンと互換性のあるバージョンにアップグレードします。これには、次のライブラリとフレームワークを最新の利用可能なメジャーバージョンに更新することが含まれます。
  - Apache Commons IO
  - Apache HttpClient
  - bc-fips
  - Cucumber-JVM
  - 休止
  - jackson-annotations
  - JakartaEE
  - Javax
  - javax.servlet

- jaxb-api
- jaxb-impl
- jaxen
- jcl-over-slf4j
- json-simple
- jsr305
- junit
- junit-jupiter-api
- Log4j
- Micronaut
- Mockito
- mockito-core
- Okio
- PowerMockito
- Quarkus
- slf4j
- slf4j-api
- Spring Boot
- Spring Framework
- Spring Security
- Swagger
- testng

> **Note**
>
> クライアント側のビルドでは安定したネットワーク接続が必要であるため、コード変換中にローカルマシンをオフまたは閉じないでください。

ローカル環境でのコードの構築

変換中、Amazon Q はローカル環境で検証ビルドを実行します。Amazon Q は、サーバー側のコードを複数のステップで変換します。各ステップの後、Amazon Q はローカル環境にコードを送信し、変更をビルドしてテストします。その後、コードはサーバー側に送信され、変換が続行されます。

ローカル環境のビルドは、Amazon Q がプライベートリソースへのアクセスを必要とするテストを実行できるようにすることで、変換されたコードを検証するのに役立ちます。ローカル環境で AI が生成するコードの構築に関連するセキュリティリスクを最小限に抑えるために、Amazon Q はセキュリティ上の懸念に対応するために生成するコードを確認して更新します。

変換の概要を確認し、変更を受け入れる

変換が完了すると、Amazon Q は、プロジェクト全体がアップグレードされたかどうかを示す最終ビルドのステータスなど、変更の詳細を含む変換の概要を用意します。ビルドログの概要を確認して、Amazon Q がアップグレードされたバージョンでコードをビルドできない原因となった問題を解明することもできます。

変換の概要には、変換プランの提案された変更とコードのアップグレードのために最終的に Amazon Q が行った変更の違い、元のプランになかった追加の変更についても記載されています。

変換の概要を確認したら、Amazon Q が提案している変更をファイル差分ビューで表示できます。Amazon Q が提案するコード変更は、変更を受け入れるまで現在のプロジェクトファイルには影響しません。変換されたコードは、変換が完了してから最大 30 日間使用できます。

部分的に成功した変換を完了する

コードベースの複雑さと詳細によっては、変換が部分的に成功する場合があります。これは、Amazon Q がプロジェクト内の特定のファイルやコード領域のみを変換できたことを意味します。この場合、更新された言語バージョンでビルドできるように、プロジェクトの残りのコードを手動で更新する必要があります。

残りのコードを変換するには、IDE で Amazon Q チャットを使用します。Amazon Q に部分的に更新されたファイルを確認し、コンパイルエラーなどの問題に対処するための新しいコードを提供するようにリクエストします。また、`/dev` や `@workspace` などの機能を使用して、コンテキストとして含めるプロジェクトを増やし、一度に複数のファイルの提案を取得することもできます。

Amazon Q Developer を使用した Java アプリケーションの埋め込み SQL の変換

IDE でのコード変換用の Amazon Q Developer エージェントは、埋め込み SQL を変換して AWS Database Migration Service（ AWS DMS) で Oracle から PostgreSQL データベースへの移行を完了するのに役立ちます。

AWS DMS は、リレーショナルデータベース、データウェアハウス、NoSQL データベース、およびその他のタイプのデータストアを移行できるようにするクラウドサービスです。DMS の AWS DMS Schema Conversion は、ターゲットデータベースに適用できるデータベーススキーマとコードオブジェクトを変換するのに役立ちます。詳細については、「 AWS Database Migration Service ユーザーガイド」の「What is AWS Database Migration Service ? 」を参照してください。

AWS DMS および DMS Schema Conversion を使用してデータベースを移行する場合、アプリケーションに埋め込まれた SQL をターゲットデータベースと互換性があるように変換する必要がある場合があります。手動で変換するのではなく、IDE で Amazon Q を使用して変換を自動化できます。Amazon Q は、DMS Schema Conversion のメタデータを使用して、アプリケーションの埋め込み SQL をターゲットデータベースと互換性のあるバージョンに変換します。

現在、Amazon Q は PostgreSQL に移行する Oracle データベースの Java アプリケーションの SQL を変換できます。アプリケーションに Oracle SQL ステートメントが含まれている場合にのみ、IDE で SQL コードを変換するオプションが表示されます。詳細については、前提条件を参照してください。

ステップ 1: 前提条件

続行する前に、「IDE で Amazon Q のセットアップ」の手順が完了していることを確認してください。

SQL 変換のコード変換ジョブを開始する前に、次の前提条件を満たしていることを確認してください。

- SQL が埋め込まれた Java アプリケーションを Oracle データベースから PostgreSQL データベースに移行します。変換の対象となるには、アプリケーションに Oracle SQL ステートメントが含まれている必要があります。
- DMS Schema Conversion を使用して AWS データベーススキーマを変換するプロセスが完了しました。詳細については、「 Database Migration Guide」の「Migrating Oracle database to Amazon RDS for PostgreSQL with DMS Schema Conversion」を参照してください。
- スキーマ変換が完了したら、DMS AWS コンソールから移行プロジェクトファイルをダウンロードします。

ステップ 2: アプリケーションを設定する

埋め込み SQL コードを変換するには、Java プロジェクトに少なくとも 1 つの.javaファイルが含まれている必要があります。

JetBrains IDE を使用している場合は、プロジェクト構造設定の SDK フィールドを該当する JDK に設定する必要があります。プロジェクト構造の設定については、JetBrains ドキュメントの「プロジェクト構造の設定」を参照してください。

ステップ 3: 埋め込み SQL を変換する

Java アプリケーションの埋め込み SQL コードを PostgreSQL ターゲットデータベースと互換性のある形式に変換するには、次の手順を実行します。

1. Amazon Q がインストールされている IDE で、変換する必要がある埋め込み SQL を含む Java コードベースを開きます。
2. Amazon Q アイコンを選択してチャットパネルを開きます。
3. チャットパネル**/transform**に と入力します。
4. Java アプリケーションが SQL 変換の対象である場合、実行する変換のタイプを選択するように Amazon Q から求められます。**SQL conversion** と入力してください。
5. Amazon Q は、Amazon S3 から取得したスキーマメタデータファイルをアップロードするように求めます。チャットで、Amazon Q はファイルを取得する手順を提供します。
6. Amazon Q では、埋め込み SQL とデータベーススキーマファイルを含むプロジェクトを指定するように求められます。チャットパネルのドロップダウンメニューから適切なファイルを選択します。
7. データベーススキーマから取得した Amazon Q の詳細が正確であることを確認します。
8. Amazon Q は SQL コードの変換を開始します。これには数分かかる場合があります。
9. Amazon Q は SQL コードを変換した後、ファイルに加えた更新を差分として提供します。差分の変更を確認し、変更を受け入れてコードを更新します。

   Amazon Q は、変更の詳細を含む変換の概要も提供します。
10. コードを更新したら、DMS AWS コンソールに戻り、新しい SQL が移行されたデータベースと互換性があることを確認します。

Amazon Q Developer によるコマンドラインでのコードの変換

Amazon Q Developer コマンドライン変換ツールを使用して、コマンドラインからアプリケーションを変換できます。コードを変換するには、ソースコードと必要な設定ファイルへのパスを指定します。Amazon Q は一連のステップで新しいコードを生成します。変換全体を通して、Amazon Q はローカル環境にコードを構築し、変更を検証します。詳細については、「ローカル環境でのコードの構築」を参照してください。Amazon Q は、コード変更をコミットする新しいブランチをリポジトリに作成します。変換が完了したら、ブランチを元のブランチにマージして、変更をコードベースに組み込むことができます。

開始するには、コマンドラインツールをインストールして認証し、変換を設定して開始するコマンドを確認します。

トピック

- ローカル環境でのコードの構築
- コマンド
- Amazon Q Developer を使用してコマンドラインで変換を実行する
- コマンドラインでの変換のトラブルシューティング
- Amazon Q Developer コマンドライン変換ツールのバージョン履歴

ローカル環境でのコードの構築

変換中、Amazon Q はローカル環境で検証ビルドを実行します。Amazon Q は、サーバー側のコードを複数のステップで変換します。各ステップの後、Amazon Q はローカル環境にコードを送信し、変更をビルドしてテストします。その後、コードはサーバー側に送信され、変換が続行されます。

ローカル環境のビルドは、Amazon Q がプライベートリソースへのアクセスを必要とするテストを実行できるようにすることで、変換されたコードを検証するのに役立ちます。ローカル環境で AI が生成するコードの構築に関連するセキュリティリスクを最小限に抑えるために、Amazon Q はセキュリティ上の懸念に対応するために生成するコードを確認して更新します。

> **Note**
>
> Amazon Q は、プロジェクトのリクエスト、説明、コンテンツに基づいて変換を実行します。セキュリティを維持するには、外部で未検証のアーティファクトをプロジェクトリポジトリに含めないようにし、機能とセキュリティの両方について変換されたコードを常に検証します。

コマンド

これらのコマンドを実行する手順についてはstep-by-step「」を参照してくださいAmazon Q Developer を使用してコマンドラインで変換を実行する。

変換を設定し、Amazon Q Developer Pro に認証するには、以下を実行します。

```
qct configure
```

Java アップグレードの変換を開始するには、次のコマンドを実行します。*<your-source-java-version>* には、JAVA_1.8、、JAVA_8JAVA_11、JAVA_17、または を入力できますJAVA_21。*<your-target-java-version>* には、 JAVA_17または を入力できますJAVA_21。--source_version および --target_version はオプションです。--trust フラグを使用すると、セキュリティを維持するためにコードを検査しながら変換を実行できます。

```
qct transform --source_folder <path-to-folder>
    --source_version <your-source-java-version>
    --target_version <your-target-java-version>
    --trust
```

SQL 変換の変換を開始するには、以下を実行します。

```
qct transform --source_folder <path-to-folder>
    --sql_conversion_config_file <path-to-sql-config-file>
```

使用している変換用のコマンドラインツールのバージョンを確認するには、以下を実行します。

```
qct -v
```

変換に関するヘルプを取得するには、以下を実行します。

```
qct -h
```

変換ジョブ履歴を表示するには、以下を実行します。

```
qct history
```

変換ジョブ履歴の表示と管理の詳細については、「」を参照してくださいコマンドラインでのジョブ履歴の表示。

Amazon Q Developer を使用してコマンドラインで変換を実行する

Amazon Q Developer コマンドラインツールを使用してコマンドライン上のコードを変換するには、以下の手順を実行します。

前提条件

コマンドラインで変換を開始する前に、次の前提条件を満たす必要があります。

- Java コードバージョンをアップグレードする場合、プロジェクトは Amazon Q で Java バージョンをアップグレードするための前提条件を満たしています。
- Java アプリケーションで埋め込み SQL を変換する場合、アプリケーションは Amazon Q で埋め込み SQL を変換するための前提条件を満たしています。
- コマンドライン環境に Python がインストールされている。このようにコマンドラインツールをインストールします。サポートされている最小 Python バージョンは 3.12 です。
- macOS または Linux で変換を実行しています。
- アプリケーションのサイズは 2 GB 以下です。
- Amazon Q でアップグレードする特定の依存関係がある場合は、依存関係のアップグレードファイルを設定済みです。

ステップ 1: 認証方法を選択し、アクセス許可を追加する

IAM Identity Center を認証して、コマンドラインで変換を実行できます。適切なアクセス許可があることを確認します。

> **Note**
>
> カスタマーマネージドキーは、コマンドラインで実行される変換ではサポートされていません。

アクセス許可を追加する

認証に使用する Amazon Q Developer サブスクリプションに関連付けられている IAM ID には、コマンドラインで変換を実行するアクセス許可が必要です。続行する前に、IAM ID に で定義されているアクセス許可があることを確認してくださいユーザーにコマンドラインで変換の実行を許可する。

Amazon Q Developer サブスクリプションを使用して IAM Identity Center で認証する

IAM Identity Center で認証するには、管理者がワークフォースユーザーとして Amazon Q Developer Pro にサブスクライブし、サブスクリプションを通じて認証するための開始 URL を指定する必要があります。ユーザーまたは管理者は、Amazon Q Developer コンソールで開始 URL を見つけることができます。詳細については、Amazon Q Developer で使用する開始 URL の検索 を参照してください。

必要なアクセス許可を追加するには、「」を参照してくださいアクセス許可を追加する。

で開始 URL を指定しますステップ 4: を設定して認証する。

ステップ 2: ツールをインストールする

1. 変換用の Amazon Q コマンドラインツールをダウンロードして解凍します。

   コマンドラインツールの以前のバージョンをダウンロードするには、「」を参照してくださいバージョン履歴。
2. ツールをインストールするには、Python で仮想環境を設定することをお勧めします。仮想環境を作成するには、ツールをインストールして実行するディレクトリでターミナルウィンドウを開きます。

   ```
   python -m venv qct-cli
   ```
3. 仮想環境をアクティブ化するには、以下を実行します。

   ```
   source qct-cli/bin/activate
   ```
4. コマンドラインにツールをインストールするには、マシンアーキテクチャに基づいて、ツールを解凍した場所へのパスを指定して次のコマンドを実行します。

   Linux_aarch64

   ```
   pip install <path/to/unzipped-tool>/Linux_aarch64/amzn_qct_cli-1.2.0-py3-none-any.whl
   ```

   Linux_x86_64

   ```
   pip install <path/to/unzipped-tool>/Linux_x86_64/amzn_qct_cli-1.2.0-py3-none-any.whl
   ```

> **Note**
>
> 変換に古いバージョンのコマンドラインツールを使用している場合は、 をダウンロードしたバージョン`1.2.0`に置き換えます。

5. ツールがインストールされていることを確認するには、以下を実行します。

```
which qct
```

ステップ 3: 依存関係のアップグレードファイルを作成する (オプション)

依存関係のアップグレードファイル、プロジェクトの依存関係を一覧表示する YAML ファイル、および変換中にアップグレードするバージョンを Amazon Q に提供できます。依存関係のアップグレードファイルを提供することで、Amazon Q がアップグレードについて認識していないサードパーティーとサードパーティーの依存関係を指定できます。

ファーストパーティの依存関係とは、組織が維持し、ローカルまたは組織のプライベートネットワークでのみ利用できるライブラリ、プラグイン、フレームワークを指します。Amazon Q は、ローカル環境でビルドを実行するときに、ファーストパーティの依存関係にアクセスできます。詳細については、「ローカル環境でのコードの構築」を参照してください。サードパーティーの依存関係は、公開されているか、組織に固有ではないオープンソースの依存関係です。

YAML ファイルでアップグレードするファーストパーティの依存関係を指定できます。Amazon Q は JDK アップグレード中にそれらをアップグレードします (Java 8 から 17 など) 。最初の JDK アップグレード後に別の変換 (17 から 17 または 21 から 21) を開始して、サードパーティーの依存関係をアップグレードできます。

Amazon Q が最小限の JDK アップグレードを実行したら、別の変換を開始して、すべてのサードパーティーの依存関係をアップグレードできます。または、YAML ファイルでサードパーティーの依存関係とそのバージョンを指定して、ライブラリのアップグレード変換中にのみそれらの依存関係をアップグレードすることもできます。

Amazon Q では、変換中に依存関係のアップグレードファイルを指定するように求められます。指定する場合は、まず ファイルが正しく設定されていることを確認してください。YAML ファイルには次のフィールドが必要です。

- name - 依存関係のアップグレードファイルの名前。

- description (オプション) - 依存関係のアップグレードファイルと変換の説明。
- dependencyManagement - アップグレードする依存関係とプラグインのリストが含まれます。
- dependencies - アップグレードするライブラリの名前とバージョンが含まれます。
- plugins - アップグレードするプラグインの名前とバージョンが含まれます。
- identifier - ライブラリ、プラグイン、またはその他の依存関係の名前。
- targetVersion - アップグレード先の依存関係のバージョン。
- versionProperty (オプション) - アプリケーションの pom.xml ファイルの properties タグで設定されている、定義する依存関係のバージョン。
- originType - FIRST_PARTY または Third_PARTY のいずれかで指定された、依存関係がファーストパーティかサードパーティーか。

以下は、依存関係のアップグレード YAML ファイルの例と、Amazon Q が解析するために必要な設定です。

```
name: dependency-upgrade

description: "Custom dependency version management for Java migration from JDK 8/11/17
 to JDK 17/21"


dependencyManagement:

  dependencies:

    - identifier: "com.example:library1"

      targetVersion: "2.1.0"

      versionProperty: "library1.version"  # Optional

      originType: "FIRST_PARTY"

    - identifier: "com.example:library2"

      targetVersion: "3.0.0"

      originType: "THIRD_PARTY"

  plugins:
```

```
  - identifier: "com.example.plugin"
    targetVersion: "1.2.0"
    versionProperty: "plugin.version"  # Optional
    originType: "THIRD_PARTY"
```

ステップ 4: を設定して認証する

変換を開始する前に、IAM Identity Center で認証し、変換の設定の詳細を指定する必要があります。

1. 変換設定プロセスを開始するには、次のコマンドを実行します。

   ```
   qct configure
   ```

2. サポートされている Java バージョンごとに JDK パスを入力するように求められます。Java アプリケーションのソースバージョンの JDK へのパスのみを指定する必要があり、ターゲットバージョンを指定する必要はありません。
3. 次に、IAM Identity Center で認証するには、Amazon Q Developer Pro サブスクリプションプロファイルの開始 URL を入力するように求められます。

   次に、サブスクライブした AWS リージョン を の形式で入力しますus-east-1。サポートされているリージョンのリストについては「サポート対象のリージョン」を参照してください。リージョンコードのリストについては、 AWS 全般のリファレンス ガイドの「リージョンエンドポイント」を参照してください。
4. 設定設定は configuration.ini ファイルに保存されます。

ステップ 5: 変換を実行する

実行する変換のタイプを選択して、必要な設定とコマンドを確認します。

> **Note**
>
> クライアント側のビルドでは安定したネットワーク接続が必要であるため、コード変換中にローカルマシンをオフまたは閉じないでください。

Java upgrade

変換プランの変更

Java バージョンのアップグレード中、Amazon Q は変換プランを生成し、変換を開始する前に確認できます。プランには、次の変更をリクエストするオプションがあります。

- プランに含まれるリストから Amazon Q がアップグレードするライブラリ
  - プロンプトの例：
    - <dependency1>、<dependency2>、<dependency5> のみをアップグレードする
    - <dependency1> または <dependency2> をアップグレードしない
- ライブラリをアップグレードするターゲットバージョン
  - プロンプトの例：
    - <version> の代わりに <dependency> をこのバージョンにアップグレードする
- Amazon Q が実行するステップ
  - プロンプトの例：
    - ステップ 1～7 のみを完了する
    - ステップ 5～9 を実行しない
- アップグレードする依存関係を追加する (新しい JDK バージョンにアップグレードする場合はオプションのみ)
  - プロンプトの例：
    - また、<dependency1> を <version2> にアップグレードします。

Java コードのアップグレード

1. 次のコマンドを実行して、Java アップグレードの変換を開始します。<path-to-folder> を変換するコードでフォルダへのパスに、 を JAVA_17または <your-target-java-version>に置き換えますJAVA_21。

   ```
   qct transform --source_folder <path-to-folder>
       --target_version <your-target-java-version>
   ```

   追加のコマンドオプション：

   - アップグレードする依存関係を指定する場合は、依存関係のアップグレードファイルへのパスを指定して --dependency_upgrade_fileオプションを追加します。

- 変換プランを確認または更新しない場合は、 コマンドに `--no-interactive`フラグを追加します。Amazon Q はプランに関するフィードバックを求めず、変更をリクエストする機会もありません。

2. Maven のバージョンは、変換を開始する前に検証されます。少なくともサポートされている最小バージョンがある場合は、次の出力が表示されます。

```
Running command: mvn --version at: path/to/current/directory
Your Maven version is supported for transformations.
```

サポートされているバージョンの Maven がない場合は、更新して続行する必要があります。詳細については、「前提条件」を参照してください。

3. `--no-interactive` フラグを追加しなかった場合、Amazon Q は変換計画に関するフィードバックを提供するように求めます。変更内容を英語の自然言語で説明できます。Amazon Q は、リクエストした変更をサポートできる場合、プランを更新します。

4. Amazon Q が変換を開始します。変換中にステータスの更新が出力されます。完了すると、Amazon Q は変換結果、ログ、および設定ファイルが出力されるパスを提供します。

アップグレードされたコードは、Amazon Q が作成した新しいブランチにコミットされます。Amazon Q は、 の実行時に選択した内容に応じて、1 つ以上のコミットでコードをコミットしますqct configure。

5. Java バージョンのアップグレード後に別の変換を実行している場合は、最初の変換からの変更をコミットしたのと同じブランチで 2 番目の変換を開始します。

SQL conversion

開始する前に、このタイプの変換の前提条件を理解Amazon Q Developer を使用した Java アプリケーションの埋め込み SQL の変換したことを確認してください。

1. 埋め込み SQL を変換するには、まず AWS DMS Schema Conversion からスキーマメタデータファイルへのパスを含む YAML ファイルを作成する必要があります。

ファイルの必須形式は次のとおりです。

```
schema_conv_metadata_path: <path-to-metadata-zip-file>
```

2. 次のコマンドを実行して、SQL 変換の変換を開始します。を変換するコードでフォルダへのパス<path-to-folder>に、 をステップ 1 で作成した YAML ファイルへのパス<path-to-sql-config-file>に置き換えます。

```
qct transform --source_folder <path-to-folder>
    --sql_conversion_config_file <path-to-sql-config-file>
```

3. Amazon Q がスキーマメタデータファイルで複数のスキーマを検出した場合、変換を停止し、検出されたスキーマのリストを提供します。SQL 変換に使用するスキーマを選択し、schema: <schema-name>YAML ファイルに新しいフィールドを追加します。
4. Amazon Q が変換を開始します。変換中にステータスの更新が出力されます。完了すると、Amazon Q は変換結果、ログ、および設定ファイルが出力されるパスを提供します。

   アップグレードされたコードは、Amazon Q が作成した新しいブランチにコミットされます。

変換を一時停止またはキャンセルする

現在の変換ジョブを一時停止またはキャンセルできます。変換ジョブは、再開する前に最大 12 時間一時停止できます。

コード変換ジョブを一時停止またはキャンセルするには

1. CLI ターミナルで、キーボードの Ctrl+C を押します。
2. 変換を一時停止するかキャンセルするかを選択します。

   - コード変換ジョブを消去1する場合は、 を入力します。12 時間以内にジョブを再開して、次の QCT コマンド を使用してコード変換を続行できます`qct transform --source_folder=≤/Path/Given/Originally/To/QCT>`。
   - コード変換ジョブをキャンセル2する場合は、 を入力します。

コマンドラインでの変換のトラブルシューティング

以下の情報は、Amazon Q Developer でコマンドラインでアプリケーションを変換する際の一般的な問題のトラブルシューティングに役立ちます。

ベアラートークンが更新されないのはなぜですか？

次のエラーが表示された場合は、認証に使用されるベアラートークンを更新する必要があることを意味します。

```
Refreshing bearer token
('Error refreshing bearer token due to: ', InvalidGrantException('An error occurred
 (InvalidGrantException) when calling the CreateToken operation: '))
('Error getting bearer token due to: ', RuntimeError(('Error refreshing bearer token
 due to: ', InvalidGrantException('An error occurred (InvalidGrantException) when
 calling the CreateToken operation: '))))
```

このエラーに対処するには、次のコマンドを実行します。

```
rm ~/.aws/qcodetransform/credentials.json
```

古い認証情報ファイルを削除したら、 `qct transform` を再度実行して変換を再開します。

コマンドラインツールの最新バージョンが使用されていないのはなぜですか？

変換用にコマンドラインツールの新しいバージョンをダウンロードすると、以前のバージョンのツールが引き続き使用されることがあります。

ツールの最新バージョンを使用するには、最新バージョンをダウンロードします。次に、マシンアーキテクチャに基づいて、ツールを解凍した場所へのパスで次のコマンドを実行します。

Linux_aarch64

```
pip install <path/to/unzipped-tool>/Linux_aarch64/amzn_qct_cli-1.2.0-py3-none-
any.whl --force-reinstall
```

Linux_x86_64

```
pip install <path/to/unzipped-tool>/Linux_x86_64/amzn_qct_cli-1.2.0-py3-none-any.whl
 --force-reinstall
```

> **Note**
>
> 変換に古いバージョンのコマンドラインツールを使用している場合は、 をダウンロードしたバージョン`1.2.0`に置き換えます。

Amazon Q Developer コマンドライン変換ツールのバージョン履歴

Amazon Q Developer コマンドライン変換ツールの現在および過去のリリースの詳細については、次の情報を参照してください。この表には、各バージョンのダウンロードリンク、リリース日、リリースノートが含まれています。

バージョン	リリース日	リリースノート
1.2.0 (最新)	2025 年 8 月 7 日	Maven Java プロジェクトのジョブ履歴の表示とモジュール構造の視覚化のサポートが追加されました。
1.1.0	2025 年 7 月 21 日	変換に関するテレメトリの収集のサポートが含まれています。
1.0.0	2025 年 6 月 27 日	コマンドライン変換ツールは一般提供されており、Amazon Q Developer Pro サブスクリプションでのみ AWS IAM Identity Center による認証をサポートしています。欧州 (フランクフルト) リージョンでのサブスクリプションのサポートが追加されました。
0.6.0	2025 年 6 月 6 日	依存関係のアップグレードファイルの提供と変換プランの反復処理のサポートが含まれています。
0.5.2	2025 年 4 月 16 日	ファーストパーティ依存関係を持つアプリケーションのジョブと障害を再開する際の問題を解決するためのバグ修正。

バージョン	リリース日	リリースノート
0.5.1	2025 年 3 月 13 日	IAM で認証する場合、 を指定する必要はありません AWS リージョン。出力ログにジョブステータスを含めるバグ修正も含まれています。
0.5.0	2025 年 2 月 28 日	を介した IAM による認証のサポートが含まれています AWS CLI。
0.4.1	2025 年 2 月 17 日	Amazon Q Developer サブスクリプションが設定され AWS リージョン ている の入力のサポートを含むバグ修正。
0.4.0	2025 年 2 月 14 日	Java アプリケーションを Java 21 にアップグレードするためのサポートが含まれています。
0.3.0	2025 年 2 月 12 日	Java アプリケーションでの埋め込み SQL の変換のサポートが含まれています。
0.2.0	2025 年 2 月 3 日	複数のコミットでアップグレードされた Java コードを受信するためのサポートが含まれています。
0.1.0	2024 年 11 月 27 日	初回リリース。コマンドラインから Java コードバージョンをアップグレードするためのサポートが含まれています。

変換ジョブ履歴の表示

Amazon Q は、Java 変換ジョブ履歴の包括的な概要を提供します。これにより、IDEs とコマンドラインの両方で変換ジョブを追跡および確認できます。

変換ジョブ履歴には、ジョブに関する次の情報が含まれます。

- 日付 – 変換ジョブが実行された日時
- プロジェクト名 – 変換されたプロジェクトの名前
- ステータス – 変換ジョブの現在のステータス
- 所要時間 — 変換の完了にかかった時間
- ジョブ ID – 変換ジョブの一意の識別子
- 差分パッチ – すべてのコード変更を示す最終的な差分パッチファイルへのリンクまたはパス
- 概要 – 変更の詳細を含む変換概要ファイルへのリンクまたはパス

> **Note**
>
> この機能がリリースされてから実行される変換のみがジョブ履歴で使用できます。機能のリリース日については、「」を参照してくださいAmazon Q Developer ユーザーガイドのドキュメント履歴。

IDEs でのジョブ履歴の表示

> **Note**
>
> この機能は現在、 Visual Studio Codeでのみ使用できます。

の Transformation Hub Visual Studio Codeは、Java 変換ジョブ履歴の包括的なビューを提供します。

Transformation Hub のテーブルには、過去 30 日間の最新の 10 個の変換ジョブが一覧表示されます。テーブルから変換アーティファクトと更新ジョブにアクセスして、進行状況を追跡し、欠落しているアーティファクトを取得できます。

変換アーティファクトを取得する

差分パッチや概要ファイルなどの変換アーティファクトには、ジョブ履歴テーブルからアクセスできます。適切なリンクを選択して、IDE で差分または概要を開きます。

アーティファクトは .aws/transform ディレクトリにローカルに保存されるため、過去のジョブから以前にダウンロードした変換アーティファクトにアクセスすることもできます。

ジョブのステータスを更新する

ジョブ履歴テーブルからジョブステータスを更新できます。失敗したジョブを更新して、Amazon Q が失敗したジョブを再開できる場合など、サーバー側からまだサーバーに達していない可能性のある更新されたステータスを取得します。完了したジョブを更新して、まだ表示されていないアーティファクトをダウンロードすることもできます。

IDE で実行されるジョブのジョブ履歴の保存方法

の場合Visual Studio Code、すべての変換ジョブ情報とアーティファクトは .aws/transform ディレクトリにローカルに保存されます。ストレージは次のように整理されています。

```
.aws/transform/
### [project-name-1]/
#   ### [job-id-1]/
#   #   ### diff.patch
#   #   ### [summary-1]/
#   #   #   ### summary.md
#   #   #   ### buildCommandOutput.log
#   ### [job-id-2]/
#       ### diff.patch
#       ### [summary-2]/
#       #   ### summary.md
#       #   ### buildCommandOutput.log
### [project-name-2]/
    ### [job-id-3]/
        ### diff.patch
        ### [summary-3]/
        #   ### summary.md
        #   ### buildCommandOutput.log
```

コマンドラインでのジョブ履歴の表示

コマンドラインでの変換の場合、 qct history コマンドはカスタマイズオプションを使用して変換ジョブ履歴へのアクセスを提供します。

CLI の場合、変換ジョブ履歴情報は .aws/qcodetransform/history/ ディレクトリにローカルに保存されます。

QCT 履歴コマンドの使用

変換ジョブ履歴を表示するための基本的なコマンドは次のとおりです。

```
qct history
```

デフォルトでは、このコマンドは、一時停止または進行中のジョブに加えて、最新の 10 個の変換ジョブを表示します。

--limit フラグで表示するジョブ履歴エントリの数を指定することもできます。たとえば、20 個のジョブを表示するには、以下を実行します。

```
qct history --limit 20
```

Java 変換に関する問題のトラブルシューティング

以下の情報は、Amazon Q Developer で Java アプリケーションを変換する際の一般的な問題のトラブルシューティングに役立ちます。

トピック

- Amazon Q がプロジェクトをアップロードできないのはなぜですか?
- Maven コマンドが失敗するのはなぜですか?
- Maven を PATH に追加するにはどうすればよいですか?
- Amazon Q がコードをビルドできないのはなぜですか?
- 55 分経過後に変換が失敗したのはなぜですか?
- 変換されたコードをダウンロードできないのはなぜですか?
- コード変換ログにアクセスするにはどうすればよいですか?
- 変換ジョブ ID を見つけるにはどうすればよいですか?

Amazon Q がプロジェクトをアップロードできないのはなぜですか?

プロジェクトのアップロードに失敗した場合、次の問題のいずれかが原因である可能性があります。Amazon Q で表示されるエラーに対応するトピックを参照してください。

トピック

- プロジェクトサイズを縮小する
- IDE でプロキシを設定する
- Amazon S3 へのアクセスを許可する

プロジェクトサイズを縮小する

コードを変換するために、Amazon Q はソースコード、プロジェクトの依存関係、ビルドログを含むプロジェクトアーティファクトを生成します。変換ジョブのプロジェクトアーティファクトサイズの上限は 2 GB です。プロジェクトアーティファクトのサイズに関連するエラーが発生した場合は、プロジェクトのサイズを小さくするか、より小さなプロジェクトを変換する必要があります。プロジェクトアーティファクトファイルのサイズは、コード変換ログで確認できます。詳細については、コード変換ログにアクセスするにはどうすればよいですか?を参照してください。

IDE でプロキシを設定する

コードを変換するために、Amazon Q はプロジェクトアーティファクトをサービス所有の Amazon S3 バケットにアップロードします。アップロードプロセスの一部では、SSL 証明書または TLS 証明書を使用して Amazon S3 と IDE 間の通信を確立します。プロキシサーバーを使用している場合、プロキシサーバーで使用される SSL 証明書または TLS 証明書は信頼されている必要があります。信頼されていない場合、Amazon Q はプロジェクトをアップロードできません。

プロキシまたは証明書に関連するエラーが発生した場合は、証明書を信頼するか、他のプロキシ設定を更新するように IDE またはオペレーティングシステムを設定する必要があります。

> **Note**
>
> また、組織のプロキシサーバーまたはファイアウォールの後ろにいる場合、証明書とは無関係の問題が発生する可能性があります。次の手順で証明書を設定しても問題が解決しない場合は、ネットワーク管理者に連絡して、IDE から Amazon S3 との通信が許可されていることを確認してください。詳細については、「Amazon S3 へのアクセスを許可する」を参照してください。

JetBrains の証明書を設定する

プロキシサーバーで使用される SSL 証明書または TLS 証明書を信頼するように JetBrains IDE の Java ランタイム環境 (JRE) を設定するには、SSL 証明書または TLS 証明書を JRE 内の `cacerts` ファイルにインポートする必要があります。`cacerts` ファイルは、HTTPS や SSL などのセキュアな接続に称される信頼されたルート証明書を含むファイルであり、JRE のセキュリティ設定の一部です。証明書をインポートする手順は次のとおりです。

> **Note**
>
> `cacerts` ファイルを変更する前にバックアップを作成することをお勧めします。ミスがあると、セキュアな接続に問題が発生する可能性があるためです。

1. JRE 内の `cacerts` ファイルへのパスを特定します。JetBrains IDE に同梱されている内部 JRE 内の `cacerts` ファイルのパスは、オペレーティングシステムと使用している JetBrains IDE のバージョンによって異なります。

   以下は、一般的なオペレーティングシステムの `cacerts` ファイルへのパスの例です。オペレーティングシステムを選択すると、例が表示されます。

   > **Note**
   >
   > `<JetBrains Installation Folder>` は、JetBrains 製品がインストールされているディレクトリを指します。このディレクトリは通常、インストールプロセス中に選択されます。
   > `jbr` フォルダは、JetBrains IDE にバンドルされた JRE を表します。これは、JetBrains IDE での使用に合わせてカスタマイズされた JRE の特定のバージョンです。

   Windows

   Windows にインストールされている JetBrains IDE の `cacerts` ファイルパスは次のとおりです。

   ```
   <JetBrains Installation Folder>\jbr\bin\cacerts
   ```

例えば、Windows に JetBrains IDE をデフォルトの場所にインストールした場合、パスは次のようになります。

```
C:\Program Files\JetBrains\jbr\bin\cacerts
```

macOS

macOS にインストールされている JetBrains IDE の `cacerts` ファイルパスは次のとおりです。

```
/Applications/JetBrains Toolbox/<version>/JetBrains Toolbox.app/Contents/jbr/
Contents/Home/lib/security/cacerts
```

例えば、macOS に JetBrains IDE をデフォルトの場所にインストールした場合、パスは次のようになります。

```
/Applications/JetBrains Toolbox/2022.3.4/JetBrains Toolbox.app/Contents/jbr/
Contents/Home/lib/security/cacerts
```

Linux

Linux にインストールされている JetBrains IDE の `cacerts` ファイルパスは次のとおりです。

```
/opt/jetbrains/jbr/lib/security/cacerts
```

2. `cacerts` ファイルにインポートする必要がある証明書を特定します。証明書ファイルの拡張子は通常 `.cer`、`.crt`、または `.der` です。追加すべき証明書がわからない場合は、ネットワーク管理者にお問い合わせください。
3. 証明書を `cacerts` キーストアにインポートします。これは Java の `keytool` コマンドで行うことができます。

   a. コマンドプロンプトを開き、次のコマンドを使用します。

   ```
   keytool -import -alias <alias> -file <certificate_file> -keystore
    <path_to_cacerts>
   ```

b. <alias> については、インポートする証明書の名前を追加しておけば、後で参照できます。このオプションはオプションです。

c. <certificate_file> には、インポートする証明書へのパスを指定します。これは、証明書を含む .cer、.crt、または .der ファイルへのパスを指定してください。

d. <path_to_cacerts> には、ステップ 1 で保存した cacerts キーストアファイルへのパスを指定します。これは、証明書をインポートするファイルです。

例えば、Windows 上にある IntelliJ IDEA のバンドル JRE の cacerts キーストアに my_certificate.cer という名前の証明書をインポートし、証明書に myalias というエイリアスを付与する場合、 コマンドは次のようになります。

```
keytool -import -alias myalias -file my_certificate.cer -keystore "C:\Program Files
\JetBrains\IntelliJ IDEA 2022.3.2\jbr\bin\cacerts"
```

4. インポートプロセス中に、キーストアパスワードの入力を求められます。cacerts キーストアのデフォルトのパスワードは changeit です。
5. コマンドを実行すると、証明書を信頼するように求められます。証明書が信頼できることを確認してインポートを完了するには、yes と入力します。
6. JRE に加えて、IDE 自体に証明書を追加する必要がある場合もあります。詳細については、JetBrains ドキュメントの「Server Certificates」を参照してください。

Visual Studio Code の証明書を設定する

プロキシサーバーで使用される SSL 証明書または TLS 証明書を信頼するように Visual Studio Code を設定するには、オペレーティングシステムに次のプロキシ設定が設定されていることを確認してください。

macOS で Visual Studio Code の証明書を設定する

macOS 上の Visual Studio Code で次のプロキシ設定を行います。

macOS キーチェーンに証明書を追加する

まだ追加していない場合は、プロキシサーバーが使用する証明書を macOS キーチェーンに追加する必要があります。キーチェーンに証明書を追加する方法については、「Keychain Access User Guide」の「Add certificates to a keychain using Keychain Access on Mac」を参照してください。

Mac CA VSCode 拡張機能をインストールする

Mac CA VSCode 拡張機能を使用すると、Amazon Q は Mac の Keychain Access に追加した証明書にアクセスできます。

拡張機能をインストールするには:

1. VS Code 拡張機能ペインで `mac-ca-vscode` を検索し、[インストール] を選択します。
2. VS Code を再起動します。

macOS で VS Code のプロキシ設定を更新する

次の設定を更新して、VS Code がプロキシに正しく設定されていることを確認します。

1. VS Code で設定を開きます。
2. 検索バーに「`proxy`」と入力します。
3. [Http: Proxy] フィールドに、プロキシ URL を追加します。
4. [Http: Proxy Strict SSL] のチェックボックスをオフにします。
5. [Http: Proxy Support] ドロップダウンリストで、 [on] を選択します。
6. 設定検索バーに「`http.experimental.systemCertificatesV2`」と入力します。[Http › Experimental: System Certificates V2] を選択します。

Windows で Visual Studio Code の証明書を設定する

Windows 上の Visual Studio Code で次のプロキシ設定を行います。

Windows で信頼されたルート証明書として証明書を追加する

まだ追加していない使用していない場合は、プロキシサーバーが使用する証明書を Windows の信頼されたルート証明機関のストアに追加する必要があります。証明書を追加するには、次の手順を実行します。

1. 検索ツールまたは Run コマンドウィンドウを開きます。
2. 次のコマンドを入力して、Certificate Manager ツールを開きます。

```
certmgr.msc
```

3. [信頼されたルート証明機関] ストアを選択します。

4. [証明書] を右クリックし、[すべてのタスク]、[インポート] の順に選択します。
5. 指定された手順に従って、プロキシ証明書をインポートします。
6. 証明書をインポートしたら、証明書が追加されていることを確認します。

   [信頼されたルート証明機関] ストアで、[証明書] をダブルクリックします。追加した証明書を右クリックし、[プロパティ] を選択します。[証明書の目的] で、[この証明書の目的をすべて有効にする] オプションが選択されていることを確認します。

Win-CA VSCode 拡張機能をインストールする

Win-CA VSCode 拡張機能により、Amazon Q は Windows で信頼されたルート証明書に追加した証明書にアクセスできます。

拡張機能をインストールするには:

1. VS Code 設定ペインで「`win-ca `」を検索します。
2. [Inject] ドロップダウンリストで、[append] を選択します。

Windows の VS Code でプロキシ設定を更新する

次の設定を更新して、VS Code がプロキシに正しく設定されていることを確認します。

1. VS Code で設定を開きます。
2. 検索バーに「`proxy`」と入力します。
3. [Http: Proxy] フィールドに、プロキシ URL を追加します。
4. [Http: Proxy Strict SSL] のチェックボックスをオフにします。
5. [Http: Proxy Support] ドロップダウンリストで、 [on] を選択します。
6. 設定検索バーに「`http.experimental.systemCertificatesV2`」と入力します。[Http › Experimental: System Certificates V2] を選択します。
7. VS Code を再起動します。

Amazon S3 へのアクセスを許可する

変換中、Amazon Q はサービス所有の Amazon S3 バケットにコードをアップロードします。ネットワークまたは組織が Amazon S3 へのアクセスを設定していない場合、Amazon Q はプロジェクトをアップロードできません。

Amazon Q がプロジェクトをアップロードできるようにするには、プロキシ設定や Data Lost Prevention (DLP) ポリシーなどのネットワークコンポーネントが Amazon S3 へのアクセスを許可するように設定されていることを確認します。また、Amazon Q がプロジェクトをアップロードする Amazon S3 バケットを許可リストに登録する必要が生じる場合もあります。詳細については、「許可リストの Amazon S3 バケット URLs と ARNs」を参照してください。

大規模なプロジェクトを変換する場合、DLP ポリシーなどのネットワークコンポーネントが Amazon S3 バケットを許可リストに登録するように設定されていないと、遅延が発生し、アップロードが成功しない可能性があります。バケットを許可リストに登録しない場合、Amazon Q がバケットをアップロードできるように、より小さなプロジェクトを変換する必要がある場合があります。

Maven コマンドが失敗するのはなぜですか?

以下は、JetBrains IDE および Visual Studio Code IDE で発生する可能性のある Maven の設定の問題です。問題に対処しても Maven エラーが表示される場合は、プロジェクトに問題がある可能性があります。エラーログの情報を使用して、プロジェクトに関する問題に対処し、プロジェクトをもう一度変換してみてください。

JetBrains で Maven の設定を更新する

JetBrains での変換が Maven コマンドの問題が原因で失敗すると、エラーログが [Run] タブに表示されます。ログの情報を使用して問題に対処します。以下は、対処する必要がある問題の例です。

- Maven のホームパスが [Bundled] に設定されていることを確認します。[Settings] ダイアログボックスで、[Build, Execution, Deployment] セクションを展開します。[Build Tools] セクションを展開し、[Maven] を展開します。[Maven home path] ドロップダウンリストで、[Bundled] を選択します。
- Java ランタイム環境 (JRE) でプロジェクト JDK が使用されていることを確認します。[Settings] ダイアログボックスで、[Build, Execution, Deployment] セクションを展開します。[Maven] を展開して [Runner] を選択します。[JRE] ドロップダウンリストで、[Use Project JDK] を選択します。
- 「Maven」が有効になっていることを確認します。[Settings] に移動し、[Plugins] を選択します。「Maven」を検索して Maven プラグインを選択します。[Enable] ボタンが表示された場合は、選択して Maven を有効にします。

Visual Studio Code で Maven の設定を更新する

VS Code での変換が Maven コマンドの問題が原因で失敗すると、エラーログを含むテキストファイルが新しいタブで開きます。ログの情報を使用して問題に対処します。

次のオプションのいずれかが設定されていることを確認します。

- プロジェクトのプロジェクトルートフォルダに Maven ラッパーが含まれている
- Amazon Q でサポートされている Maven のバージョンが PATH で利用可能になっている

詳細については、Maven を PATH に追加するにはどうすればよいですか?を参照してください。

Maven を **PATH** に追加するにはどうすればよいですか?

Maven ラッパーを使用せずに VS Code でコードを変換するには、Maven をインストールして PATH 変数に追加する必要があります。

Maven が既に正しくインストールされているかどうかを確認するには、Visual Studio Code の外部にある新しい OS ターミナルで `mvn -v` を実行します。Maven のバージョンが出力に表示されるはずです。

Visual Studio Code ターミナルで出力が得られるのに OS ターミナルで出力が得られない場合、またはコマンドが見つからない場合は、Maven を PATH に追加する必要があります。

Maven を PATH に追加するには、ご利用のマシン向けの手順を実行します。

macOS

Maven を macOS PATH に追加するには、次の手順を実行します。

1. Maven インストールディレクトリ、または Maven をインストールしたフォルダを見つけ、そのフォルダへのパスを保存します。
2. 任意の エディタでシェルの設定ファイルを開きます。最近の macOS バージョンでは、デフォルトのシェルは `zsh` で、デフォルトの設定ファイルは `~/.zshrc` にあります。

   設定ファイルに次の行を追加します。`M2_HOME` の値は、ステップ 1 で保存したパスに設定します。

   ```
   export M2_HOME="your Maven installation directory"
   export PATH="${M2_HOME}/bin:${PATH}"
   ```

   これらのコマンドにより、`mvn` コマンドがすべてのターミナルで使用できるようになります。
3. すべての OS ターミナルウィンドウを閉じて、すべての Visual Studio Code インスタンスを終了します。

4. Maven が PATH に追加されたことを確認するには、新しい OS ターミナルを開き、次のコマンドを実行します。

   ```
   mvn -v
   ```

   Maven のバージョンが出力に表示されるはずです。

5. Maven 出力が表示されたら、Visual Studio Code を再起動します。マシンを再起動する必要がある場合もあります。Visual Studio Code ターミナルを開き、次のコマンドを実行します。

   ```
   mvn -v
   ```

   出力は、ステップ 4 の出力と同じであるはずです。Visual Studio Code 出力が異なる場合は、以下を試して設定が正しいことを確認します。

   - Visual Studio Code で PATH 変数を確認します。IDE 拡張機能によって、ローカルの PATH 変数とは異なるPATHように が変更されている可能性があります。拡張機能をアンインストールして、PATH から削除します。
   - Visual Studio Code でデフォルトのシェルを確認します。zsh 以外に設定している場合は、シェルに対して上記のステップを繰り返します。

Windows

Maven を Windows PATH に追加するには、次の手順を実行します。

1. Maven インストールディレクトリ、または Maven をインストールしたフォルダを見つけ、そのフォルダへのパスを保存します。
2. [環境変数] ウィンドウを開きます。

   a. Windows キーを押して検索バーを開きます。

   b. 「Edit environment variables for your account」と入力して選択します。
3. [環境変数] ウィンドウで、Path 変数を探します。Path 変数が既にある場合は、[編集...] を選択して更新します。Path 変数がない場合は、[新規...] を選択して追加します。
4. 表示される [環境変数名の編集] ウィンドウで、既存のパスをダブルクリックして編集するか、[新規] を選択して新しいパスエントリを追加します。

既存の Maven パスエントリを、ステップ 1 で保存したパスに置き換えるか、パスを新しいエントリとして追加します。パスの最後に、次の例のように、サフィックスとして \bin を追加します。

```
C:\Users\yourusername\Downloads\apache-maven-3.9.6-bin\apache-maven-3.9.6\bin
```

5. [OK] を選択してパスエントリを保存し、[環境変数] ウィンドウで再度 [OK] を選択します。
6. 新しいコマンドプロンプトを開き、次のコマンドを入力します。

```
mvn -v
```

Maven のバージョンが出力に表示されるはずです。

Amazon Q がコードをビルドできないのはなぜですか?

Amazon Q がコードをビルドする際に変換が失敗する場合は、Amazon Q がコードをビルドする環境に合わせてプロジェクトが正しく設定されていない可能性があります。ビルド設定またはコード実装を更新する必要があるかもしれません。

Amazon Q が提供するビルドログ出力を確認して、プロジェクトに加えられる変更があるかどうかを判断します。以下は、Amazon Q によるコードのビルドを妨げる可能性のある一般的な問題です。

pom.xml の絶対パスを削除する

pom.xml ファイルに絶対パスがある場合、Amazon Q は関連するファイルを見つけることができず、その結果、コードをビルドできない可能性があります。

以下は、pom.xml ファイルに含まれている可能性のある絶対パスの例です。

```
<toolspath>
    <path>/Library/Java/JavaVirtualMachines/jdk-11.0.11.jdk/Contents/Home/lib/
tools.jar</path>
</toolspath>
```

絶対パスを使用する代わりに、ポインタを使用して相対パスを作成できます。以下は、前の絶対パスを相対パスに置き換える方法の例です。

```
<toolspath>
    <path>${java.home}/../lib/tools.jar</path>
```

```
</toolspath>
```

ユニットテストのローカルデータベースまたは外部データベースを削除する

Amazon Q は、コードをビルドする際、プロジェクト内のすべてのユニットテストを実行します。ユニットテストがローカルデータベースまたは外部データベースを呼び出すと、Amazon Q はデータベースにアクセスできず、ビルドが失敗します。ビルドが失敗しないようにするには、変換を送信する前に、ユニットテストからデータベース呼び出しを削除するか、ユニットテストを削除する必要があります。

55 分経過後に変換が失敗したのはなぜですか?

コード変換ジョブが 55 分経過後に失敗した場合、コードのビルド時間がビルドの時間制限を超えている可能性があります。現在、コードのビルドには 55 分の制限があります。

ローカルビルドに 55 分以上かかる場合、コードを変換するには、プロジェクトのビルド時間を短縮します。ローカルビルドがコード変換を使用したビルドよりも速い場合は、環境が変わると障害が発生したり、かかる時間が長くなったりする可能性のあるタスクがないかプロジェクトをチェックします。長時間の実行が必要なテストケースを無効にすることを検討してください。また、セキュアな IDE 環境やインターネットから利用できないリソースにアクセスする試みに、タイムアウトを使用することも検討してください。

変換されたコードをダウンロードできないのはなぜですか?

変換の完了後にコードをダウンロードできない場合は、次のいずれかの問題が原因である可能性があります。Amazon Q で表示されるエラーに対応するトピックを参照してください。

トピック

- プロジェクトサイズを縮小する
- 30 日以内にコード差分をダウンロードする
- IDE でプロキシを設定する
- JetBrains プロキシ設定のワイルドカード文字を削除する

プロジェクトサイズを縮小する

変換が完了すると、Amazon Q は、アップグレードされたコードを含む 差分と、変更に関する情報が記載された変換の概要を含む出力アーティファクトを生成します。IDE で出力アーティファクトをダウンロードするには、出力アーティファクトが 1 GB 以下である必要があります。

出力アーティファクトが制限を超えると、アップグレードされたコードまたは変換の概要をダウンロードできなくなります。出力アーティファクトが大きくならないように、小さなプロジェクトを変換してみてください。問題が解決しない場合は、 にお問い合わせください サポート。Amazon Q サポート へのお問い合わせについては、「」を参照してくださいAmazon Q Developer を使用して とチャットする サポート。

30 日以内にコード差分をダウンロードする

アップグレードされたコードを含むコード差分ファイルは、変換完了後 30 日間のみ使用できます。変換が完了してから 30 日以上経過している場合は、変換を再起動して差分ファイルをダウンロードします。

IDE でプロキシを設定する

Amazon Q は、サービス所有の Amazon S3 バケットからアップグレードされたコードをダウンロードします。ダウンロードプロセスの一部では、SSL 証明書または TLS 証明書を使用して Amazon S3 と IDE 間の通信を確立します。プロキシサーバーを使用している場合、プロキシサーバーで使用される SSL 証明書または TLS 証明書は信頼されている必要があります。信頼されていない場合、Amazon Q はプロジェクトをアップロードできません。

コードをダウンロードするには、証明書を信頼するように IDE を設定したり、他のプロキシ設定を更新したりする必要がある場合があります。プロキシ設定の更新の詳細については、「IDE でプロキシを設定する」を参照してください。

JetBrains プロキシ設定のワイルドカード文字を削除する

JetBrains IDE でプロキシ設定を行っている場合、アップグレードされたコードのダウンロード時に次のエラーが表示されることがあります。

```
software.amazon.awssdk.core.exception.SdkClientException:
Unable to execute HTTP request: Dangling meta character '*' near index 0
```

これは、IDE の [No proxy for] フィールドにワイルドカード文字 (*) が存在することが原因で発生する可能性があります。Amazon Q で使用される Java SDK は、このフィールドのワイルドカードエントリをサポートしていません。

コードをダウンロードするには、[No proxy for] フィールドからワイルドカードを削除し、IDE を再起動します。プロキシをバイパスするホストを指定する必要がある場合は、ワイルドカードの代わりに正規表現を使用します。JetBrains IDE のプロキシ設定を更新するには、JetBrains ドキュメントの「HTTP Proxy」を参照してください。

コード変換ログにアクセスするにはどうすればよいですか?

JetBrains でログにアクセスする

JetBrains ログファイルにアクセスする方法については、JetBrains ドキュメントの「Locating IDE log files」を参照してください。

で Amazon Q によって出力されたログを検索するにはJetBrains、IDE ログで次の文字列を検索します。

```
software.aws.toolkits.jetbrains.services.codemodernizer
```

コード変換ログは、上記の文字列で始まります。Maven によって生成されたログは [Run] タブに表示され、ログエントリの前後に上記の文字列があります。

Visual Studio Code でログにアクセスする

で Amazon Q によって出力されたログを検索するにはVS Code、次の手順を実行します。

1. 上部のナビゲーションバーで [ビュー] を選択し、[コマンドパレット] を選択します。
2. 表示されるコマンドパレットで「`Amazon Q: View Logs`」を検索します。
3. ログが IDE で開きます。ログファイルで `CodeTransformation` を検索するには、`CMD + F` または `Control + F` を使用します。

VS Code のコード変換のログには、`CodeTransformation:` というプレフィックスが付いています。以下は、Maven コピー依存関係エラーに対して VS Code で生成されたログの例です。

```
2024-02-12 11:29:16 [ERROR]: CodeTransformation: Error in running Maven copy-
dependencies command mvn = /bin/sh: mvn: command not found
```

変換ジョブ ID を見つけるにはどうすればよいですか?

JetBrains でジョブ ID を見つける

JetBrains で変換ジョブ ID を見つけるには、[変換ハブ] の [変換の詳細] タブに移動し、[ジョブステータスの表示] (時計) アイコンをクリックします。

Visual Studio Code でジョブ ID を見つける

VS Code で変換ジョブ ID を検索するには、[変換ハブ] に移動し、[ジョブステータスの表示] (時計) アイコンをクリックします。

Amazon Q Developer を使用した .NET アプリケーションの変換

Amazon Q Developer は、生成 AI を活用したリファクタリングワークフローを通じて、Windows ベースの .NET アプリケーションを Linux 互換のクロスプラットフォーム .NET アプリケーションに移植できます。Amazon Q では、クロスプラットフォーム .NET アプリケーションの古いバージョンを新しいバージョンにアップグレードすることもできます。

.NET ソリューションまたはプロジェクトを変換するために、Amazon Q はコードベースを分析し、アプリケーションの移植に必要な更新を決定し、変換を開始する前に変換計画を生成します。この分析中、Amazon Q は .NET ソリューションまたはプロジェクトを変換プランで表示できるコードグループに分割します。コードグループは、動的リンクライブラリ (DLL) や実行可能ファイルなどのビルド可能なコード単位を一緒に生成するプロジェクトとそのすべての依存関係です。

変換中、Amazon Q はstep-by-stepの更新を提供します。アプリケーションを変換すると、Amazon Q は提案された変更を含む概要を差分ビューで生成し、オプションで変更を受け入れる前に検証できます。変更を受け入れると、Amazon Q は .NET ソリューションまたはプロジェクトをインプレース更新します。

Amazon Q は 4 つのキータスクを実行して、.NET アプリケーションを Linux に移植します。

- 言語バージョンのアップグレード – 古い C# バージョンのコードを Linux 互換の C# バージョンに置き換えます。
- .NET Framework からクロスプラットフォーム .NET への移行 – プロジェクトとパッケージを Windows 依存の .NET Framework から Linux と互換性のあるクロスプラットフォーム .NET に移行します。
- Linux 互換性のためのコードを書き換える – 非推奨および非効率的なコードコンポーネントをリファクタリングおよび書き換えます。
- Linux 互換性準備状況レポートの生成 – Linux でコードを構築して実行するためにユーザーの介入が必要なオープンエンドタスクの場合、Amazon Q は変換後にアプリケーションを設定するために必要なアクションの詳細なレポートを提供します。

Amazon Q が .NET 変換を実行する方法の詳細については、「仕組み」を参照してください。

トピック

- クォータ
- Visual Studio での Amazon Q Developer を使用した .NET アプリケーションの移植
- Amazon Q Developer が .NET アプリケーションを変換する方法

- IDE での .NET 変換に関する問題のトラブルシューティング

クォータ

IDE の Amazon Q による .NET 変換では、次のクォータが維持されます。

- ジョブあたりのコード行 – Amazon Q が特定の変換ジョブで変換できるコード行の最大数。これは、.NET 変換の毎月の合計制限でもあります。
- 同時ジョブ – 同時に実行できる変換ジョブの最大数。このクォータは、Java 変換を含む IDE 内のすべての変換に適用されます。

リソース	クォータ
ジョブあたりのコード行	100,000 行のコード
同時ジョブ	ユーザーあたり 1 ジョブ AWS アカウントあたり 2 ジョブ

Visual Studio での Amazon Q Developer を使用した .NET アプリケーションの移植

Visual Studio の Amazon Q Developer を使用して、Windows ベースの .NET アプリケーションを Linux 互換のクロスプラットフォーム .NET アプリケーションに移植するには、以下の手順を実行します。

ステップ 1: 前提条件

続行する前に、「IDE で Amazon Q のセットアップ」の手順が完了していることを確認してください。

.NET 変換ジョブを開始する前に、アプリケーションの次の前提条件を満たしていることを確認してください。

- アプリケーションには、C# で記述された .NET プロジェクトのみが含まれます。
- アプリケーションには Microsoft が作成した NuGet パッケージの依存関係のみがあります
- アプリケーションは UTF-8 文字のみを使用します。アプリケーションで non-UTF-8文字を使用している場合でも、Amazon Q はコードの変換を試みます。

- アプリケーションがインターネットインフォメーションサービス (IIS) に依存している場合、デフォルトの IIS 設定のみが使用されます。
- Amazon Q は、選択したプロジェクトのタイプとその依存関係を評価してコードグループを作成します。コードグループは、次のプロジェクトタイプのみを持つことができます。
  - コンソールアプリケーション
  - クラスライブラリ
  - ウェブ API
  - WCF サービス
  - Model View Controller (MVC) と Single Page Application (SPA) のビジネスロジックレイヤー
  - テストプロジェクト

> **Note**
>
> Amazon Q は、Razorビューや WebForms ASPX ファイルなどの UI レイヤーコンポーネントの変換をサポートしていません。Amazon Q がソリューションまたはプロジェクト内の UI レイヤーコンポーネントを検出すると、UI レイヤーコンポーネントを除外して部分変換を実行し、ターゲットの .NET バージョンでコードを構築できるようにさらにリファクタリングする必要がある場合があります。

ステップ 2: アプリケーションを変換する

.NET ソリューションまたはプロジェクトを変換するには、次の手順を実行します。

1. 変換する C# ベースのソリューションまたはプロジェクトを Visual Studio で開きます。
2. エディタで任意の C# コードファイルを開きます。
3. Solution Explorer を選択します。
4. Solution Explorer から、変換するソリューションまたはプロジェクトを右クリックし、Amazon Q Developer でポートを選択します。
5. Amazon Q Developer のポートウィンドウが表示されます。

   選択したソリューションまたはプロジェクトは、「変換するソリューションまたはプロジェクトを選択する」ドロップダウンメニューで選択されます。メニューを展開して、変換する別のソリューションまたはプロジェクトを選択できます。

.NET ターゲットの選択ドロップダウンメニューで、アップグレードする .NET バージョンを選択します。

6. 確認 を選択して変換を開始します。
7. Amazon Q はコードの変換を開始します。生成する変換プランを表示して、アプリケーションの変換方法の詳細を確認できます。

   Transformation Hub が開き、変換の期間中の進行状況をモニタリングできます。Amazon Q がジョブ変換のスタートアップ待機ステップを完了したら、変換の期間中、プロジェクトまたはソリューションから移動できます。
8. 変換が完了したら、Transformation Hub に移動し、差分の表示を選択して、差分ビューで Amazon Q から提案された変更を確認します。
9. Amazon Q が行った変更の詳細については、「コード変換の概要を表示する」を選択します。Download summary as .md を選択して、変換の概要をダウンロードすることもできます。

   コードグループテーブルのいずれかの項目が Linux 移植ステータスで入力を必要とする場合は、Linux でアプリケーションを実行するためにいくつかのファイルを手動で更新する必要があります。

   a. Actions ドロップダウンメニューから、Linux 準備状況レポートのダウンロードを選択します。
   b. .csv ファイルが開き、アプリケーションが Linux 互換になる前に完了する必要があるプロジェクトまたはソリューションへの変更がすべて表示されます。これには、更新する必要があるプロジェクトとファイル、更新する項目の説明、問題の説明が含まれます。Linux の準備状況の問題に対処する方法については、「レコメンデーション」列を参照してください。
10. ファイルを更新するには、アクションドロップダウンメニューから変更を受け入れるを選択します。

Amazon Q Developer が .NET アプリケーションを変換する方法

> **Note**
>
> IDE での Amazon Q を使用した .NET アプリケーションの変換はプレビュー中であり、変更される可能性があります。

Amazon Q Developer での .NET 変換の仕組みの詳細については、以下のセクションを参照してください。

アプリケーションの分析と変換計画の生成

変換を開始する前に、Amazon Q はコードをローカルに構築して、ビルド可能で変換用に正しく設定されていることを確認します。次に、Amazon Q は、 上の安全で暗号化されたビルド環境にコードをアップロードし AWS、コードベースを分析し、アプリケーションを移植するために必要な更新を決定します。

この分析中、Amazon Q は .NET ソリューションまたはプロジェクトをコードグループに分割します。コードグループは、動的リンクライブラリ (DLL) や実行可能ファイルなどのビルド可能なコード単位を一緒に生成するプロジェクトとそのすべての依存関係です。変換するすべてのプロジェクトの依存関係を選択していなくても、Amazon Q は選択したプロジェクトの構築に必要な依存関係を決定し、変換したアプリケーションが構築可能で使用できるようにも変換します。

コードを分析した後、Amazon Q は、変換されるコードグループとその依存関係のリストなど、提案された変更の概要を示す変換計画を生成します。

アプリケーションの変換

変換を開始するため、Amazon Q は安全なビルド環境でコードを再度構築し、リモートで構築できるようにします。その後、Amazon Q はアプリケーションの移植を開始します。最下位レベルの依存関係から順に動作します。Amazon Q が依存関係の移植で問題が発生した場合、変換を停止し、エラーの原因に関する情報を提供します。

変換には、アプリケーションに対する以下の更新が含まれます。

- 古い C# バージョンのコードを Linux 互換の C# バージョンに置き換える
- 以下を含む、.NET Framework のクロスプラットフォーム .NET へのアップグレード
  - パッケージ、ライブラリ、APIs の特定と反復置換
  - NuGet パッケージと APIsのアップグレードと置き換え
  - クロスプラットフォームランタイムへの移行
  - ミドルウェアのセットアップとランタイム設定の更新
  - プライベートパッケージまたはサードパーティーパッケージの置き換え
  - IIS および WCF コンポーネントの処理
  - ビルドエラーのデバッグ

- 既存のコードを移植するための非推奨および非効率的なコードのリファクタリングや書き換えなど、Linux との互換性のためのコードを書き換える

変換の概要を確認し、変更を受け入れる

変換が完了すると、Amazon Q は、変更されたファイル、更新されたパッケージ、変更された APIs の数など、アプリケーションに対して提案された更新に関する情報を含む変換の概要を提供します。影響を受けるファイルやファイルの一部、ビルドの試行中に発生したエラーなど、失敗した変換にフラグを付けます。ビルドログを使用してビルド概要を表示して、どのような変更が行われたかを確認することもできます。

また、変換の概要には Linux 移植ステータスも表示されます。これは、アプリケーションの Linux と互換性を持たせるために追加のユーザー入力が必要かどうかを示します。コードグループ内のいずれかの項目で入力が必要な場合は、ビルド時に Amazon Q が対処できなかった Windows 固有の考慮事項を含む Linux 準備状況レポートをダウンロードします。コードグループまたはファイルに対して入力が必要な場合は、レポートを確認して、まだ行う必要がある変更の種類の詳細を確認し、該当する場合は、コードの更新方法に関する推奨事項を確認してください。これらの変更は、Linux でアプリケーションを実行する前に手動で行う必要があります。

ファイルへのインプレース更新として受け入れる前に、差分ビューで Amazon Q が行った変更案を確認できます。ファイルを更新し、Linux 準備状況レポートの項目に対応すると、アプリケーションはクロスプラットフォーム .NET で実行できるようになります。

IDE での .NET 変換に関する問題のトラブルシューティング

> **Note**
>
> IDE での Amazon Q を使用した .NET アプリケーションの変換はプレビュー中であり、変更される可能性があります。

以下のセクションでは、Amazon Q Developer を使用した IDE での .NET 変換に関する一般的な問題のトラブルシューティングを行います。

ジョブの進行状況を確認するにはどうすればよいですか？

Amazon Q が Transformation Hub のステップに長い時間を費やしているように見える場合は、出力ログでジョブがまだアクティブかどうかを確認できます。診断メッセージが生成されている場合、ジョブはまだアクティブです。

出力を確認するには、Visual Studio の出力タブを選択します。Show output from: メニューで、Amazon Q Language Client を選択します。

次のスクリーンショットは、Amazon Q が変換中に生成する出力の例を示しています。

一部のプロジェクトが変換対象として選択されていないのはなぜですか？

Amazon Q は、サポートされているプロジェクトタイプを C# 言語でのみ変換できます。現在、Amazon Q は、VB.NET または F# 言語で記述された UI レイヤーコンポーネントまたはプロジェクトの移植をサポートしていません。.NET プロジェクトを変換するためのサポートされているプロジェクトタイプとその他の前提条件のリストについては、「」を参照してくださいステップ 1: 前提条件。

プロジェクトまたはソリューションが変換されていない場合、サポートを受けるにはどうすればよいですか？

自分で問題をトラブルシューティングできない場合は、 または AWS アカウント チームに連絡してサポート サポートケースを送信できます。

サポートを受けるには、失敗したジョブを AWS が調査できるように、変換ジョブ ID を指定します。変換ジョブ ID を検索するには、Visual Studio の出力タブを選択します。Show output from: メニューで、Amazon Q Language Client を選択します。

ファイアウォールが変換ジョブに干渉しないようにするにはどうすればよいですか？

組織がファイアウォールを使用している場合、Visual Studio の変換を妨げる可能性があります。Node.js のセキュリティチェックを一時的に無効にして、変換の実行を妨げる原因をトラブルシューティングまたはテストできます。

環境変数は、重要なセキュリティチェックNODE_TLS_REJECT_UNAUTHORIZEDを制御します。"0" NODE_TLS_REJECT_UNAUTHORIZEDに設定すると、Node.js による不正な TLS/SSL 証明書の拒否が無効になります。つまり、次のようになります。

- 自己署名証明書が受け入れられます

- 期限切れの証明書は許可されます
- ホスト名が一致しない証明書は許可されます
- その他の証明書検証エラーは無視されます

プロキシが自己証明書を使用している場合は、 を無効にする代わりに、次の環境変数を設定できますNODE_TLS_REJECT_UNAUTHORIZED。

```
NODE_OPTIONS = –use-openssl-ca
NODE_EXTRA_CA_CERTS = Path/To/Corporate/Certs
```

それ以外の場合は、 を無効にするためにプロキシが使用する CA 証明書を指定する必要がありますNODE_TLS_REJECT_UNAUTHORIZED。

Windows で NODE_TLS_REJECT_UNAUTHORIZED を無効にするには：

1. スタートメニューを開き、環境変数を検索します。
2. システム環境変数の編集 を選択します。
3. システムプロパティウィンドウで、環境変数を選択します。
4. システム変数で、新規 を選択します。
5. 変数名を NODE_TLS_REJECT_UNAUTHORIZED に設定し、変数値を 0 に設定します。
6. OK を選択して変更を保存します。
7. Visual Studio を再起動します。

Amazon Q Developer を使用した機能の開発

> **Note**
>
> 機能開発に関するヘルプは、チャットで直接リクエストできます。/dev を使用すると、リクエストはチャットにリダイレクトされます。

Amazon Q Developer は、統合開発環境 (IDE) でのコード機能の開発やプロジェクトへのコード変更に役立ちます。実行するタスクについて説明すると、Amazon Q は現在のプロジェクトやワークスペースのコンテキストを使用して、変更を実装するためのコードを生成します。Amazon Q は、AWS プロジェクトまたは独自のアプリケーションの構築に役立ちます。

まったく新しいプロジェクトを開始することも、IDE で開いているプロジェクトに取り組むこともできます。既存のプロジェクトでの開発では、Amazon Q はワークスペースルート内のすべての関連ファイルをコンテキストとして使用してコードを生成します。Amazon Q は、`.gitignore` ファイルで定義されたファイルまたはフォルダをフィルタリングし、サポートされているファイルタイプのみを使用してコードを生成します。サポートされているファイルタイプのリストについては、ご利用の IDE の GitHub リンクを参照してください。

- JetBrains IDE でサポートされているファイルタイプ – https://github.com/aws/aws-toolkit-jetbrains/blob/main/plugins/core/jetbrains-community/src/software/aws/toolkits/jetbrains/services/telemetry/TelemetryUtils.kt
- Visual Studio Code でサポートされているファイルタイプ – https://github.com/aws/aws-toolkit-vscode/blob/master/packages/core/src/shared/filetypes.ts

開始するには、新規または既存のプロジェクトを開き、Amazon Q チャットパネルに **/dev** と入力します。新しいチャットタブが開き、Amazon Q を操作して機能の新しいコードを生成します。

トピック

- /dev で機能を開発する
- ベストプラクティス
- タスクの例
- クォータ
- Amazon Q '/dev' コマンドによって生成されたコード提案のテスト
- Amazon Q Developer による機能開発に関する問題のトラブルシューティング

/dev で機能を開発する

> **Note**
>
> 機能開発に関するヘルプは、チャットで直接リクエストできます。/dev を使用すると、リクエストはチャットにリダイレクトされます。

IDE で Amazon Q を使用してコードタスクを操作するには、次の手順を実行します。

1. IDE で、機能を開発する新規または既存のプロジェクトまたはワークスペースを開きます。

2. Amazon Q アイコンをクリックして、Amazon Q チャットパネルを開きます。
3. Amazon Q チャットパネルに **/dev** と入力し、次に実行するタスクまたは解決する問題の説明を入力します。タスクの概要を記載したり、詳細を追加したりできます。Amazon Q は、説明とプロジェクト内のコードを使用してコードを生成します。

   以下は、Amazon Q に実装をリクエストできるコード変更の例です。

   /dev Create a new REST API endpoint /api/authenticate to handle user authentication. This endpoint should accept POST requests with user credentials and return a JWT token upon successful authentication. Additionally, update the user management system to integrate with the new authentication endpoint and enforce authentication for relevant API endpoints.

   また、**/dev** のみを入力して、タスクの例を表示することもできます。
4. 新しいタブが開きます。まだ入力していない場合は、タスクまたは問題の説明を入力します。
5. Amazon Q は、説明したタスクのコードの生成を開始します。コード生成中、Amazon Q は、新しいコードを提案するために使用している手順、使用または更新しているファイルの概要を提供します。これには数分かかることがあります。

   Amazon Q でコードの生成を停止する場合は、停止 を選択します。不完全なコード生成は、このタスクのコード生成クォータにカウントされます。生成を停止した後、別のタスクの説明を入力して、タスクを引き続き処理することができます。

   オプションで、Amazon Q はチャットウィンドウで提示する前に、別の開発環境でコード提案をテストできます。詳細については、「「/dev」提案のテスト」を参照してください。
6. Amazon Q がタスクのコードを生成した後、提案された変更を含むファイルのリストを提供します。ファイルを選択すると、変更を含むファイル差分が表示されます。
7. 提案されたすべてのコード変更でファイルを更新するには、すべての変更を受け入れるを選択します。個々のファイルへの変更を承認および拒否することもできます。個々のファイル変更を受け入れるには、ファイル名にカーソルを合わせ、緑色のチェックマークを選択します。変更を拒否するには、赤い X を選択します。また、残りの変更を受け入れるを選択して、まだ承認していない変更を適用することもできます。

   コード変更を受け入れると、元に戻すことはできません。コード変更を拒否する場合は、ファイル名にカーソルを合わせ、矢印を選択してコード変更を元に戻すことができます。その後、コード変更を再度承認または拒否するオプションがあります。

8. 提案された変更に満足できない場合は、改善点に関するフィードバックを提供できます。フィードバックの提供と再生成を選択し、Amazon Q で行う変更を記述します。

   Amazon Q は、フィードバックに基づいて新しいコードを生成します。生成されたコードが要件を満たしたら、すべての変更を受け入れるか、個々のファイルを受け入れるを選択してプロジェクトまたはワークスペースのコードを更新します。

9. コードを更新すると、Amazon Q から新しいタスクでの作業を開始するように求められます。

   プロジェクトまたはワークスペース内の別のタスクで作業を開始するには、[はい、別のタスクがあります] を選択します。以前の会話と生成されたコードは、新しいタスクのコンテキストとして使用されません。

   機能開発セッションを終了するには、いいえ、ありがとうございます。Amazon Q でチャットを続けるか、別のワークフローを開始するには、新しいチャットタブを開きます。

ベストプラクティス

Amazon Q で機能開発を最大限に活用するには、次のベストプラクティスに従います。

- コードが何を達成するかの具体的な説明など、新機能またはコードの変更の詳細な説明を提供します。これにより、Amazon Q は包括的で実装可能なコード変更を提案できます。例については「タスクの例」を参照してください。
- 機能で、一度に 5 つ以上のファイルの更新を必要としないようにしてください。Amazon Q により大規模な変更をリクエストすると、機能実装の品質と管理性に影響する可能性があります。ファイル差分に多くのファイルへの変更が含まれている場合は、機能の説明の範囲を狭めてみてください。

タスクの例

以下に、Amazon Q に実行を依頼できる機能開発タスクの例と、それに対応する説明を示します。

- 応答型レイアウトの CSS スタイルを更新する: CSS スタイルを更新することで、アプリケーションのレイアウトの応答性を向上させます。さまざまな画面サイズに対応するレイアウトの調整に重点的に取り組み、さまざまなデバイスで表示を最適化します。
- ユーザープロファイル画像のアップロードを修正する: ユーザーがプロファイル画像をアップロードできない問題を解決します。ファイルアップロードプロセスを調べ、ファイルタイプの検証が適切に行われるようにし、アップロードの成功を妨げるサーバー側のエラーに対処します。

- コードをリファクタリングしてコードの可読性を高める: リファクタリングすることで、特定のコードモジュールの可読性を向上させます。複雑な関数を管理しやすい小さな単位に分割し、意味のある変数名と関数名を適用することで、わかりやすいコードにします。
- 問い合わせフォームに入力検証を実装する: クライアント側とサーバー側の入力検証を問い合わせフォームに追加して、空のデータや無効なデータの送信を防ぎます。検証に失敗した場合は、適切なエラーメッセージをユーザーに表示します。
- ナビゲーションメニューの壊れたリンクを直す: アプリケーションのナビゲーションメニューの壊れたリンクを調査して修正します。URL またはルートを更新して、すべてのナビゲーションリンクが正しいページにつながることを確認します。
- ページの読み込みを高速化するために画像の読み込みを最適化する: アプリケーションの主要ページの画像読み込みを最適化して、ページの読み込み時間を短縮します。遅延読み込みや非同期読み込みを実装して、見えているコンテンツの表示を優先します。
- 重要な API エンドポイントにエラーのログ記録を追加する: 重要な API エンドポイントにログ記録を実装することで、エラー処理を強化します。デバッグとトラブルシューティングを容易にするために、リクエストパラメータやスタックトレースなどの関連するエラーの詳細を記録します。
- API エンドポイントのドキュメントを更新する: 既存の API エンドポイントのドキュメントを確認して更新し、最近の変更や追加を反映します。API を使用するデベロッパー向けに、情報の正確性と完全性を確保します。
- データベースクエリをリファクタリングして効率を高める: データベースクエリを分析して最適化し、効率を向上させ、リソースの消費量を減らします。冗長クエリを特定して取り除き、インデックス作成を最適化し、必要に応じてキャッシュ戦略を検討します。

クォータ

Amazon Q による機能開発では、次のクォータが維持されます。

- タスクあたりのコード生成 – 最初のコード生成を含む、特定の開発タスクのコードを再生成するためのフィードバックを提供できる回数。このクォータは、新しいタスクを開始するたびにリセットされます。
- コードプロジェクトサイズ – Amazon Q が新しいコードを生成するコンテキストとして使用できるコードファイルまたはフォルダの最大サイズ。

クォータ

リソース	クォータ
タスクあたりのコード生成回数	10
コードプロジェクトサイズ	200 MB (非圧縮) 50 MB (圧縮)

Amazon Q '/dev' コマンドによって生成されたコード提案のテスト

デフォルトでは、 **/dev** コマンドを使用する場合、Amazon Q はチャットで表示されるコードを自動的にテストしません。コードを保存してから個別にテストする必要があります。Amazon Q は、作業を完了するために必要なツールを備えた環境がないため、コードをテストできません。

Amazon Q がコードをテストできるようにするには、devfile を指定して dev 環境を起動できるようにする必要があります。

> **Note**
>
> **/dev** コマンドと の詳細についてはAmazon Q Developer Agent for software development、「」を参照してくださいAmazon Q Developer を使用した機能の開発。

開発環境とは

開発環境は、Amazon Q がチャットで提示する前にコードをテストできる Docker コンテナです。コンテナには、コードをテストするために必要なプリインストールされたツール、言語、フレームワークが含まれています。

Amazon Q はサンドボックスで Docker コンテナを起動します。つまり、Amazon Q が生成する可能性のある問題のあるコードから保護するために、コンテナは AWS サービスから分離されます。

devfile とは

devfile は、devfile 仕様に準拠する YAML ファイルです。これは、コンテナ化された開発環境を定義するオープンスタンダード、つまり短期間の dev 環境です。

devfile は以下を記述します。

- 開発環境として使用する Docker コンテナイメージの名前と場所。
- 開発環境の起動時に自動的に実行するコマンド。

devfile には他のタイプの情報を含めることができますが、前述の 2 つは Amazon Q に関連する情報です。

独自の devfile を指定するか、Amazon Q に生成させることができます。

仕組み

Amazon Q チャットパネルに**/dev**コマンドを入力するたびに、Amazon Q はプロジェクトのルートで devfile を検索します。

devfile が存在する場合、Amazon Q:

1. サンドボックス内の dev 環境を起動します AWS。
2. コード提案を生成します (ただし、まだ提示されていません)。
3. devfile の手順を使用して、dev 環境で提案を構築してテストします。
4. テストに合格しなかった場合は、提案を繰り返し行います。Amazon Q は 3 つの反復を実行しますが、これは設定できません。
5. チャットウィンドウにテスト済みのコードを表示します。
6. 開発環境を終了します。

devfile が存在しない場合、Amazon Q は生成するかどうかを尋ねます。「はい」と言うと、Amazon Q はプロジェクトのコンテキストに基づいて devfile を生成します。この devfile の詳細については、「」を参照してくださいAmazon Q Developer の devfile の例。devfile を生成した後、Amazon Q は使用を開始しません。devfile を呼び出すには、別の**/dev**コマンドを入力する必要があります。

サンドボックスの仕様

開発環境が起動されるサンドボックスには、2 つの vCPUsと 4 GB のメモリがあります。CPU とメモリは設定できません。

Amazon VPC のサポート

Amazon Q は、Amazon Virtual Private Cloud (Amazon VPC) 内での開発環境の実行をサポートしていません。詳細については、「Amazon Virtual Private Cloud ユーザーガイド」の「Amazon VPC とは?」を参照してください。

トピック

- Amazon Q での devfile の設定
- Amazon Q のコード提案のテストを無効にする
- Amazon Q Developer の devfile の例
- ユニバーサルイメージの内容

Amazon Q での devfile の設定

Amazon Q が**/dev**コマンドに応答して生成するコード提案をテストできるようにするには、devfile を設定する必要があります。次のいずれかを行うことができます。

- Amazon Q にプロジェクトのコンテキストに基づいて devfile を生成してもらいます。
- devfile を自分で作成します。

Amazon Q での devfiles とテストの詳細については、「」を参照してくださいAmazon Q '/dev' コマンドによって生成されたコード提案のテスト。

Amazon Q で devfile を生成するには

1. IDE で、 **/dev** コマンドを実行します。Amazon Q はテストされていない提案を生成し、チャットウィンドウで表示します。
2. 提案の最後に、devfile の生成を選択してコードを作成します。Amazon Q は devfile を生成します。
3. ファイルをプロジェクトのルートフォルダ`devfile.yaml`に として保存します。

   devfile を保存すると、Amazon Q は次回**/dev**コマンドが使用されたときにその devfile の使用を開始します。
4. （オプションですが推奨) devfile をコミットしてプッシュし、他のユーザーが使用できるようにします。

独自の devfile を設定するには

1. 基本的な devfile から始めます。Amazon Q に生成させるか (前の手順を参照）、 の例のいずれかを使用できますAmazon Q Developer の devfile の例。
2. 次のセクション「」のガイドラインに従って、必要に応じて devfile Guidelines for configuring your own devfile を変更します。

3. ファイルをプロジェクトのルートフォルダdevfile.yamlに として保存します。

   devfile を保存すると、Amazon Q は次回**/dev**コマンドが使用されたときにその devfile の使用を開始します。

4. （オプションですが推奨) devfile をコミットしてプッシュし、他のユーザーが使用できるようにします。

独自の dev ファイルを設定するためのガイドライン

- で指定された Docker イメージは、DockerHub などのパブリックイメージレジストリまたはパブリック Amazon ECR レジストリに存在するimage:必要があります。プライベートレジストリはサポートされていません。
- devfile は devfile 2.2.0 スキーマに準拠している必要があります。
- devfile 2.2.0 スキーマ内のすべてのオブジェクトがサポートされているわけではありません。Amazon Q はinstall、、build、および test コマンドのみをサポートします。他のコマンドを含めることはできますが、実行されません。
- のコマンドは、合計で 5 分未満で実行commands:する必要があります。5 分後にタイムアウトします。
- アーティファクトを参照する必要がある場合は、PyPI や npmjs.com などのパブリックアーティファクトリポジトリに存在する必要があります。プライベートアーティファクトリポジトリはサポートされていません。
- Amazon S3 バケットなど、認証されたアクセスを必要とするリソース AWS （またはリソース) を参照しないでください。 Amazon S3 これは、開発環境の Docker コンテナがサンドボックス化されており、インターネット上のパブリックリソースにのみアクセスできるためです。

Amazon Q のコード提案のテストを無効にする

Amazon Q が開発環境でコード提案をテストできるようにする機能をオフにするには、IDE の設定に移動し、Amazon Q: Q /dev がコードとテストコマンドの設定を実行できるようにするをオフにします。（設定名の下にある settings.json リンクで編集を選択し、対応する JSON 設定を に設定falseして設定をオフにする必要がある場合があります）。

この設定をオフにすると、次のようになります。

- IDE で **/dev** コマンドを実行すると、Amazon Q はコードを構築するための devfile の生成ボタンを表示しなくなります。

- Amazon Q は、devfile を探したり、dev 環境を起動したり、生成したコード提案をテストしたりしなくなりました。

> **Note**
>
> Amazon Q Developer コンソールでは、開発環境の使用を有効または無効にすることはできません。

Amazon Q での devfiles とテストの詳細については、「」を参照してくださいAmazon Q '/dev' コマンドによって生成されたコード提案のテスト。

Amazon Q Developer の devfile の例

次の例は、devfile 仕様に準拠し、Amazon Q での使用に適した devfile を示しています。例については、このページの最後に説明します。

devfiles とその Amazon Q での使用の詳細については、「」を参照してくださいAmazon Q '/dev' コマンドによって生成されたコード提案のテスト。

Python の例

次の例は、Python プロジェクトで使用できる devfile を示しています。

```
schemaVersion: 2.0.0
components:
  - name: dev
    container:
      image: public.ecr.aws/aws-mde/universal-image:latest
commands:
  - id: install
    exec:
      component: dev
      commandLine: "pip3.11 install -r requirements.txt"
  - id: test
    exec:
      component: dev
      commandLine: "python3.11 manage.py makemigrations && python3.11 manage.py test"
```

ノードの例

次の例は、Node.js プロジェクトで使用できる devfile を示しています。

```
schemaVersion: 2.0.0
components:
  - name: dev
    container:
      image: public.ecr.aws/aws-mde/universal-image:latest
commands:
  - id: install
    exec:
      component: dev
      commandLine: "npm install"
  - id: build
    exec:
      component: dev
      commandLine: "npm run build"
  - id: test
    exec:
      component: dev
      commandLine: "npm run test"
```

例の説明

次の表は、前述の devfile の例の各行またはコードブロックを示しています。

行またはコードブロック	説明
`schemaVersion: 2.0.0`	devfile スキーマのバージョンを指定します。Amazon Q はスキーマバージョン 2.0.0 のみをサポートしています。
`components:`	開発環境にデプロイするコンポーネントのリスト。

`- name: dev`
`  container:`
`  image: public.ecr.aws/aws-mde/universal-image:latest`
	パブリック Amazon Elastic Container Registry (Amazon ECR) で利用可能な最新のユニバーサルイメージ Docker イメージを開発環境として使用することを示します。このイメージのバージョンを変更することも、イメージを独自のイ

行またはコードブロック	説明
	メージに置き換えることもできます。詳細については、「ユニバーサルイメージの内容」を参照してください。
`commands:`	実行するコマンドのリスト。

`- id: install|build|test`
`exec:`
`component: dev`
`commandLine: " cli-instructions "`
	install、build、または test コマンド。Amazon Q は、開発環境、ユニバーサルイメージ (または で指定されている任意のイメージ) で *cli-instructions* 手順を実行しますimage:。*cli-instructions* は、プロジェクトのコンテキストによって異なります。これらのコマンドは、ニーズに合わせて変更できます。

ユニバーサルイメージの内容

Amazon Q が devfile を生成すると、ユニバーサルイメージを指すように devfile image:の設定が設定されます。ユニバーサルイメージは、Amazon Q がコード提案をテストするために使用できるプリインストールされたフレームワーク、言語、ツールを含むパブリック Docker イメージです。

ユニバーサルイメージには 2 つのサポートされているバージョンがあります。

- 3.0
- 4.0

> **Note**
>
> 最新のユニバーサルイメージは 3.0 (4.0 ではなく) にマッピングされます。

ユニバーサルイメージのバージョン 3.0 および 4.0 にインストールされているパッケージのリストについては、次の表を参照してください。

Amazon は、バージョニングスキーマ列のバージョンスキーマに従って、ユニバーサルイメージパッケージを定期的にパッチ適用します。

テーブルにリストされているパッケージの新しいバージョンを使用している場合、または別のパッケージセットが必要な場合は、2 つのオプションがあります。devfile の `commands:` セクションに必要なパッケージをインストールする方法を記述するコマンドを追加するか、 `image:` フィールドに正しいパッケージを含む独自の Docker イメージを指定できます。

コード提案のテストの詳細については、「」を参照してくださいAmazon Q '/dev' コマンドによって生成されたコード提案のテスト。

ユニバーサルイメージ 3.0 (最新)

パッケージ	バージョニングスキーマ
aws-cli	2.x
docker-compose	2.x
dotnet	6.x、7.x
golang	1.21.x
java	corretto11.x、corretto17.x
nodejs	18.x、20.x
openssl	3.x
php	8.2.x
python	3.x
ruby	3.x
terraform	1.x

ユニバーサルイメージ 4.0

パッケージ	バージョニングスキーマ
aws-cli	2.x
docker-compose	2.x
dotnet	8.x
golang	1.22.x
java	corretto21.x
nodejs	20.x
php	8.2.x
python	3.x
ruby	3.x
terraform	1.x

Amazon Q Developer による機能開発に関する問題のトラブルシューティング

以下の情報は、Amazon Q Developer で IDE の機能を開発する際の一般的な問題のトラブルシューティングに役立ちます。

会話 ID を見つけるにはどうすればよいですか?

Amazon Q での機能開発中に発生する問題のサポートを受けるには、会話 ID が必要になる場合があります。エラーメッセージによっては、会話 ID が Amazon Q からのメッセージに表示されます。表示されない場合は、次のステップを使用して IDE で会話 ID を見つけます。

Visual Studio Code

1. コマンドパレットを開き、次のコマンドを実行します。

```
Amazon Q: View Logs.
```

2. 出力コンソールが VS Code ターミナルで開きます。次の文字列を検索します。

```
Amazon Q Developer Agent for software development Conversation ID:
```

会話 ID はコロンの後に表示されます。タイムスタンプが、ID が必要な会話に対応していることを確認します。

JetBrains

1. IDE の上部で、[Help]、[Show Log in Finder] の順に選択します。

JetBrains IDE とオペレーティングシステムによっては、ログメニューテキストが異なる場合があります。詳細については、IntelliJ IDEA ドキュメントの「Locating IDE log files」を参照してください。

2. ファイルファインダーが開きます。[idea.log] を選択して、ログファイルを開きます。
3. 次の文字列を検索します。

```
Amazon Q Developer Agent for software development Conversation ID:
```

会話 ID はコロンの後に表示されます。タイムスタンプが、ID が必要な会話に対応していることを確認します。

Amazon Q を使用したユニットテストの生成

> **Note**
>
> チャットで直接ユニットテストを生成するように に依頼できます。/test を使用すると、リクエストはチャットにリダイレクトされます。「」と質問してみてください**`generate unit tests for my application`**。

Amazon Q Developer エージェントは、ソフトウェア開発ライフサイクル全体でユニットテストの作成を自動化する AI を活用したユニットテスト生成機能を提供します。この機能は、開発者がコード品質を確保しながら機能開発を加速することに集中するのに役立ちます。

ユニットテスト用の Amazon Q エージェントは、次のステップを自動化します。

- テストケースの識別: エージェントは、ワークスペース内のプロジェクト構造、既存のコード、およびターゲットファイルを使用して、適切なテストケースを識別します。
- モックとスタブの作成: Amazon Q は、分離されたテストに必要なモックとスタブを生成します。
- テストコードの生成: エージェントは、特定されたテストケースに基づいてユニットテストを生成します。

ユニットテスト用の Amazon Q エージェントは、VS Code および JetBrains IDEs で Java および Python プロジェクトをサポートしています。各プログラミング言語でサポートされているテストフレームワークについては、「」を参照してくださいthe section called “ユニットテスト生成 (/test)”。

前提条件

ユニットテスト生成機能を使用するには、VS Code または JetBrains IDE 用の Amazon Q IDEs 拡張機能をダウンロードしてインストールする必要があります。the section called “Amazon Q のインストール”「」の手順に従って拡張機能を設定します。

> **Note**
>
> IDE でテストフレームワークが設定されている機能プロジェクトはオプションです。

/test を使用してユニットテストを生成する

> **Note**
>
> チャットで直接ユニットテストを生成するように に依頼できます。/test を使用すると、リクエストはチャットにリダイレクトされます。「」と質問してみてくださいgenerate unit tests for my application。

コードを記述した後、デベロッパーは主に 2 つの方法でユニットテスト生成を開始できます。

- Amazon Q チャット/testに と入力し、必要に応じて考慮するクラス、関数、またはメソッドの手順を指定します。
- コードのセクションを強調表示し、右クリックの Amazon Q メニューを開き、テストの生成オプションを選択します。

開発者が追加のプロンプトなしで /test コマンドを使用すると、エージェントは次の操作を行います。

- IDE で現在アクティブなファイルを調べます。
- テストディレクトリ内のこのアクティブなファイルに対応するユニットテストファイルが存在するかどうかを確認します。
- このようなテストファイルが見つからない場合は、新しいテストファイルが自動的に作成されます。新しいテストファイルの名前は元のファイルにちなんで付けられ、使用中のテストフレームワークに基づいてtestプレフィックスまたはサフィックスが追加され、テストディレクトリに配置されます。
- テストファイルが既に存在する場合は、新しいユニットテストを既存のファイルに追加します。

ユニットテスト生成プロセス

このプロセスには、以下のステップが含まれます。

1. 入力推論： Amazon Q は、ワークスペースコンテキストと設定されたテストフレームワークに基づいて、ターゲットコード、出力ファイル、適切なテストとモックを推論します。エージェントは、生成されたテストを既存または新しいテストファイルに配置する場所を特定し、テストするターゲットコードを決定します。開発者は、ターゲットコードとターゲットファイル名を指定するオプションの手順を提供できます。
2. コンテキストエンリッチメント： Amazon Q ユニットテストエージェントはオープンプロジェクトをコンテキストとして使用し、IDE プロジェクトのコードと依存関係に沿ったテストを生成できます。
3. テスト生成： Amazon Q エージェントは、ユニットテスト生成に適切な入力を推測するか、ユーザーがその情報を手動で提供できるようにします。次に、ユニットテストを生成します。このプロセス全体を通して、エージェントはチャットで進行状況の更新を提供します。

> **Note**
>
> - 大規模なプロジェクトを使用する場合、最初のプロジェクト分析には時間がかかることがあります。このプロセス中、進行状況インジケータは長期間 0% のままになることがあります。これは、Amazon Q がインデックスを作成し、プロジェクトファイルをアップロードする際の通常の動作です。
> - ユニットテストの生成は、一度に 1 つのファイルで実行されます。複数のファイルの場合は、各ファイルで個別に /test コマンドを使用します。

4. ユーザーレビュー： Amazon Q は、最終的に生成されたテストの差分を提供します。開発者は、生成されたテストを確認し、以下を選択できます。
   - 推奨されるテストを受け入れます。
   - 変更を拒否します。

使用法と考慮事項

ユニットテスト用の Amazon Q エージェントには、サブスクリプション階層に基づく一般的な使用制限が適用されます。これらの制限は、すべての Amazon Q 機能で共有されます。

> **Note**
>
> 各/testコマンドは、無料利用枠内の使用制限にカウントされます。

Amazon Q のサービス階層、クォータ、および料金の詳細については、「Amazon Q のサービス階層について」を参照してください。

AWS マネジメントコンソールの Amazon Q ダッシュボードを使用して、生成されたユニットテストの数、承諾されたユニットテストの数、承諾率などのユニットテスト生成メトリクスを追跡できます。クォータ制限機能の使用状況をモニタリングし、機能別の詳細なコストレポートにアクセスすることもできます。Amazon Q ダッシュボードの詳細については、「」を参照してくださいthe section called “使用状況メトリクスの表示 (ダッシュボード) ”。

特殊なケースの処理

Amazon Q は、ユニットテスト生成プロセス中にさまざまな特殊なケースと制限を処理し、スムーズなユーザーエクスペリエンスと役立つガイダンスを提供します。

- サポートされていないプログラミング言語： サポートされていない言語の場合、Amazon Q はテストファイルを生成する代わりにチャットで提案を提供します。
- サポートされていないテストフレームワーク： サポートされていないテストフレームワークが発生すると、Amazon Q は利用可能な最善の機能を使用してテストを生成しようとします。
- 非アクティブなファイルに対してリクエストされたテスト： Amazon Q は現在アクティブなファイルのテストのみを生成し、再試行する前に目的のファイルを開くようにユーザーに指示します。

- 関数またはメソッドが選択されていない： 選択した行で関数またはメソッドが検出されない場合、Amazon Q はユニットテストを生成する関数またはメソッドの本文内の行を選択するようにユーザーに指示します。
- コード要素が見つからない： 指定されたクラス、関数、メソッド、またはテストケースが見つからない場合、Amazon Q は要素の名前と場所の検証を求めます。
- 検出された非パブリックメソッド (Java): Java プロジェクトの場合、Amazon Q はパブリックメソッドのテストのみを生成し、プライベートメソッドまたは保護されたメソッドが指定されている場合はユーザーに通知します。
- 月単位の使用制限に達した： 月単位のクォータに達すると、Amazon Q はユーザーに通知し、使用制限に関する情報を提供します。

> **Note**
>
> Amazon Q は一度に 1 つのファイルを処理します。複数のファイルのテストを同時にリクエストした場合、エラーは発生しませんが、テストは生成されません。テストを生成するには、一度に 1 つのファイルを使用していることを確認してください。

Amazon Q Developer でコードを確認する

Amazon Q Developer は、コードベースでセキュリティの脆弱性とコード品質の問題を確認し、開発サイクル全体でアプリケーションの体制を改善できます。コードベース全体のレビューを開始したり、ローカルプロジェクトまたはワークスペース内のすべてのファイルを分析したり、記述時にコードを評価する自動レビューを有効にしたりできます。

コードレビュー中、Amazon Q はカスタムコードとコード内のサードパーティーライブラリの両方を評価します。コードレビューを開始する前に、Amazon Q はフィルタリングを適用して、関連するコードのみがレビューされるようにします。フィルタリングプロセスの一環として、Amazon Q はサポートされていない言語、テストコード、オープンソースコードを除外します。

Amazon Q は、コードで潜在的なセキュリティ脆弱性または品質の問題を検出すると、問題の説明と推奨される修正を含むコード問題を生成します。一部の問題では、コード修正を生成して適用し、コードファイルをインプレースで更新できます。

レビューは、生成 AI とルールベースの自動推論の両方を利用しています。Amazon Q ディテクターは、長年のセキュリティのベストプラクティス AWS と Amazon.com セキュリティのベストプラク

ティスに基づいており、ルールベースのセキュリティと品質のレビューを強化します。セキュリティポリシーが更新され、ディテクターが追加されると、レビューは自動的に新しいディテクターを組み込み、コードがup-to-dateポリシーに準拠していることを確認します。

この機能でサポートされている IDEs「」を参照してくださいサポートされている IDE。サポートされている言語の詳細については、「」を参照してください/review を使用したコードレビューの言語サポート。

トピック

- コードの問題のタイプ
- クォータ
- Amazon Q Developer でコードレビューを開始する
- Amazon Q Developer によって生成されたコードの問題を理解する
- Amazon Q Developer でのコードの問題への対応
- コードの問題のフィルタリング
- Amazon Q Developer コードレビューのコード問題の重要度

コードの問題のタイプ

Amazon Q は、次のタイプのコードの問題についてコードを確認します。

- SAST スキャン — ソースコードのセキュリティ上の脆弱性を検出します。Amazon Q は、リソースリーク、SQL インジェクション、クロスサイトスクリプティングなど、さまざまなセキュリティ問題を特定します。
- シークレット検出 — コード内の機密情報や機密情報の漏洩を防止します。Amazon Q は、コードとテキストファイルで、ハードコードされたパスワード、データベース接続文字列、ユーザー名などのシークレットを確認します。シークレットの検出結果には、保護されていないシークレットとその保護方法に関する情報が含まれます。
- IaC の問題 — インフラストラクチャファイルのセキュリティ体制を評価します。Amazon Q は、Infrastructure as Code (IaC) コードファイルを確認して、設定ミス、コンプライアンス、セキュリティの問題を検出できます。
- コード品質の問題 — コードが品質、保守性、効率の基準を満たしていることを確認します。Amazon Q は、パフォーマンス、機械学習ルール、AWS のベストプラクティスなど、さまざまな品質問題に関連するコード問題を生成します。

- コードデプロイリスク — コードのデプロイに関連するリスクを評価します。Amazon Q は、アプリケーションのパフォーマンスやオペレーションの中断など、コードをデプロイまたはリリースするリスクがあるかどうかを判断します。
- ソフトウェアコンポジション分析 (SCA) — サードパーティーのコードを評価します。 Amazon Q は、コードに統合されているサードパーティーのコンポーネント、ライブラリ、フレームワーク、依存関係を調べ、サードパーティーのコードが安全かつ最新であることを確認します。

Amazon Q がコードの確認に使用するディテクターの完全なリストについては、「Amazon Q Detector Library」を参照してください。

クォータ

Amazon Q セキュリティスキャンは、次のクォータを維持します。

- 入力アーティファクトサイズ – サードパーティーライブラリ、ビルド JAR ファイル、一時ファイルなど、IDE プロジェクトワークスペース内のすべてのファイルの最大サイズ。
- ソースコードサイズ – すべてのサードパーティーライブラリとサポートされていないファイルをフィルタリングした後に Amazon Q がスキャンするソースコードの最大サイズ。

次の表は、自動スキャンとフルプロジェクトスキャンで維持されるクォータを示しています。

リソース	自動レビュー	ファイルまたはプロジェクトのレビュー
入力アーティファクトのサイズ	200 KB	500 MB
ソースコードサイズ	200 KB	50 MB

Amazon Q Developer でコードレビューを開始する

Amazon Q は、コードベース全体を確認するか、記述時にコードを自動的にレビューできます。

開始する前に、コードレビューをサポートする IDE に Amazon Q がインストールされていることを確認してください。詳細については、「IDE に Amazon Q Developer 拡張機能またはプラグインをインストールする」を参照してください。

トピック

- コードとして を確認する
- ファイルまたはプロジェクトを確認する

コードとして を確認する

> **Note**
>
> Amazon Q 自動レビューは、Amazon Q Developer Pro サブスクリプションでのみ使用できます。

自動レビューは、Amazon Q ディテクター によるルールベースのレビューです。Amazon Q は、アクティブにコーディングしているファイルを自動的に確認し、コード内で検出されるとすぐにコードの問題を生成します。Amazon Q が自動レビューを実行すると、インプレースコード修正は生成されません。

Amazon Q を使用する場合、自動レビューはデフォルトで有効になっています。自動レビューを一時停止または再開するには、次の手順を使用します。

自動レビューの一時停止と再開

自動レビューを一時停止するには、次の手順を実行します。

1. IDE ウィンドウの下部から Amazon Q を選択します。

   Amazon Q タスクバーが開きます。
2. 自動レビューの一時停止を選択します。自動レビューを再開するには、自動レビューを再開を選択します。

ファイルまたはプロジェクトを確認する

チャットパネルからレビューを開始して、Amazon Q に特定のファイルまたはプロジェクトのレビューを依頼することもできます。ファイルとプロジェクトのレビューには、ルールベースのレビューと生成 AI を活用したレビューの両方が含まれます。ファイルまたはプロジェクトのレビュー中に生成されたコードの問題には、インプレースコード修正が含まれる場合があります。

ファイルまたはプロジェクトのレビューを開始するには、次の手順を実行します。

JetBrains

1. IDE で確認するファイルまたはプロジェクトを開きます。
2. Amazon Q アイコンを選択してチャットパネルを開きます。**/review** と入力してください。
3. 新しいチャットタブが開きます。Amazon Q では、確認するプロジェクトまたはファイルを選択するように求められます。ファイルを確認する場合、レビューするファイルは IDE で開いていてアクティブである必要があります。プロジェクトを確認する場合、プロジェクトは IDE で開いている必要があります。
4. 実行するレビューのタイプを選択します。Amazon Q は、IDE で現在開いているプロジェクトまたはファイルの確認を開始します。
5. レビューが完了すると、コードの問題タブがチャットパネルの上に開き、Amazon Q が検出した問題のリストが表示されます。
6. 脆弱または低品質のコードが検出されたファイルの特定の領域にリダイレクトする問題を選択できます。

   問題の詳細については、コード問題タブのコード問題名の右側にある拡大鏡アイコンを選択します。コード問題の詳細パネルが開き、問題に関する情報が表示されます。
7. コードの問題に対処するには、「」を参照してくださいAmazon Q Developer でのコードの問題への対応。

Visual Studio Code

1. IDE で確認するファイルまたはプロジェクトを開きます。
2. Amazon Q アイコンを選択してチャットパネルを開きます。
3. 自然言語を使用して、実行するコードレビューのタイプを記述します。最近のコード変更のみ、または ファイル全体を確認できます。コードの変更は、 ファイルの git diff コマンドの出力に基づいて決定されます。

   コードプロジェクトまたはファイルを IDE で開いた状態で、次のような内容を入力できます。

   - **Review my code changes**
   - **Run a code review on this file**
4. Amazon Q はファイルまたはプロジェクトの確認を開始します。完了すると、最も優先度の高い問題と観察結果が要約されます。

問題が検出されると、コード問題パネルがチャットパネルの上に開き、Amazon Q が検出した問題のリストが表示されます。

5. コードの問題の詳細については、チャットパネルの上にあるコードの問題パネルに移動します。そこから、以下を実行できます。

   a. 脆弱または低品質のコードが検出されたファイルの特定の領域にリダイレクトされる問題を選択します。

   b. コードの問題の説明を取得するには、コードの問題の名前の横にある拡大鏡アイコンを選択します。Amazon Q は問題の詳細を提供し、コードに挿入することができる修復を提案します。

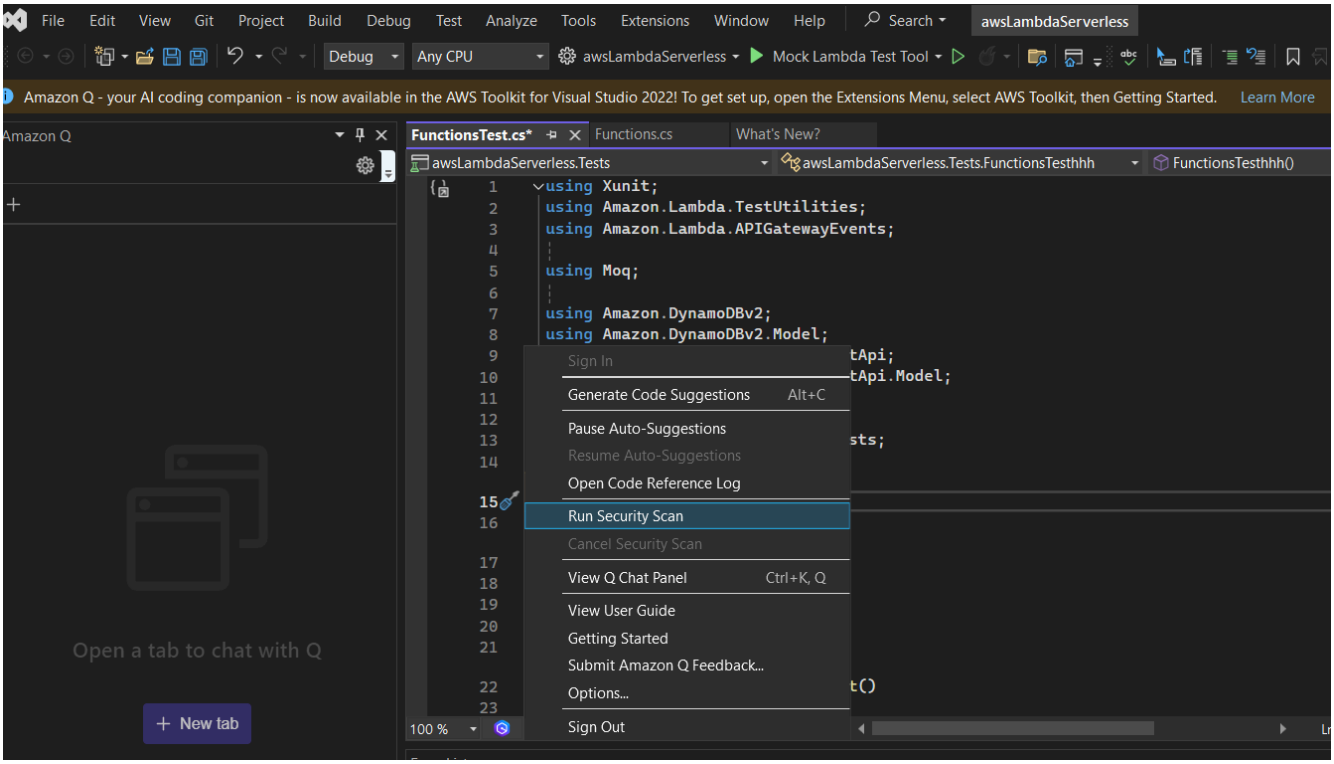
   c. コードの問題を修正するには、コードの問題の名前の横にあるレンチアイコンを選択します。Amazon Q は、修正の簡単な説明を提供し、コードファイルにインプレース修正を行います。ファイルでコードの変更が表示され、チャットパネルから変更を元に戻すオプションがあります。

6. コードの問題に対処する方法の詳細については、「」を参照してくださいAmazon Q Developer でのコードの問題への対応。

Visual Studio

1. Visual Studio でスキャンするプロジェクトのファイルを開きます。
2. ファイル下部の [Amazon Q] アイコンをクリックして、Amazon Q タスクバーを開きます。
3. タスクバーから、[セキュリティスキャンを実行する] を選択します。Amazon Q がプロジェクトのスキャンを開始します。

   次の画像では、ユーザーが Visual Studio で [Amazon Q] アイコンをクリックし、[セキュリティスキャンを実行する] を選択できるタスクバーが表示されています。



4. Visual Studio 出力ペインにあるスキャンのステータスが更新されます。スキャンが完了すると通知があります。

   検出結果の表示と対処に関する詳細については、「Amazon Q Developer でのコードの問題への対応」を参照してください。

Amazon Q Developer によって生成されたコードの問題を理解する

> **Note**
>
> コードの問題に関する以下の情報は IDEs JetBrains で入手できます。Visual Studio のコードの問題については、「」を参照してくださいthe section called “Visual Studio の問題に対処する”。

Amazon Q レビューによって生成されたコードの問題は、コードでセキュリティまたは品質の問題が検出されたことを示します。コードの問題には、次の詳細が含まれます。

- コード問題の名前と重要度 – パネルの上部に、問題の重要度に加えて、問題の名前と CWEs が表示されます。重要度の詳細については、「」を参照してくださいthe section called “コードの問題の重要度”。

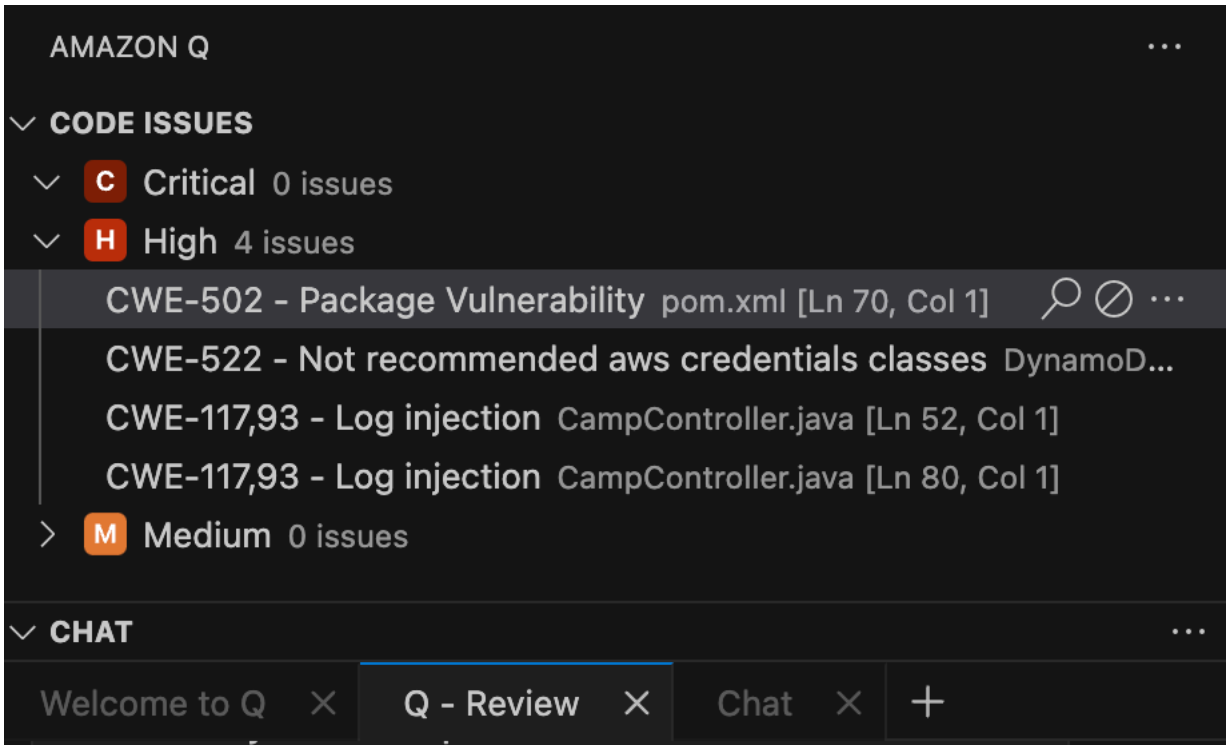
- コード問題の説明 – コード問題を生成したコード行に関する問題を記述します。
- Common Weakness Enumeration (CWE) – コードの問題を特定したディテクターに適用される 1 つ以上の CWE タイプ。詳細については、CWE リンクを選択してください。
- ディレクトリライブラリ – コードの問題を生成した Amazon Q Detector Library 内のディテクターへのリンク。
- ファイルパス – コードの問題を生成したコードを含むファイルの場所。

コードの問題の詳細

Amazon Q が IDEs JetBrains でコードの問題を検出すると、影響を受けるコードが IDE ファイルで強調され、問題がコード問題タブに追加されます。コードの問題の詳細については、次のアクションを実行します。

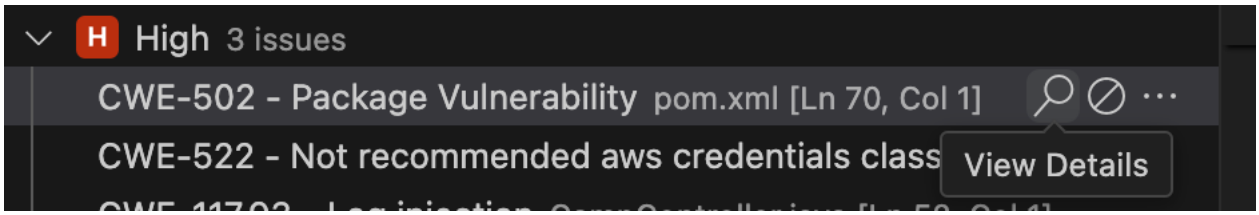
1. コードの問題を生成したコードを表示するには、コードの問題タブから問題を選択します。コードが書き込まれるファイルを開き、問題のあるコードに下線を付けます。

   次の図は、コードの問題タブを示しています。



2. コードの問題の詳細を表示するには、問題を選択し、拡大鏡アイコンを選択してコードの問題の詳細パネルを開きます。下線付きのコードにカーソルを合わせると、同じ情報を含むポップオーバーを表示することもできます。

次の図は、 のコード問題の拡大鏡アイコンを示していますVisual Studio Code。



Amazon Q Developer でのコードの問題への対応

このセクションのトピックでは、コードの問題に対処して解決する方法と、該当する場合は問題を無視する方法について説明します。

トピック

- JetBrains および のコード問題に対処する Visual Studio Code
- Visual Studio のコード問題に対処する

JetBrains および のコード問題に対処する Visual Studio Code

JetBrains および のコード問題に対処するにはVisual Studio Code、インプレース修正を生成するか、コードを手動で更新するために使用できる説明を生成するオプションがあります。

次のアクションを実行できます。

- インプレースコード修正の生成
- 問題の説明と新しいコードの取得
- 問題を無視するか、同様の問題をすべて無視する

ファイルのインプレース修正を生成する

Amazon Q はファイルをインプレースで更新して、検出されたコードの問題を自動的に修正できます。

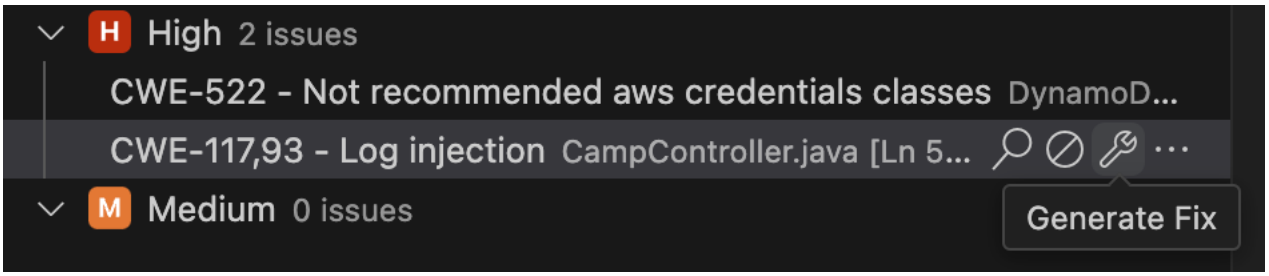
コードの問題を自動的に修正するには：

JetBrains IDEs

1. コードの問題タブで、対処するコードの問題を選択します。

2. レンチガラスアイコンを選択してコード問題の詳細パネルを開き、Amazon Q が修正の生成を開始します。

   次の図は、コードの問題のレンチアイコンを示しています。



3. 更新されたコードは、コードの問題の詳細パネルに表示されます。以下のアクションを実行できます。

   a. 提案されたコードの下で、差分を開く を選択すると、コードの発信元のファイルで提案されたコード変更の差分が表示されます。

   b. 推奨コードの下で、コピーを選択してコードをクリップボードにコピーします。

4. 生成された修正をファイルに適用するには、パネルの下部にある修正を受け入れるを選択して、問題のあるコードを提案されたコードに置き換えます。

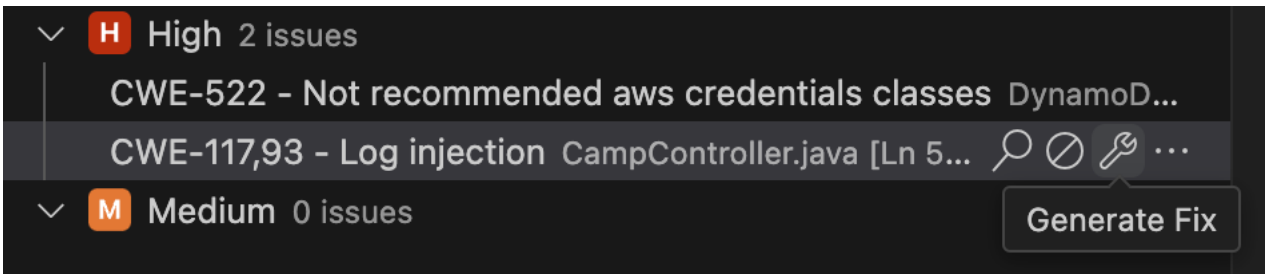
   a. 提案されたコードが要件を満たしていない場合は、修正を再生成を選択して代替ソリューションを表示できます。

   b. 修正を適用すると、コード問題の詳細パネルが閉じ、問題は自動的に解決され、コード問題タブの問題のリストから削除されます。

Visual Studio Code

1. コードの問題 パネルで、対処するコードの問題を選択します。
2. レンチガラスアイコンを選択します。

   次の図は、 のコード問題のレンチアイコンを示していますVisual Studio Code。



3. チャットパネルで、Amazon Q は修正の簡単な説明を提供し、コードファイルにインプレース修正を適用します。
4. 生成された修正をファイルに適用するには、パネルの下部にある修正を受け入れるを選択して、問題のあるコードを提案されたコードに置き換えます。
5. ファイルでコードの変更が表示され、チャットパネルから変更を元に戻すオプションがあります。

コードの問題を説明し、新しいコードを取得する

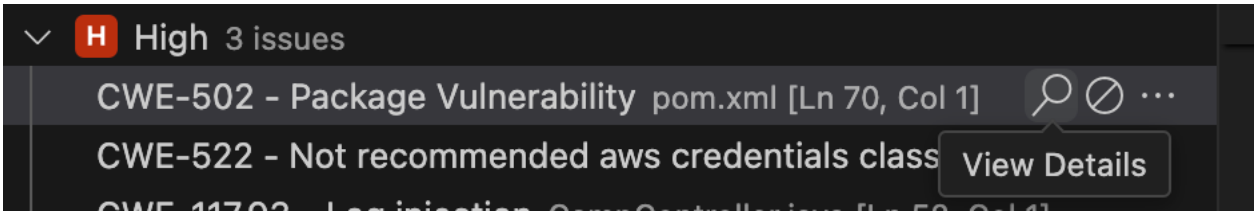
Amazon Q は、コードの問題の詳細な説明を提供し、ファイルに を追加するためのコードを添付した修復オプションを提供します。

コードの問題を説明するには：

JetBrains IDEs

1. コードの問題タブで、対処するコードの問題を選択します。
2. 拡大鏡アイコンを選択して、コード問題の詳細パネルを開きます。

   次の図は、 のコード問題の拡大鏡アイコンを示していますVisual Studio Code。



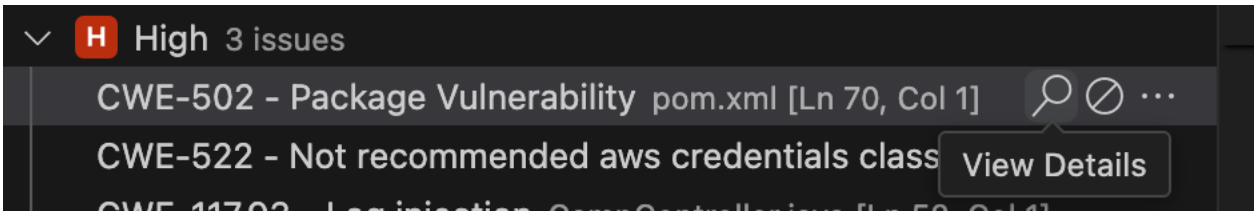
3. コード問題の詳細パネルの下部で、説明を選択します。
4. コードの問題のタイプと関連するコードは、チャットパネルの Amazon Q に送信されます。Amazon Q は問題の説明を提供し、問題のあるコードを置き換えるために使用できる新しいコードを提案します。
5. ファイルにコードを追加するには、カーソルで挿入を選択して、カーソルがファイルにある新しいコードを追加します。自分でコードをコピーしてファイルに貼り付けることもできます。

   下線付きのコードを削除して、新しいコードに置き換えてください。

Visual Studio Code

1. コードの問題 パネルで、対処するコードの問題を選択します。
2. 拡大鏡アイコンを選択します。

   次の図は、 のコード問題の拡大鏡アイコンを示していますVisual Studio Code。



3. チャットパネルで、Amazon Q は問題の詳細を提供し、ファイルに挿入できるコードによる修復を提案します。
4. ファイルを更新するには、Amazon Q の指示に従ってコードを追加または置換し、提供されたコードをファイル内の正しい場所にコピーします。

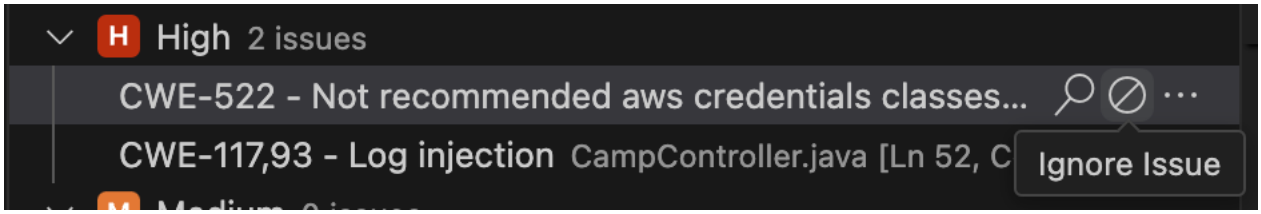
コードの問題を無視する

検出されたコードの問題が適用されない場合は、それを無視するか、それとすべての同様の問題 (同じ CWE の問題) を無視するかを選択できます。問題はコードの問題タブから削除されます。

コードの問題を無視するには：

1. コードの問題 パネルで、無視するコードの問題を選択します。
2. そのコードの問題のみを無視するには、無視アイコンを選択します。

   次の図は、 のコード問題の無視アイコンを示していますVisual Studio Code。



3. コードの問題などを無視するには、楕円アイコンを選択し、類似する問題を無視を選択します。

JetBrains IDEs では、コード問題の詳細パネルから無視を選択することもできます。問題は無視され、コード問題タブの問題のリストから削除されます。コード問題の詳細パネルからすべて無視を選択すると、同じ CWE でのこのコードの問題やその他のコードの問題は無視されます。

Visual Studio のコード問題に対処する

Visual Studio で Amazon Q によって検出されたコードの問題を表示するには、Visual Studio メインメニューの「表示」見出しを展開し、「エラーリスト」を選択して、Visual Studio エラーリストを開きます。

コードの問題の情報を使用してコードを更新できます。コードを更新したら、コードを再度確認して、問題が対処されたかどうかを確認します。

デフォルトでは、Visual Studio の [エラー一覧] には、コードベースのすべての警告とエラーが表示されます。Visual Studio エラーリストから Amazon Q コードの問題をフィルタリングするには、次の手順を実行してフィルターを作成します。

> **Note**
>
> コードの問題は、Amazon Q が問題を検出したコードレビューを実行した後にのみ表示されます。
> コードの問題は、Visual Studio で警告として表示されます。Amazon Q によって検出された問題をエラーリストに表示するには、エラーリスト見出しの警告オプションを選択する必要があります。

エラーリストのコード問題をフィルタリングする

1. Visual Studio のメインメニューで [表示] を選択し、次に [エラー一覧] を選択して [エラー一覧] ペインを開きます。
2. [エラー一覧] ペインで、ヘッダー行を右クリックしてコンテキストメニューを開きます。
3. コンテキストメニューから [列の表示] を展開し、展開したメニューで [ツール] を選択します。
4. [ツール] 列が [エラー一覧] に追加されます。
5. ツール列ヘッダーから、フィルターアイコンを選択し、Amazon Q を選択して Amazon Q コードの問題をフィルタリングします。

コードの問題のフィルタリング

> **Note**
>
> コードの問題は IDEs JetBrains と でのみフィルタリングできますVisual Studio Code。

コード問題をフィルタリングすると、選択した基準を満たす問題のみがコード問題パネルに表示されます。重要度に基づいて問題をフィルタリングできます。

また、コードの問題パネルでコードの問題の整理方法を制御することもできます。問題を重要度またはファイルの場所に基づいてグループ化できます。

コードの問題をフィルタリングするには：

1. Code Issues パネルから、フィルターアイコンを選択します。

   次の図は、 のコード問題タブのフィルターアイコンを示していますVisual Studio Code。

2. フィルターの問題メニューが開きます。

   フィルタリングする重要度タイプの横にあるボックスを選択または選択解除し、OK を選択します。選択した問題のみがコードの問題パネルに表示されます。

コード問題をグループ化するには：

1. コードの問題パネルから、グループ化アイコンを選択します。

   次の図は、 のコード問題タブのグループ化アイコンを示していますVisual Studio Code。

2. グループの問題メニューが開きます。
3. 重要度を選択して、重要度に基づいてコードの問題パネルで問題をグループ化します。Location を選択して、どのコードファイルにあるかに基づいて問題をグループ化します。

Amazon Q Developer コードレビューのコード問題の重要度

Amazon Q は、コードで検出されたコードの問題の重要度を定義し、アプリケーションのセキュリティ体制に対処して追跡する問題を優先できるようにします。以下のセクションでは、コードの問題の重要度を判断するために使用される方法と、各重要度レベルが意味する方法について説明します。

重要度の計算方法

コードの問題の重要度は、問題を生成したディテクターによって決まります。Amazon Q Detector Library のディテクターには、共通脆弱性評価システム (CVSS) を使用して重要度が割り当てられます。CVSS は、検出結果をコンテキストで悪用する方法 (インターネット経由で行うか、物理アクセスが必要かなど) と、取得できるアクセスレベルを考慮します。

次の表は、悪意のある攻撃者がシステムを正常に攻撃するために必要なアクセスレベルと労力のレベルに基づいて重要度がどのように決定されるかを示しています。

	労力のレベル			
	悪用不可	システムへのアクセスが必要	LoE が高いインターネット	インターネット経由
アクセスレベル				
システムまたはその出力を完全に制御する	該当なし	高	[非常事態]	[非常事態]
機密情報へのアクセス	該当なし	Medium	高い	高
システムがクラッシュまたは遅くなる可能性がある	低	低	Medium	Medium
追加のセキュリティを提供します	情報	情報	低	低

	労力のレベル			
	悪用不可	システムへのアクセスが必要	LoE が高いインターネット	インターネット経由
ベストプラクティス	情報	該当なし	該当なし	該当なし

重要度の定義

重要度レベルは次のように定義されます。

重大 – コードの問題は、エスカレーションされないようにすぐに対処する必要があります。

重要なコードの問題は、攻撃者がシステムを制御したり、中程度の労力で動作を変更したりできることを示唆しています。重要な検出結果を最大限の緊急性で扱うことをお勧めします。また、リソースの重大度も考慮する必要があります。

高 – コードの問題は、短期的な優先事項として対処する必要があります。

重要度の高いコードの問題は、攻撃者がシステムを制御したり、高い労力で動作を変更したりできることを示唆しています。重要度の高い検出結果を短期的な優先度として扱い、すぐに修復手順を実行することをお勧めします。また、リソースの重大度も考慮する必要があります。

中 – コードの問題は、中間優先度として対処する必要があります。

重要度が中程度の検出結果は、クラッシュ、応答不能、またはシステムの可用性の低下につながる可能性があります。できるだけ早く、暗黙的なコードを調査することをお勧めします。また、リソースの重大度も考慮する必要があります。

低 – コードの問題では、単独でのアクションは必要ありません。

重要度が低い検出結果は、プログラミングエラーまたはアンチパターンを示唆しています。重要度の低い結果については、すぐにアクションを実行する必要はありませんが、他の問題と相関関係がある場合は、コンテキストを入手できます。

情報 – 推奨アクションはありません。

情報の検出結果には、品質や読みやすさの向上、または代替 API オペレーションの提案が含まれます。即時のアクションは必要ありません。

Amazon Q Developer を使用したドキュメントの生成

> **Note**
>
> チャットでドキュメントを直接生成するように に依頼できます。/doc を使用すると、リクエストはチャットにリダイレクトされます。「」と質問してみてください**generate documentation for my application**。

Amazon Q Developer は、コードの READMEs を生成することで、コードを理解し、ドキュメントを最新の状態に保つのに役立ちます。

Amazon Q は、コードベースで新しいドキュメントを作成し、既存のドキュメントを更新できます。プロジェクト、コードオブジェクト、コードベース内の依存関係を分析することで、Amazon Q は複雑なコーディング概念を文書化し、新しいコード変更に基づいてドキュメントを更新できます。

ドキュメントを生成するには、IDE でプロジェクトまたはワークスペースを開き、チャット**/doc**にと入力します。作業するドキュメント更新のタイプを選択すると、Amazon Q はコードを分析し、ドキュメントを生成し、変更点を差分として提供します。提案された変更を受け入れることも、Amazon Q に追加の変更を加えるフィードバックを提供することもできます。

この機能でサポートされている IDEs「」を参照してくださいサポートされている IDE。サポートされている言語の詳細については、「」を参照してくださいIDE の Amazon Q Developer でサポートされている言語。

トピック

- ユースケース
- [サポートされているファイルの種類]
- クォータ
- Amazon Q Developer を使用した READMEsの生成と更新
- Amazon Q Developer を使用したドキュメント生成のベストプラクティス

ユースケース

Amazon Q は、IDE のチャットパネルから次のタスクを実行できます。

- 新しいドキュメントの作成 – Amazon Q は、選択したフォルダのコードに基づいてプロジェクトの新しい READMEs を作成できます。既に README があり、新しい README を作成することを選択した場合、Amazon Q は既存の README を更新せずに上書きします。Amazon Q が元の README を上書きする前に、新しいコンテンツを確認するオプションがあります。
- 図の作成 – プロジェクトフォルダに infrastructure-as-code ファイル (AWS CDK ファイル AWS CloudFormation 、テンプレート、Terraform 設定ファイルなど) が含まれている場合、Amazon Q はdocsフォルダ内の`infra.svg`ファイルに出力され、README で参照されるインフラストラクチャ図を作成します。を使用している場合はVisual Studio Code、IDE 内の図を表示するために SVG プレビュー (または同様の) 拡張機能をインストールする必要がある場合があります。
- 特定の更新を行う – Amazon Q が README に加える変更を自然言語で記述できます。これを行うには、既存の README を更新し、特定の変更を行うオプションを選択します。Amazon Q がドキュメントを生成したら、追加の変更を記述するオプションがあります。セクションの追加、既存のセクションの削除、既存のセクションの詳細化などの更新をリクエストできます。
- 新しいコードを確認し、ドキュメントの更新を提案する – コードを変更した後、Amazon Q は新しいコードを確認し、関連する更新を README に提案できます。

[サポートされているファイルの種類]

Amazon Q は、ドキュメントの生成時に、以下のファイルタイプを含むがこれらに限定されないソースコードと設定ファイルを確認します。

- .template ファイル
- requirements.txt
- package.json
- tsconfig.json
- Dockerfile
- .git/config
- LICENSE
- LICENSE.md
- 貢献
- CONTRIBUTING.md

Amazon Q は、ファイルで定義された.gitignoreファイルまたはフォルダを除外します。ドキュメント生成のレビュー対象から除外するファイルまたはフォルダがある場合は、プロジェクトまたはワークスペースの.gitignoreファイルに含めることができます。

クォータ

Amazon Q によるドキュメント生成では、次のクォータが維持されます。

- README サイズ – Amazon Q が確認または生成できる README の最大サイズ。既存の README がこのクォータを超えると、Amazon Q は既存のドキュメントを更新できません。生成された README がこのクォータを超えると、Amazon Q は更新された README を返すことができません。
- コードプロジェクトサイズ – Amazon Q がドキュメントの生成に使用できるプロジェクトまたはワークスペースの最大サイズ。

  ドキュメントを生成する小さなフォルダを選択した場合でも、親プロジェクトまたはワークスペースはこのクォータ内にある必要があります。
- タスクあたりのドキュメント生成 – 最初のドキュメント生成など、生成されたドキュメントに変更を加えるためにフィードバックを提供できる回数。このクォータは、新しいドキュメントタスクを開始するたびにリセットされます。

Amazon Q Developer Agent for software development のクォータ

リソース	クォータ
README サイズ	30 KB
コードプロジェクトサイズ	200 MB (非圧縮) 50 MB (圧縮)
タスクあたりのドキュメント生成数	10

Amazon Q Developer を使用した READMEsの生成と更新

> **Note**
>
> チャットでドキュメントを直接生成するように に依頼できます。/doc を使用すると、リクエストはチャットにリダイレクトされます。「」と質問してみてください**generate documentation for my application**。

開始する前に、ドキュメント生成をサポートする IDE に Amazon Q がインストールされていることを確認してください。詳細については、「the section called “Amazon Q のインストール”」を参照してください。

IDE で Amazon Q を使用して READMEs を生成または更新するには、次の手順を実行します。

1. Amazon Q 拡張機能がインストールされている IDE を開きます。
2. Amazon Q アイコンを選択してチャットパネルを開きます。**/doc** と入力してください。
3. 新しいチャットタブが開きます。Amazon Q では、コードの新しい README を作成するか、既存の README を更新するように求められます。
4. 作業するドキュメントタスクのタイプを選択します。

   既存の README を更新する場合は、Amazon Q で最新のコード変更で README を更新するか、Amazon Q が README に対して行う特定の変更を提供するかを選択します。
5. Amazon Q は、IDE で現在開いているフォルダのドキュメントを生成することを確認するように求めるプロンプトを表示します。または、新しいフォルダを選択できます。

   README に特定の変更を加えることを選択した場合は、Amazon Q で実行する更新の詳細を記述します。説明に含める内容に関するアイデアについては、「」を参照してくださいベストプラクティス。

   Amazon Q がドキュメントの生成を開始します。プロジェクトフォルダに infrastructure-as-code ファイルが含まれている場合、Amazon Q は`docs`フォルダ内の`infra.svg`ファイルに出力され、README で参照されるインフラストラクチャ図を作成します。
6. Amazon Q がドキュメントを生成した後、変更があれば差分が提供されます。差分を確認し、チャットで承諾を選択して README を更新できます。

   生成されたドキュメントを変更する場合は、「変更する」を選択します。その後、Amazon Q で変更する内容を記述すると、更新された README が生成されます。

Amazon Q Developer を使用したドキュメント生成のベストプラクティス

Amazon Q を使用したドキュメント生成の結果を向上させるには、次のベストプラクティスに従います。

- Amazon Q は、コードプロジェクトサイズのクォータ内の任意の量のコードのドキュメントを生成できますが、非常に大きなリポジトリでは のドキュメントの生成に時間がかかり、それほど具体的ではない場合があります。大規模なリポジトリがある場合は、より具体的な結果を得るために、コードのサブセットまたは単一のファイルのドキュメントをリクエストすることを検討してください。
- Amazon Q が生成するドキュメントの品質は、コードが適切にコメントおよび整理され、プログラミングエンティティの命名規則が良好であり、他の標準コーディング規則に従っている場合に向上する可能性があります。
- Amazon Q は、サポートされている 1 つ以上の言語で記述されたコードについて、最高品質のドキュメントを作成します。他の言語で記述されたコードは、生成されたドキュメントに反映されないか、品質の低いドキュメントを生成する可能性があります。言語の詳細については、「」を参照してくださいIDE の Amazon Q Developer でサポートされている言語。
- 自然言語で README への特定の変更をリクエストする場合は、既存の README を更新してから、特定の変更を行うオプションを選択します。Amazon Q がドキュメントを生成したら、変更を行い、Amazon Q で実行する更新を記述することもできます。
- Amazon Q でドキュメントに加える更新を記述するときは、次の点を考慮してください。
  - 変更の説明には、変更するセクション、追加または削除するコンテンツ、修正が必要な特定の問題を含める必要があります。
  - 変更は、プロジェクトの機能が README にどのように反映されるかに関連する必要があります。
  - 参照するコンテンツは、コードベースで利用できる必要があります。
- Amazon Q は、プライベートプラットフォームまたは内部プラットフォームにアクセスできません。Amazon Q には、コード内のサードパーティーのツールやソフトウェア、または特殊なツールに関する知識がない場合もあります。コードに Amazon Q がアクセスできないリソースが含まれている場合、または使い慣れていない場合、そのコードは文書化されません。README を手動で編集して、Amazon Q が生成できないコンテンツを含めることができます。

IDE の Amazon Q Developer でサポートされている言語

Amazon Q Developer の以下の機能は、任意のプログラミング言語で IDE で使用できます。

- Chat
- インラインチャット
- 機能開発 - /dev
- ドキュメントの生成 - /doc

これらの機能を使用する際の出力の品質は、言語の人気によって異なります。

IDE の Amazon Q の残りの機能については、サポートされている言語を以下のセクションに示します。

インライン提案の言語サポート

Amazon Q は、複数のプログラミング言語のインラインコード提案をサポートしています。プログラミング言語のコード生成の精度と品質は、トレーニングデータのサイズと品質によって異なります。

トレーニングデータの品質の観点から、最もサポートが充実しているプログラミング言語は次のとおりです。

- C
- C++
- C#
- Dart
- Go
- Java
- JavaScript
- Kotlin
- Lua
- PHP
- PowerShell
- Python
- R
- Ruby
- Rust
- Scala

- シェル
- SQL
- Swift
- SystemVerilog
- TypeScript

サポートが最も充実している Infrastructure as Code (IaC) 言語は次のとおりです。

- CDK (Typescript、Python)
- HCL (Terraform)
- JSON
- YAML

変換の言語サポート

変換でサポートされている言語は、コードを変換する環境によって異なります。

IDEs JetBrains および ではVisual Studio Code、Amazon Q は次の言語でコードを変換できます。

- Java
- Oracle から PostgreSQL データベースへの移行用の埋め込み SQL 変換

Visual Studio では、Amazon Q は次の言語でコードを変換できます。

- .NET アプリケーションの C#

サポートされている言語と変換のその他の前提条件の詳細については、実行中の変換のタイプに関するトピックを参照してください。

/test を使用したユニットテスト生成の言語とフレームワークのサポート

Amazon Q は、次の言語とテストフレームワークのユニットテストを生成できます。

- Python
  - サポートされているテストフレームワーク: Pytest、Unittest

- Java
  - サポートされているテストフレームワーク: JUnit (JUnit 4 および 5、JUnit Jupiter)、Mockito

/review を使用したコードレビューの言語サポート

Amazon Q はコードレビューを作成し、次の言語で記述されたファイルとプロジェクトに対して自動コード修正を提供できます。

- Java - Java 17 以前
- JavaScript - ECMAScript 2021 以前
- Python - Python 3.11 以前、Python 3 シリーズ内
- C# - すべてのバージョン (.NET 6.0 以降を推奨)
- TypeScript - すべてのバージョン
- Ruby - Ruby 2.7 および 3.2
- Go - Go 1.18
- C - C11 以前
- C++ - C++17 以前
- PHP - PHP 8.2 以前
- Kotlin - Kotlin 2.0.0 以前
- Scala - Scala 3.2.2 以前
- JSX - React 17 以前
- Infrastructure as Code (IaC) 言語
  - AWS CloudFormation - 2010-09-09
  - Terraform - 1.6.2 以前
  - AWS CDK - TypeScript と Python

カスタマイズの言語サポート

Amazon Q は、次の言語のカスタマイズをサポートし、リストされているファイルタイプを使用してカスタマイズを作成します。

- Bash/Shell (.sh、.zsh、.bash)
- C (.c、.h)

- C# (.cs)
- C++ (.cpp、.hpp、.h)
- ダート (.dart)
- Go (.go)
- HCL (.hcl)
- HTML (.html、.htm)
- Java (.java)
- JavaScript (.js、.jsx)
- JSON (.json)
- Kotlin (.kt、.kts)
- マークダウン (.md、.mdx)
- PHP (.php)
- Powershell (.ps1、.psm1、.psd1)
- Python (.py)
- reStructuredText (.rst)
- Ruby (.rb)
- Rust (.rs)
- Scala (.scala)
- Terraform (.tf、.tfvars)
- テキスト (.txt)
- TypeScript (.ts、.tsx)
- YAML (.yaml、.yml)

コマンドラインで Amazon Q Developer を使用する

Amazon Q Developer を使用して、、、git、npm、 などの何百もの一般的な CLIs の完了を有効にできますdockeraws。Amazon Q for コマンドラインはコンテキスト情報を統合して、Amazon Q にユースケースの理解を深め、コンテキストに対応した関連性の高いレスポンスを提供できるようにします。入力を開始すると、Amazon Q はコンテキストに関連するサブコマンド、オプション、引数を入力します。

Amazon Q for コマンドラインは、自動補完、Amazon Q チャット、インライン zsh 補完などの機能を提供する AppImage や Ubuntu パッケージなど、macOS および特定の Linux 環境にインストールできます。Amazon Q CLI を使用して、以下を実行できます。

- ターミナル環境内で自然言語の会話、質問、応答をチャットするためのコマンドラインインターフェイス。
- 自然言語の手順を実行可能シェルコードスニペットに変換します。
- コマンドプロンプトに入力するときのインライン提案と、一般的な CLIs の完了。

詳細については、サポートされているコマンドライン環境 およびコマンドライン用の Amazon Q をインストールする を参照してください。

トピック

- サポートされているコマンドライン環境
- コマンドライン用の Amazon Q をインストールする
- コマンドラインでのチャットの使用
- カスタムエージェント
- CLI でエディタコマンドを使用する
- Git 対応ファイル選択の使用
- コンテキスト管理
- コンテキストフックの使用
- Amazon Q の設定
- Amazon Q CLI コマンドリファレンス
- コマンドライン支援機能
- コマンドライン用の Amazon Q への寄稿

サポートされているコマンドライン環境

コマンドライン用の Amazon Q のオートコンプリート機能は、macOS および AppImage や Ubuntu などの特定の Linux 環境でサポートされています。

macOS と Linux の両方で次の環境がサポートされています。

- シェル: bash、zsh、fish
- CLIs: git、aws、docker、npm、 yarn など、500 を超える最も一般的な CLIs

macOS

Amazon Q for コマンドラインは、macOS の次の環境と統合されます。

- ターミナルエミュレータ: iTerm2、macOS ターミナル、Hyper、Alacritty、Kitty、WezTerm。ターミナルの完全なリストを確認するには、Amazon Q for コマンドラインオープンソースコードを参照してください。
- IDE: VS Code ターミナル、JetBrains ターミナル (フリートを除く)
- macOS 10.15 (Catalina) 以降
- アーキテクチャ: x86_64 (Intel) または arm64 (Apple Silicon)

リナックス

Amazon Q for コマンドラインは、Linux の次の環境と統合されます。

- プラットフォームの要件: Linux のコマンドライン用 Amazon Q は、Ubuntu 22 と 24、および Ubuntu 20 の機能のサブセットの互換性をサポートしています。それ以外の場合は、GNOME v42+ または表示サーバーが Xorg で、入力メソッドフレームワークが IBus である環境で動作します。
- ターミナルエミュレータ: GnomeConsole、GnomeTerminal、Kitty、Hyper、WezTerm、Alacritty、Tilix、Terminator
- アーキテクチャ: x86_64 または aarch64
- 注: デスクトップ機能は、現在 x86_64 アーキテクチャでのみ使用できます。

自然言語

Amazon Q Developer は、multi-natural-languageサポートを提供します。サポートされている自然言語には、北京語、フランス語、ドイツ語、イタリア語、日本語、スペイン語、韓国語、ヒンディー語、ポルトガル語があり、さらに多くの言語を使用できます。

この機能を利用するには、任意の自然言語を使用して Amazon Q Developer との会話を開始できます。Amazon Q は言語を自動的に検出し、適切な言語でレスポンスを提供します。

コマンドライン用の Amazon Q をインストールする

Amazon Q アプリケーションのファイルダウンロードを開始することで、macOS のコマンドラインに Amazon Q をインストールできます。詳細については、「サポートされているコマンドライン環境」を参照してください。

コマンドラインに Amazon Q をインストールするときに考慮すべき 2 つのモダリティがあります。最小インストールでは、Linux for Amazon Q チャットおよびオートコンプリート機能が SSH (q および) で機能するために必要なバイナリのみが分散されますqterm。フルディストリビューションにはデスクトップアプリケーションが含まれており、オートコンプリート機能を使用する必要があります。Amazon Q チャットのみを使用する場合は、最小ディストリビューションもシェルqtermに出荷およびインストールされることを考慮してください。qterm for zsh を使用してインライン補完をサポートでき、フルバージョンは必要ありません。詳細については、「zip ファイルを使用した のインストール」を参照してください。

> **Note**
>
> Windows で Amazon Q CLI を使用する方法については、dev.to のブログ記事を参照してください。

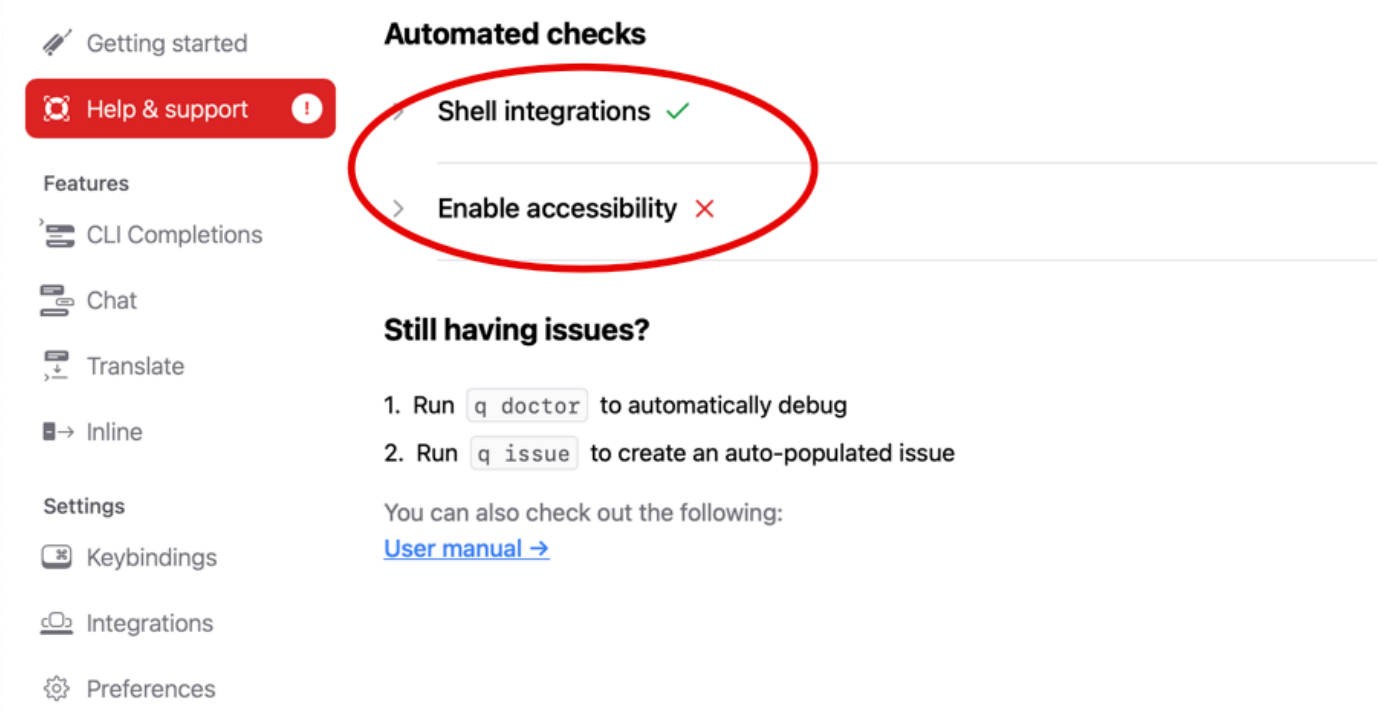
macOS

Amazon Q for macOS は、アプリケーションをダウンロードするか、Homebrew を使用してインストールできます。

macOS のコマンドラインに Amazon Q をインストールした後、SSH 統合を有効にして、500 を超えるコマンドラインツールでオートコンプリートを使用できます。詳細については、「ローカル macOS 統合」を参照してください。

macOS のコマンドラインに Amazon Q をインストールするには

1. macOS のコマンドライン用に Amazon Q をダウンロードします。
2. (オプション) macOS のコマンドライン用に Amazon Q 用にダウンロードしたファイルを確認します。詳細については、「ダウンロードを検証するには (オプション) 」を参照してください。
3. ダウンロードした .dmg ファイルをダブルクリックし、アプリケーションをアプリケーションフォルダにドラッグします。
4. アプリケーションフォルダで、Amazon Q をダブルクリックします。GUI が開きます。
5. シェル統合を有効にします。これにより、シェルから Amazon Q を実行でき、Amazon Q はシェルコマンドの自動補完に役立ちます。
6. Builder ID または IAM アイデンティティセンターで、アカウント管理者から提供された開始 URL を使用して認証を行います。
7. 手順に従ってシェル統合をインストールし、macOS アクセシビリティアクセス許可を付与します。



Linux AppImage

> **Warning**
>
> このインストール方法には GUI が必要です。GUI を使用せずに Linux にインストールする場合は、zip ファイルを使用した のインストール「」を参照してください。

Amazon Q for Linux のコマンドラインは、AppImage 形式を使用してインストールできます。AppImage 形式は、インストールを必要とせずにほとんどの Linux ディストリビューションで動作するポータブル形式です。

AppImage を使用して Linux のコマンドラインに Amazon Q をインストールするには

1. Linux AppImage のコマンドライン用の Amazon Q をダウンロードします。
2. AppImage を実行可能にします。

```
chmod +x amazon-q.appimage
```

3. AppImage を実行します。

```
./amazon-q.appimage
```

4. Builder ID または IAM アイデンティティセンターで、アカウント管理者から提供された開始 URL を使用して認証を行います。

Ubuntu

> **Warning**
>
> このインストール方法には GUI が必要です。GUI を使用せずに Linux にインストールする場合は、zip ファイルを使用した のインストール「」を参照してください。

.deb パッケージを使用して、Ubuntu のコマンドラインに Amazon Q をインストールできます。

Ubuntu のコマンドラインに Amazon Q をインストールするには

1. Ubuntu のコマンドライン用に Amazon Q をダウンロードします。

```
wget https://desktop-release.q.us-east-1.amazonaws.com/latest/amazon-q.deb
```

2. パッケージをインストールします。

```
sudo apt-get install -f
sudo dpkg -i amazon-q.deb
```

3. コマンドライン用に Amazon Q を起動します。

```
q
```

4. Builder ID または IAM アイデンティティセンターで、アカウント管理者から提供された開始 URL を使用して認証を行います。

Homebrew

Homebrew で Amazon Q Developer CLI をインストールするには

```
brew install --cask amazon-q
```

zip ファイルを使用した のインストール

Linux と Amazon Q Developer CLI の統合を設定するには、システムに適したファイルをインストールし、ファイルを検証して、プログラムをインストールする必要があります。

インストールと更新の要件

- ダウンロードしたパッケージを抽出または解凍できる必要があります。オペレーティングシステムに組み込みの unzip コマンドがない場合は、同等のコマンドを使用します。
- Amazon Q for コマンドラインには glibc 2.34 以降が必要です。これは、2021 年以降にリリースされたほとんどの主要な Linux ディストリビューションにデフォルトで含まれています。
- 2.34 より前のバージョンの glibc を使用する古いディストリビューションでは、特別な musl ベースのバージョン (ファイル名`-musl.zip`に で指定) を使用します。
- Amazon Q for コマンドラインは、Fedora、Ubuntu、Amazon Linux 2023 の最近のディストリビューションの 64 ビット x86_64 および ARM aarch64 バージョンでサポートされています。

glibc バージョンの確認

必要なバージョンを確認するには、glibc バージョンを確認してください。

```
ldd --version
```

バージョンが 2.34 以降の場合は、標準バージョンを使用します。古い場合は、musl バージョンを使用します。

インストールファイルをダウンロードするには

システムアーキテクチャと glibc バージョンに基づいて、適切なダウンロードコマンドを選択します。

標準バージョン (glibc 2.34 以降)

Linux x86-64

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-x86_64-linux.zip" -o "q.zip"
```

Linux ARM (aarch64)

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-aarch64-linux.zip" -o "q.zip"
```

Musl バージョン (glibc < 2.34 の場合)

Linux x86-64 と musl

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-x86_64-linux-musl.zip" -o "q.zip"
```

musl を使用した Linux ARM (aarch64)

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-aarch64-linux-musl.zip" -o "q.zip"
```

ダウンロードを確認するには (オプション)

Amazon Q コマンドラインインストーラパッケージ .zip を手動でダウンロードすることを選択した場合は、GnuPG ツールを使用して署名を検証できます。

1. ご利用のパッケージマネージャーを使用し、gpg コマンドをダウンロードしてインストールします。詳細については、GnuPG ドキュメントを参照してください。
2. テキストファイルを作成してパブリックキーファイルを作成し、次のテキストに貼り付けます。

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mDMEZig60RYJKwYBBAHaRw8BAQdAy/+G05U5/EOA72WlcD4WkYn5SInri8pc4Z6D
BKNNGOm0JEFtYXpvbiBRIENMSSBUZWFtIDxxLWNsaUBhbWF6b24uY29tPoiZBBMW
CgBBFiEEmvYEF+gnQskUPgPsUNx6jcJMVmcFAmYoOtECGwMFCQPCZwAFCwkIBwIC
IgIGFQoJCAsCBBYCAwECHgcCF4AACgkQUNx6jcJMVmef5QD/QWWEGG/cOnbDnp68
SJXuFkwiNwlH2rPw9ZRIQMnfAS0A/0V6ZsGB4kOylBfc7CNfzRFGtovdBBgHqA6P
zQ/PNscGuDgEZig60RIKKwYBBAGXVQEFAQEHQC4qleONMBCq3+wJwbZSr0vbuRba
D1xr4wUPn4Avn4AnAwEIB4h+BBgWCgAmFiEEmvYEF+gnQskUPgPsUNx6jcJMVmcF
AmYoOtECGwwFCQPCZwAACgkQUNx6jcJMVmchMgEA6l3RveCM0YHAGQaSFMkguoAo
vK6FgOkDawgP0NPIP2oA/jIAO4gsAntuQgMOsPunEdDeji2t+AhV02+DQIsXZpoB
=f8yY
-----END PGP PUBLIC KEY BLOCK-----
```

3. Amazon Q コマンドラインパブリックキーを次のコマンドでインポートし、*public-key-file-name* を作成したパブリックキーのファイル名に置き換えます。

```
gpg --import public-key-file-name
```

```
gpg: directory '/home/username/.gnupg' created
gpg: keybox '/home/username/.gnupg/pubring.kbx' created
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key 50DC7A8DC24C5667: public key "Amazon Q command line Team <q-command
 line@amazon.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
```

4. ダウンロードしたパッケージの Amazon Q コマンドライン署名ファイルをダウンロードします。これは、対応する .zip ファイルと同じパスと名前を持っていますが、拡張子は .sig です。

   標準バージョン (glibc 2.34 以降):

Linux x86-64:

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-x86_64-linux.zip.sig" -o "q.zip.sig"
```

Linux ARM (aarch64):

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-aarch64-linux.zip.sig" -o "q.zip.sig"
```

Musl バージョン (glibc < 2.34 の場合):

Linux x86-64 と musl:

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-x86_64-linux-musl.zip.sig" -o "q.zip.sig"
```

Linux ARM (aarch64) と musl:

```
curl --proto '=https' --tlsv1.2 -sSf "https://desktop-release.q.us-
east-1.amazonaws.com/latest/q-aarch64-linux-musl.zip.sig" -o "q.zip.sig"
```

5. ダウンロードした .sig ファイル名と .zip ファイル名の両方をパラメータとして gpg コマンドに渡して、署名を検証します。

```
gpg --verify q.zip.sig q.zip
```

出力は次の例に類似したものになります:

```
gpg: Signature made Wed 24 Apr 2024 12:08:49 AM UTC
gpg:                using EDDSA key 9AF60417E82742C9143E03EC50DC7A8DC24C566
gpg: Good signature from "Amazon Q command line Team <q-command
 line@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 9AF6 0417 E827 42C9 143E  03EC 50DC 7A8D C24C 5667
```

> **Note**
>
> 出力内の警告は予想されており、問題を示すものではありません。これは、個人用の PGP キー (持っている場合) とコマンドライン用 Amazon Q の PGP キーの間に信頼チェーンがないために表示されます。詳細については、「信用の輪 (Web of Trust)」を参照してください。

Amazon Q CLI をインストールするには

1. インストーラを解凍します。

```
unzip q.zip
```

2. インストールプログラムを実行します。

```
./q/install.sh
```

デフォルトでは、ファイルは にインストールされます`~/.local/bin`。

リモート使用のための SSH の設定

Amazon Q CLI をインストールしたら、リモートマシンで Amazon Q のコマンドライン機能を有効にするようにリモート Linux 統合を設定できます。

ローカル macOS 統合

ローカル macOS マシンから SSH 統合を有効にするには：

1. ターミナルまたはコマンドプロントを開きます。
2. ローカル SSH 統合を有効にします。

```
q integrations install ssh
```

リモート Linux 統合

リモート Linux マシンで SSH 統合を設定するには：

1. SSH サーバー設定を編集します。

```
sudo -e /etc/ssh/sshd_config
```

2. 設定ファイルの末尾に次の行を追加します。

```
AcceptEnv Q_SET_PARENT
AllowStreamLocalForwarding yes
```

3. SSH サービスを再起動します。

```
sudo systemctl restart sshd
```

4. SSH セッションから切断して再接続します。
5. 再接続したら、Amazon Q にログインします。

```
q login
```

6. インストールを確認します。

```
q doctor
```

既知の制限事項

SSH を使用してリモートマシンに接続している間に Amazon Q デスクトップクライアントが終了すると、エラーメッセージが繰り返し出力される場合があります。

```
connect to /var/folders/tg/u1vx4xfmvqav0oxfa4zfknaxiwmbsbr/T/cwrun/remote.sock port -2
 failed: Connection refused
```

エラーを停止するには、SSH セッションを終了し、Amazon Q デスクトップクライアントを再接続するか、再起動します。

トラブルシューティング

SSH 統合で問題が発生した場合：

1. を実行して一般的な問題を特定して修正q doctorする
2. ローカル設定とリモート設定の両方が正しいことを確認します。

3. 必要な環境変数を受け入れるように SSH サーバーが正しく設定されていることを確認します。
4. システムに正しいバージョン (標準または musl) を使用していることを確認します。

コマンドライン用の Amazon Q をアンインストールする

不要になった場合は、Amazon Q for コマンドラインをアンインストールできます。

macOS で Amazon Q for コマンドラインをアンインストールするには

1. Finder で Applications フォルダを開きます。
2. Amazon Q アプリケーションを見つけます。
3. アプリケーションをゴミ箱にドラッグするか、右クリックして「ゴミ箱に移動」を選択します。
4. ゴミ箱を空にしてアンインストールを完了します。

Ubuntu で Amazon Q for コマンドラインをアンインストールするには

1. apt パッケージマネージャーを使用してパッケージを削除します。

```
sudo apt-get remove amazon-q
```

2. 残りの設定ファイルをすべて削除します。

```
sudo apt-get purge amazon-q
```

コマンドライン用の Amazon Q Developer のデバッグ

コマンドラインで Amazon Q Developer に問題がある場合は、 `q doctor`を実行して一般的な問題を特定して修正します。

正常な出力

```
$ q doctor

# Everything looks good!

Amazon Q still not working? Run q issue to let us know!
```

出力が想定どおりの出力ではない場合は、プロンプトに従って問題を解決します。それでも機能しない場合は、 を使用してバグ`q issue`を報告します。

一般的な問題

コマンドラインに Amazon Q を使用する場合に発生する可能性がある一般的な問題は次のとおりです。

認証の失敗

認証に問題がある場合は、 `q login` を実行して再認証してみてください。

オートコンプリートが機能しない

を実行して、シェル統合が正しくインストールされていることを確認します`q doctor`。

SSH 統合の問題

必要な環境変数を受け入れるように SSH サーバーが正しく設定されていることを確認します。

トラブルシューティングのステップ

コマンドラインで Amazon Q に関する問題をトラブルシューティングするには、次の手順に従います。

1. を実行して`q doctor`、一般的な問題を特定して修正します。
2. インターネット接続を確認してください。
3. サポートされている環境を使用していることを確認します。詳細については、「サポートされているコマンドライン環境」を参照してください。
4. コマンドラインに Amazon Q を再インストールしてみてください。
5. 問題が解決しない場合は、 を使用して報告します`q issue`。

コマンドラインでのチャットの使用

Amazon Q Developer CLI は、ターミナルでインタラクティブなチャットエクスペリエンスを直接提供します。コマンドライン環境を離れることなく、質問したり、AWS のサービスに関するヘルプを得たり、問題のトラブルシューティングを行ったり、コードスニペットを生成したりできます。

チャットセッションの開始

Amazon Q とのチャットセッションを開始するには、 chatサブコマンドを使用します。

```
$ q chat
```

これにより、質問やコマンドを入力できるインタラクティブなチャットセッションが開きます。

チャットセッションを終了するには、 と入力/quitするか、 を押しますCtrl+D。

チャットコマンド

Amazon Q は、チャットセッション中に使用できるいくつかのコマンドをサポートしています。これらのコマンドはスラッシュ () で始まります/。

チャットコマンド

コマンド	説明
/load	JSON ファイルから会話状態をインポートする
/save	会話状態を JSON ファイルにエクスポートする
/prompts	使用可能なすべてのプロンプトを一覧表示します
/usage	コンテキストウィンドウの使用状況の見積もりを表示します。
!	Amazon Q CLI セッション内からシェルコマンドを実行します。
ctrl-j	マルチライン入力を許可する
ctrl-k	あいまい検索
/editor	設定されたエディタを使用してプロンプトを作成します。
/help	使用可能なコマンドのリストを表示します。
/issue	問題を報告したり、機能リクエストを行ったりする
/model	使用可能なモデルを表示し、現在のチャットセッションでモデルを選択できます。

コマンド	説明
/quit	チャットセッションを終了します
/clear	現在のセッションからチャット履歴をクリアします
/reset	会話コンテキストをリセットし、以前のメッセージをすべてクリアします
/tools	Amazon Q が使用できるツールのツールとアクセス許可を管理します
/acceptall	廃止済み。Amazon Q がシステムでアクションを実行するときに確認プロンプトを無効にします
/profile	Q Developer コマンドの Q プロファイルを管理します
/context	Amazon Q で利用可能なコンテキスト情報を管理します。
/compact	会話履歴を圧縮し、圧縮された会話履歴の出力を表示します。
/agent list	環境で使用可能なすべてのエージェントを表示します。
/agent schema	エージェント設定ファイルを作成するための JSON スキーマを表示します
/agent create [name]	指定された名前で新しいエージェントを作成します。
/agent use [name]	現在のセッションで特定のエージェントを使用するように切り替えます
/agent edit [name]	エージェント設定ファイルを開いて編集します

ツールのアクセス許可の管理

/tools コマンドを使用して、Amazon Q がシステムでアクションを実行するために使用するツールのアクセス許可を管理できます。これにより、Amazon Q が実行できるアクションをきめ細かく制御できます。

ツールコマンド

コマンド	説明
help	ツールに関連するヘルプを表示します。
trust	セッションの特定のツールを信頼します。
untrust	ツールをリクエストごとの確認に戻します。
trust-all	すべてのツール (廃止された /acceptall に相当) を信頼します。
reset	すべてのツールをデフォルトのアクセス許可レベルにリセットします。

すべてのツールの現在のアクセス許可設定を表示するには：

```
$ q chat
Amazon Q> /tools
```

これにより、使用可能なすべてのツールとその現在のアクセス許可ステータス (信頼済みまたはリクエストごと) のリストが表示されます。

ツールのアクセス許可には、次の 2 つの状態があります。

- 信頼できる: Amazon Q は、毎回確認を求めることなくツールを使用できます。
- リクエストごと: Amazon Q は、ツールを使用する前に毎回確認を求める必要があります。

現在のセッションの特定のツールを信頼または信頼解除するには：

```
Amazon Q> /tools trust fs_read
Amazon Q> /tools untrust execute_bash
```

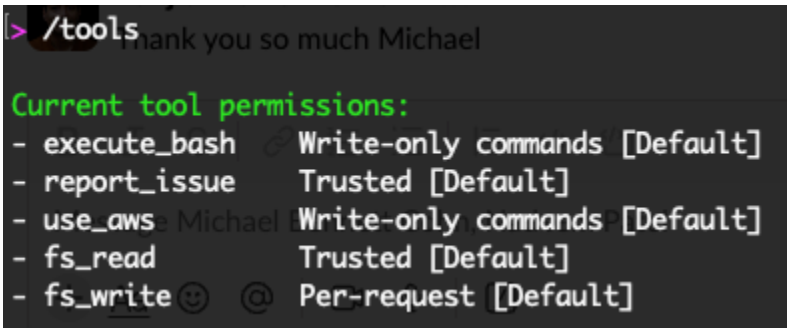
```
> /tools untrust fs_read

Tool 'fs_read' set to per-request confirmation.

> /tools trust fs_write

Tool 'fs_write' is now trusted. I will not ask for confirmation before running this tool.

> /tools

Current tool permissions:
- execute_bash     Read-only commands [Default]
- fs_read          Per-request
- report_issue     Trusted [Default]
- use_aws          Read-only commands [Default]
- fs_write         Trusted
```

ですべてのツールを一度に信頼することもできます /tools trust-all (非推奨の /acceptall コマンドに相当) 。

```
Amazon Q> /tools trust-all
```

> **Warning**
>
> の使用にはリスク/tools trust-allが伴います。詳細については、「セキュリティリスクについて」を参照してください。

次の図は、デフォルトの信頼ステータスにある CLI ツールのステータスを示しています。



Amazon Q では、次のツールをネイティブに使用できます。

使用可能なツール

ツール	説明
fs_read	システム上のファイルとディレクトリを読み取ります。
fs_write	システム上のファイルを作成および変更します。
execute_bash	システムで bash コマンドを実行します。
use_aws	AWS サービスとやり取りするための CLI AWS 呼び出しを行います。
report_issue	ブラウザを開いて、チャットに関する問題を報告します AWS。

Amazon Q が明示的なアクセス許可を持たないツールを使用しようとすると、先に進む前に承認を求められます。アクションを許可または拒否するか、セッションの残りの時間ツールを信頼するかを選択できます。

各ツールにはデフォルトの信頼動作があります。 fs_read はデフォルトで信頼される唯一のツールです。

異なるアクセス許可レベルを使用するタイミングの例をいくつか示します。

- Trust fs_read: コードベースを調べるときなど、Amazon Q が確認なしでファイルを読み取る場合。
- Trust fs_write: プロジェクトでアクティブに作業していて、Amazon Q でファイルを作成または変更できるようにする場合。
- Untrust execute_bash: 実行前にすべてのコマンドを確認する機密性の高い環境で作業する場合。
- Untrust use_aws: 意図しない変更を防ぐために本番稼働用の AWS リソースを操作する場合。

Amazon Q がツールを使用すると、使用されている信頼アクセス許可が表示されます。

q chat セッションの開始の一部として信頼アクセス許可を指定することもできます。

```
Options:
  -a, --accept-all              (Deprecated, use --trust-all-tools) Enabling this flag allows the model to execute
                                all commands without first accepting them
      --no-interactive          Print the first response to STDOUT without interactive mode. This will fail if the
                                prompt requests permissions to use a tool, unless --accept-all is also used
      --profile <PROFILE>       Context profile to use
      --trust-all-tools         Allows the model to use any tool to run commands without asking for confirmation
      --trust-tools <TOOL_NAMES> Trust only this set of tools. Example: trust some tools:
                                '--trust-tools=fs_read,fs_write', trust no tools: '--trust-tools='
  -v, --verbose...              Increase logging verbosity
  -h, --help                    Print help
```

コマンドラインでのチャットのモデルの選択

Amazon Q がチャットセッション中にリクエストに応答するために使用するモデルを選択できます。チャットセッションを開始するとデフォルトモデルが設定され、Amazon Q が特定のセッションに使用するモデルを変更するか、すべてのセッションにデフォルトモデルを設定できます。

チャットセッションでは、次の Anthropic モデルを使用できます。

- Claude 3.7 Sonnet (デフォルト)
- Claude Sonnet 4

以下のセクションでは、Amazon Q がチャットセッションに使用するモデルを変更する方法について説明します。

チャットセッションのモデルを変更する

Amazon Q が既に開始しているチャットセッションに使用するモデルを変更するには、 `/model` コマンドを使用します。

現在使用されているモデルは、サポートされているモデルのリストに表示されます。モデルを変更するには、矢印キーを使用してリスト内を移動し、キーボードで Enter を選択して選択します。

選択したモデルは、チャットセッションの間、またはユーザーが変更するまで保持されます。新しいセッションを開始すると、Amazon Q はデフォルトのモデルを再度使用します。

特定のモデルでチャットセッションを起動する

1 つのコマンドを使用して、特定のモデルでチャットセッションを開始できます。サポートされているモデルの名前で `--model`オプションを追加して、そのモデルをセッションに使用します。

```
q chat --model <model name>
```

モデル名は、次の形式で指定する必要があります。

- claude-3.5-sonnet
- claude-3.7-sonnet
- claude-4-sonnet

選択したモデルは、チャットセッションの間、またはユーザーが変更するまで保持されます。新しいセッションを開始すると、Amazon Q はデフォルトのモデルを再度使用します。

チャットセッションのデフォルトモデルを設定する

次のコマンドを使用して、チャットセッションを開始するときに Amazon Q が使用するデフォルトモデルを変更できます。

```
q settings chat.defaultModel <model name>
```

モデル名は、次の形式で指定する必要があります。

- claude-3.5-sonnet
- claude-3.7-sonnet
- claude-4-sonnet

Amazon Q は、今後すべてのチャットセッションのデフォルトモデルとして設定したモデルを使用します。前述の方法を使用して、特定のチャットセッションに別のモデルを設定することもできます。

Conversations

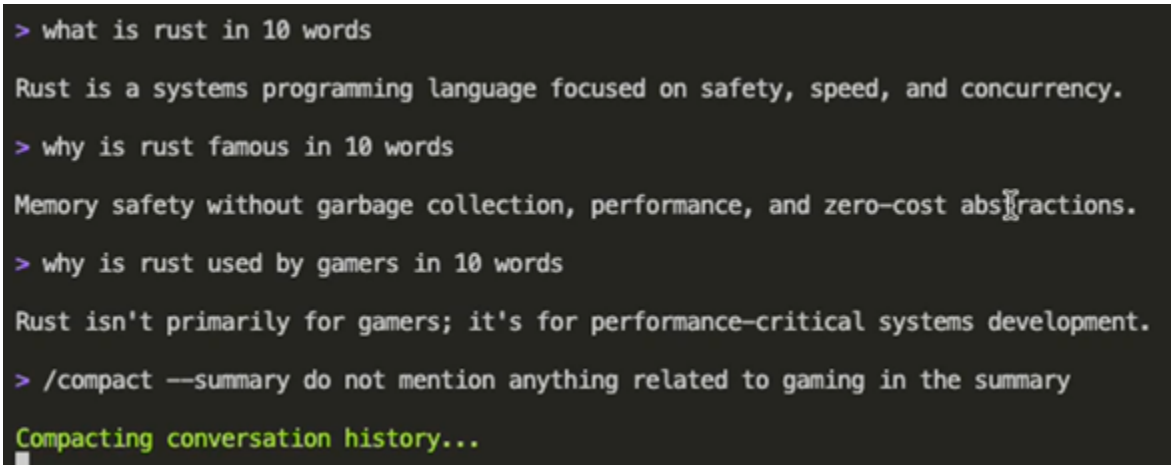
トピック

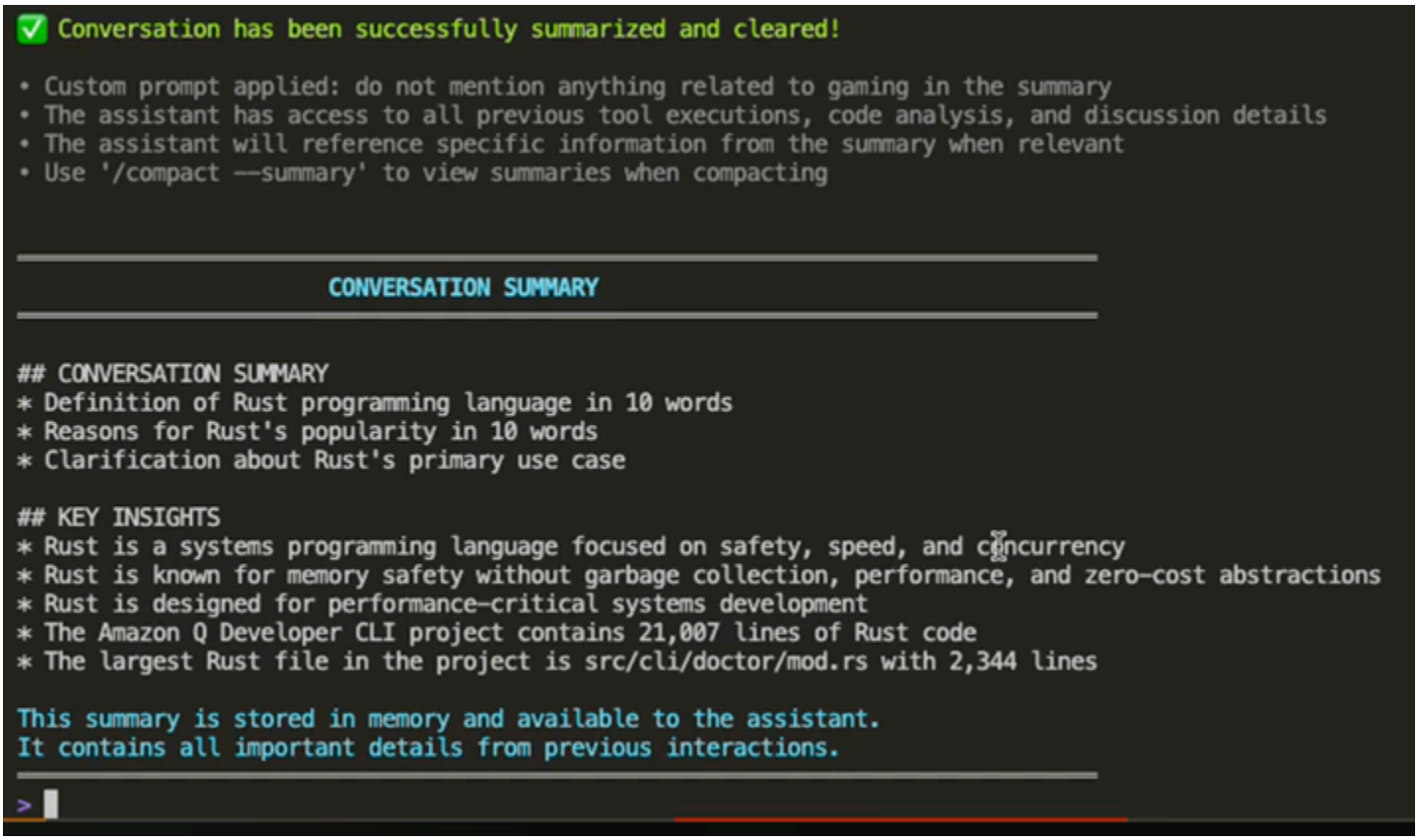
- 会話の要約
- 会話の永続性

会話の要約

`/compact` コマンドは会話履歴を圧縮し、圧縮された会話履歴の出力を表示します。

会話履歴の文字の長さが制限に近づくと、Amazon Q は会話履歴が必要であることを示す警告メッセージを表示します`/compact`。





会話の永続性

Amazon Q は、会話が行われるディレクトリに基づいて会話を記憶できます。以前に Amazon Q とチャットしたディレクトリに戻ると、その会話履歴を自動的にロードするように Q に指示できるため、シームレスに議論を続けることができます。

ディレクトリベースの永続性

そのディレクトリで初めてチャットする場合、Amazon Q は新しい会話を開始します (指定されたコンテキストを考慮します)。

現在のディレクトリで会話を明示的に再開するには、以下を使用します。

```
$ q chat --resume
```

会話の手動保存とロード

チャットセッション中に、次のコマンドを使用して会話を手動で保存およびロードすることもできます。

- `/save [path]` - 現在の会話を JSON ファイルに保存します。
  - `-f` または `--force`を追加して既存のファイルを上書きする
  - 例: `/save ~/my-project-conversation -f`
- `/load [path]` - 以前に保存した JSON ファイルから会話をロードします
  - 例: `/load ~/my-project-conversation.json`

> **Note**
>
> コマンド/saveと /load コマンドは、会話が最初に作成されたディレクトリとは独立して動作します。会話をロードするときは、保存元のディレクトリに関係なく、現在の会話が置き換えられることに注意してください。

これらのコマンドは、次の場合に便利です。

- 重要な会話のバックアップ
- チームメンバーとの会話の共有
- 異なる作業ディレクトリ間の会話の移動

イメージの操作

Amazon Q は、チャットセッションでイメージを直接分析して議論できます。Amazon Q とイメージを共有するには、イメージをターミナルウィンドウにドラッグアンドドロップするか、 イメージモードで fs_read ツールを使用します。

イメージをドラッグアンドドロップする

Amazon Q とイメージを共有する最も簡単な方法は、イメージをターミナルウィンドウに直接ドラッグアンドドロップすることです。イメージをターミナルにドラッグすると、次のようになります。

1. イメージパスはプロンプトに自動的に挿入されます。
2. その後、Amazon Q がイメージで何をするかに関するコンテキストを提供するテキストを追加できます。
3. Amazon Q はイメージを処理し、そのコンテンツに基づいて応答します

例:

```
Amazon Q> /path/to/architecture-diagram.png Can you explain this architecture and
 generate sample code for implementing it?
```

イメージでの fs_read の使用

ツールを明示的に使用fs_readしてイメージを共有することもできます。

```
Amazon Q> Can you analyze this screenshot at /path/to/screenshot.png?
```

Amazon Q は、イメージファイルについて言及すると、イメージモードで fs_read を使用することを自動的に提案します。

イメージのユースケース

Amazon Q とイメージを共有するための一般的なユースケースは次のとおりです。

- トラブルシューティングのためのエラーメッセージのスクリーンショットの分析
- アーキテクチャ図をコード実装に変換する
- UI/UX 設計について説明し、対応する HTML/CSS を生成する
- フローチャートを理解し、アルゴリズムに変換する

- イメージとして共有されているコードスニペットの確認
- ドキュメントの技術図の解釈

サポートされている形式と制限

サポートされているイメージ形式には、JPEG/JPG、PNG、GIF、WebP などがあります。イメージのサイズは 10MB 未満で、1 回のリクエストで最大 10 個のイメージを共有できます。

最善の結果を得るには:

- クリアテキストで高解像度イメージを使用する
- Amazon Q がイメージで何をするかに関する具体的な手順を提供します。
- 複雑な図については、追加のコンテキストを提供することを検討してください。

セキュリティに関する考慮事項とベストプラクティス

Amazon Q は、システムと AWS リソースを変更できる強力な機能を提供します。セキュリティへの影響を理解し、ベストプラクティスに従うと、これらの機能を安全に使用できます。

セキュリティリスクについて

Amazon Q を使用する場合は、以下の潜在的なセキュリティリスクに注意してください。

- 意図しないシステム変更: Amazon Q はリクエストを予期しない方法で解釈し、意図しない変更が発生する可能性があります。
- AWS リソースの変更: リソースを作成、変更、または削除し、本番環境に影響を与えたり、コストが発生する可能性がある
- データ損失: ファイルを削除または上書きするコマンドは、データ損失につながる可能性があります
- セキュリティの脆弱性: コマンドが適切にレビューされない場合、システムセキュリティが侵害される可能性があります

これらのリスクは`/acceptall`、確認プロンプトをバイパスする `/tools trust-all`または を使用すると大幅に増加します。

リスクの具体的な例は次のとおりです。

- 「古いファイルのクリーンアップ」をリクエストすると、重要な設定ファイルが削除される可能性があります
- EC2 インスタンスの最適化」をリクエストすると、実行中のインスタンスが終了する可能性があります
- 「セキュリティ問題の修正」のリクエストは、機密データを公開する方法でアクセス許可を変更する可能性があります

> **Warning**
>
> AWS では、本番稼働環境で`/tools trust-all`、または機密データやリソースを操作するときに、 または `/acceptall` モードを使用することはお勧めしません。これらのモードが有効になっている場合、Amazon Q によって実行されるすべてのアクションはお客様の責任となります。

一般的なセキュリティのベストプラクティス

Amazon Q を任意の環境で使用する場合、特に機密ファイル、プライベートキー、トークン、またはその他の機密情報を持つ環境では、以下のセキュリティ対策を実装することを検討してください。

ファイルアクセスの制限

デフォルトでは、Amazon Q は毎回アクセス許可を要求せずにファイルを読み取ることができます (`fs_read` はデフォルトで信頼されています) 。機密性の高い環境では、この動作を制限できます。

```
Amazon Q> /tools untrust fs_read
```

この設定では、Amazon Q はファイルを読み取る前に明示的なアクセス許可を求めます。これにより、セッション中に Amazon Q がアクセスできるファイルを細かく制御できます。

この設定を永続的にするには、シェル起動スクリプトに追加します。

```
echo 'alias q="q --untrust-fs-read"' >> ~/.bashrc
```

これにより、すべての新しい Amazon Q `fs_read` セッションが信頼されていない で開始され、ファイルアクセスに明示的なアクセス許可が必要になります。

追加のセキュリティ対策

機密情報の高い環境では、以下の追加対策を検討してください。

- 機密情報やデータを含まない専用の開発環境で Amazon Q を使用する
- プロジェクトディレクトリの外部またはアクセス許可が制限された場所に機密ファイルを保存する
- ファイルでハードコーディングするのではなく、機密値に環境変数を使用する
- AWS API コールを行う前に、 を使用して明示的なアクセス許可/tools untrust use_awsを要求することを検討してください。
- プロジェクトルールを使用してセキュリティガイドラインと制限を定義する (プロジェクトルールの使用「」を参照)

/tools trust-all を安全に使用する

特定のワークフロー/acceptallで /tools trustallまたは を使用する必要がある場合は、以下の安全プラクティスに従ってリスクを最小限に抑えます。

- 開発環境またはテスト環境でのみ使用し、本番環境では使用しない
- 特定のタスク/tools trust-allに対してのみ有効にし、 を使用してすぐに無効に/tools resetしてデフォルトのアクセス許可に戻す
- を有効にする前に重要なデータをバックアップする /tools trust-all
- が有効になっている場合、最小限のアクセス許可で AWS 認証情報/tools trust-allを使用する
- が有効になっている間に Amazon Q が実行するすべてのアクション/tools trust-allを慎重にモニタリングする

の使用後にデフォルトのアクセス許可設定に戻るには/tools trust-all、 reset コマンドを使用します。

```
Amazon Q> /tools reset
```

これにより、すべてのツールがデフォルトのアクセス許可レベルに戻り、デフォルトでのみfs_read信頼されます。

カスタムエージェント

Amazon Q Developer CLI のカスタムエージェントは、特定のユースケースやワークフロー用に設定できるカスタマイズ可能な AI アシスタントです。広範なコンテキストとプロンプトエンジニアリングを必要とする汎用アシスタントを使用する代わりに、カスタムエージェントを使用すると、さまざまなシナリオに適したツール、アクセス許可、コンテキストのセットを事前設定できます。

たとえば、AWS ツールとドキュメントにアクセスできる AWS スペシャリストカスタムエージェント、特定のリンティングと分析ツールを持つコードレビューカスタムエージェント、またはプロジェクトのドキュメントとカスタムスクリプトを含むプロジェクト固有のカスタムエージェントを作成できます。

トピック

- 概要
- カスタムエージェントの定義
- カスタムエージェントの管理
- 設定リファレンス
- 例
- トラブルシューティング

概要

カスタムエージェントは、さまざまなユースケースに特定の設定を定義することで、Amazon Q Developer CLI の動作をカスタマイズする方法を提供します。各カスタムエージェントは、エージェントがアクセスできるツール、アクセス許可、含めるコンテキストを指定する JSON 設定ファイルによって定義されます。

重要： カスタムエージェント管理には、主に設定ファイルの作成と編集が含まれます。一部のコマンドはチャットセッション (`/agent list`や など`/agent create`) で使用できますが、カスタムエージェントを切り替えるには、 で新しいチャットセッションを開始する必要がありますq chat `--agent [name]`。

カスタムエージェントがデフォルトのエクスペリエンスとどのように異なるか

デフォルトでは、Amazon Q Developer CLI は使用可能なすべてのツールにアクセスできますが、ほとんどのオペレーションでユーザーの確認が必要です。このアプローチはセキュリティを優先しますが、頻繁なアクセス許可プロンプトでワークフローを中断する可能性があります。

カスタムエージェントは、以下を可能にすることでこれを解決します。

- 特定のツールの事前承認 - プロンプトを表示せずに実行できるツールを定義する
- ツールアクセスの制限 - 複雑さを軽減するために使用できるツールを制限する
- 関連するコンテキストを含める - プロジェクトファイル、ドキュメント、またはシステム情報を自動的にロードする
- ツールの動作を設定する - ツールの動作に関する特定のパラメータを設定します。

カスタムエージェントを使用する利点

ワークフローの最適化

AWS インフラストラクチャ管理、コードレビュー、デバッグセッションなどの特定のタスクに合わせたカスタムエージェントを作成します。

中断の軽減

信頼できるツールを事前に承認して、集中的な作業セッション中のアクセス許可プロンプトを排除します。

拡張コンテキスト

関連するプロジェクトドキュメント、設定ファイル、またはシステム情報を自動的に含めます。

チームコラボレーション

カスタムエージェント設定をチームメンバーと共有して、一貫した開発環境を確保します。

セキュリティコントロール

ツールへのアクセスを特定のワークフローに必要なもののみに制限し、潜在的なセキュリティリスクを軽減します。

MCP および組み込みツールとの関係

カスタムエージェントは、組み込みの Amazon Q Developer CLI ツールと Model Context Protocol (MCP) を通じて提供される外部ツールの両方を使用します。これにより、以下を柔軟に行うことができます。

- 組み込みツールを使用する - ファイルオペレーション、コマンド実行、AWS CLI 統合、およびその他のコア機能

- MCP サーバーの統合 - MCP サーバー設定を使用してカスタムツールとサービスを追加する
- ツールアクセスの制御 - 各ソースから利用できるツールを正確に指定します
- ツールの競合を管理する - エイリアスを使用して、さまざまなツールソース間の命名競合を処理します。

MCP 統合の詳細については、「Amazon Q Developer での MCP の使用」を参照してください。

カスタムエージェントの定義

このセクションでは、Amazon Q Developer CLI ワークフローでカスタムエージェントを作成して使用する方法について説明します。

最初のカスタムエージェントの作成

最初のカスタムエージェントを作成するためのstep-by-stepのチュートリアルを次に示します。

1. Amazon Q Developer CLI チャットセッションを開始します。

   ```
   $ q chat
   ```

2. 既存のエージェントを一覧表示して、利用可能なものを確認します。

   ```
   /agent list
   ```

3. 新しいエージェントを作成します (*my-agent* を任意の名前に置き換えます)。

   ```
   /agent create --name my-agent
   ```

   これにより、新しいエージェント設定ファイルが作成され、デフォルトのエディタで開きます (EDITOR環境変数を介して設定)。

4. 必要に応じてカスタムエージェント設定をカスタマイズします。簡単に開始するには、次の操作を行います。

   - カスタムエージェントの目的を説明する説明を追加する
   - どのツールを使用できるかを指定する
   - 頻繁に使用するツールを事前承認する

5. 設定ファイルを保存し、エディタを終了してチャットセッションに戻ります。
6. カスタムエージェントで新しいチャットセッションを開始します。

```
$ q chat --agent my-agent
```

注： 既存のチャットセッション内でカスタムエージェントを切り替えることはできません。カスタムエージェントの変更では、新しいセッションを開始する必要があります。

7. 設定したツールを使用してタスクを実行するように依頼して、カスタムエージェントをテストします。

カスタムエージェント設定オプションの詳細については、「設定リファレンス」を参照してください。

カスタムエージェントコマンド

Amazon Q Developer CLI には、カスタムエージェントを管理するためのコマンドがいくつか用意されています。これらのコマンドはチャットセッション中に使用でき、 で開始します/agent。

カスタムエージェントコマンド

コマンド	説明	可用性
`/agent list`	環境で使用可能なすべてのカスタムエージェントを表示します。	チャットセッション
`/agent schema`	カスタムエージェント設定ファイルを作成するための JSON スキーマを表示します	チャットセッション
`/agent create --name [name]`	新しいカスタムエージェント設定ファイルを作成し、デフォルトのエディタで開きます	チャットセッション

注： 一部のカスタムエージェント管理オペレーションでは、インタラクティブコマンドではなく手動ファイル編集が必要です。カスタムエージェントの変更は、新しいチャットセッションを開始すると有効になります。

レガシープロファイルからの移行

エージェントをサポートする Amazon Q Developer CLI のバージョンに更新してサインインすると、次の場合にレガシープロファイルをエージェントに移行するように求められる場合があります。

- 既存のレガシープロファイル設定がある
- インタラクティブモードで実行している (移行は でスキップされます`--no-interactive`)
- 移行プロセスをまだ完了していない

移行プロセス：

- 既存のプロファイル設定を新しいエージェント形式に変換します
- 既存のツールのアクセス許可とコンテキスト設定を保持します
- ホームディレクトリにエージェント設定ファイルを作成します。
- 既存のワークフローとの下位互換性を維持します

移行はオプションであり、拒否できます。移行しないことを選択した場合は、引き続きデフォルトのエージェントを使用するか、新しいエージェントを手動で作成できます。移行動作を制御するには、`--no-interactive`フラグを使用して移行プロンプトを完全にスキップします。

デフォルトエージェントとカスタムカスタムエージェントの使用

Amazon Q Developer CLI には、事前承認された最小限のアクセス許可ですべてのツールへのアクセスを提供する組み込みのデフォルトエージェントが含まれています。このデフォルトのエージェント：

- 使用可能なすべてのツール (組み込みおよび MCP) を含む
- セキュリティの`fs_read`ツールのみを事前承認します
- `README.md` や などの一般的なプロジェクトファイルを自動的に含める `AmazonQ.md`
- 利用可能な場合はレガシー MCP 設定を使用します

カスタムカスタムエージェントを使用すると、次のことができます。

- ツールへのアクセスを必要なもののみに制限する
- 中断を減らすために追加のツールを事前に承認する
- 特定のプロジェクトドキュメントとコンテキストファイルを含める
- 特定のユースケースに合わせてツールの動作を設定する

カスタムエージェントの管理

このセクションでは、開発ワークフローでカスタムエージェントを効果的に整理、管理、操作する方法について説明します。

カスタムエージェントファイルの場所

カスタムエージェント設定ファイルは、特定のディレクトリに JSON ファイルとして保存されます。

グローバルカスタムエージェント

```
~/.aws/amazonq/cli-agents/{agent-name}.json
```

システム上のすべてのプロジェクトとディレクトリで使用できます。

プロジェクトレベルのカスタムエージェント

```
.amazonq/cli-agents/{agent-name}.json
```

特定のプロジェクトディレクトリとそのサブディレクトリ内でのみ使用できます。

重要： ファイル名{agent-name}の は参考用です。実際のエージェント名は、JSON 設定ファイル自体内の nameフィールドによって決まります。これは必須です。

カスタムエージェントの優先順位と競合の解決

Amazon Q Developer CLI がカスタムエージェントを検索する場合、特定の優先順位に従います。

1. ローカルカスタムエージェントを最初に - 現在の作業ディレクトリ内のカスタムエージェントをチェックします
2. グローバルカスタムエージェント秒 - ホームディレクトリのカスタムエージェントにフォールバックします
3. 組み込みデフォルト - カスタムエージェントが見つからない場合はデフォルトエージェントを使用します

ローカルディレクトリとグローバルディレクトリの両方に同じ名前のカスタムエージェントが含まれている場合、ローカルカスタムエージェントが優先されます。Amazon Q Developer CLI では、これが発生すると警告メッセージが表示されます。

```
WARNING: Agent conflict for my-agent. Using workspace version.
```

この優先順位システムにより、次のことが可能になります。

- グローバルカスタムエージェントをプロジェクト固有のバージョンで上書きする
- カスタムエージェントの変更をグローバルにする前にローカルでテストする
- プロジェクトごとに異なるカスタムエージェント設定を維持する

カスタムエージェントを整理するためのベストプラクティス

グローバルカスタムエージェントを使用するタイミング

グローバルカスタムエージェントを以下に使用します。

- 汎用ワークフロー - 複数のプロジェクトで使用されるカスタムエージェント
- 個人の生産性 - 個々のワークスタイルに合わせたカスタムエージェント
- 一般的な開発タスク - コードレビュー、デバッグ、ドキュメント生成
- ツール固有のワークフロー - AWS 管理、Git オペレーション、Docker ワークフロー

ローカルカスタムエージェントを使用するタイミング

ローカルカスタムエージェントを以下に使用します。

- プロジェクト固有の設定 - 特定のプロジェクトファイルにアクセスする必要があるカスタムエージェント
- チームコラボレーション - バージョン管理を通じて共有されるカスタムエージェント
- 開発環境 - 特定のプロジェクトに固有の要件を持つカスタムエージェント
- テストと実験 - グローバル設定に影響を与えない一時的なカスタムエージェントの変更

カスタムエージェントの命名規則

カスタムエージェント組織を改善するには、次の命名規則を検討してください。

- 目的ベースの名前 - `aws-specialist.json`、`code-reviewer.json`、 `documentation-writer.json`

- テクノロジー固有の名前 - python-dev.json、react-frontend.json、 terraform-ops.json
- プロジェクト固有の名前 - project-alpha.json、mobile-app.json、 api-backend.json
- 環境固有の名前 - development.json、staging.json、 production.json

チームとのカスタムエージェントの共有

ローカルカスタムエージェントは、バージョン管理システムを通じてチームメンバーと共有できます。このアプローチにより、チームは次のことが可能になります。

- 開発環境の標準化 - すべてのチームメンバーが同じツールと設定にアクセスできることを確認します
- プロジェクト固有のコンテキストを共有する - プロジェクトドキュメント、コーディング標準、カスタムスクリプトを含める
- 一貫性を維持する - チーム全体で同じツールのアクセス許可と設定を使用する
- 改善に協力する - プルリクエストを使用してカスタムエージェント設定を確認および改善する

バージョン管理のベストプラクティス

バージョン管理を通じてカスタムエージェントを共有する場合：

- カスタムエージェントディレクトリを含める - リポジトリ.aws/amazonq/agents/に追加する
- カスタムエージェントの目的を文書化する - カスタムエージェント設定ファイルで明確な説明を使用する
- カスタムエージェントの変更を確認する - カスタムエージェント設定の変更をコード変更のように処理する
- カスタムエージェント設定をテストする - 共有カスタムエージェントがすべてのチームメンバーに対して正しく動作することを確認する
- 機密情報の回避 - API キー、パスワード、または個人情報を共有カスタムエージェントに含めないでください

設定リファレンス

カスタムエージェント設定ファイルは、カスタムエージェントの動作を定義する JSON ドキュメントです。このセクションでは、設定の概念と一般的なパターンの概要を説明します。

設定リファレンス

カスタムエージェント設定ファイル形式、使用可能なフィールド、構文の詳細については、Amazon Q Developer CLI の補足ドキュメントを参照してください。

- エージェント設定形式 - すべての設定オプションの完全なリファレンス
- 組み込みツールリファレンス - すべての組み込みツールとその設定オプションのドキュメント

以下を実行して、カスタムエージェント設定ファイルの JSON スキーマを表示することもできます。

```
/agent schema
```

設定の概念

カスタムエージェント設定ファイルには、カスタムエージェントの動作のさまざまな側面を制御するいくつかのキーセクションが含まれています。

基本メタデータ

すべてのカスタムエージェントには、識別とドキュメントの基本的なメタデータを含めることができます。

- 名前 - カスタムエージェントの識別子 (指定されていない場合はファイル名から派生)
- 説明 - カスタムエージェントの目的に関する人間が読める説明

ツール設定

ツール設定は、カスタムエージェントで使用できるツールとその動作を制御します。

ツール

組み込みツールや MCP サーバーツールなど、カスタムエージェントが使用できる可能性のあるすべてのツールを一覧表示します。

allowedTools

ユーザーの確認なしで実行できるツールを指定し、ワークフロー効率を向上させる

toolAliases

ツールの代替名を提供します。命名競合の解決やショートカットの作成に役立ちます。

toolsSettings

許可されたファイルパスやサービスアクセス許可など、個々のツールに対する特定の動作を設定します。

MCP サーバーの設定

このセクションでは、カスタムエージェントがアクセスできるモデルコンテキストプロトコルサーバー`mcpServers`を定義します。各サーバー設定には以下が含まれます。

- コマンド - MCP サーバーを起動するための実行可能コマンド
- 引数 - サーバーのコマンドライン引数
- 環境変数 - サーバープロセスの環境設定
- タイムアウト設定 - リクエストタイムアウト設定

MCP 統合の詳細については、「Amazon Q Developer での MCP の使用」を参照してください。

リソースとコンテキスト

カスタムエージェントは、次の 2 つのメカニズムを使用して関連するコンテキストを自動的に含めることができます。

リソース

カスタムエージェントのコンテキストに含めるファイルとディレクトリ。柔軟なファイル選択のための glob パターンをサポート

hooks

特定のトリガーポイント (カスタムエージェントの起動やユーザー入力など) で実行し、出力をコンテキストに含めるコマンド

一般的な設定パターン

最小カスタムエージェント設定

事前承認された読み取りアクセスを持つ基本的なファイルオペレーションを提供するシンプルなカスタムエージェント。

```
{
"name": "basic-ops",
  "description": "Basic file operations custom agent",
  "tools": [
    "fs_read",
    "fs_write",
    "execute_bash"
  ],
  "allowedTools": [
    "fs_read"
  ]
}
```

特殊なワークフローのカスタムエージェント

特定のツールのアクセス許可を持つ AWS インフラストラクチャ管理用に設定されたカスタムエージェント。

```
{
"name": "infra-manage",
  "description": "AWS infrastructure management custom agent",
  "tools": [
    "fs_read",
    "fs_write",
    "execute_bash",
    "use_aws"
  ],
  "allowedTools": [
    "fs_read",
    "use_aws"
  ],
  "toolsSettings": {
    "use_aws": {
      "allowedServices": ["s3", "lambda", "cloudformation"]
    }
  },
  "resources": [
    "file://README.md",
    "file://infrastructure/**/*.yaml",
    "file://docs/deployment.md"
  ]
}
```

フックを持つプロジェクト固有のカスタムエージェント

静的ファイルと動的コマンドの両方を通じてプロジェクトコンテキストを含むカスタムエージェント。

```
{
  "name": "project-dev",
  "description": "Project development custom agent with git context",
  "tools": [
    "fs_read",
    "fs_write",
    "execute_bash",
    "@git"
  ],
  "allowedTools": [
    "fs_read",
    "@git/git_status"
  ],
  "resources": [
    "file://README.md",
    "file://CONTRIBUTING.md",
    "file://src/**/*.md"
  ],
  "hooks": {
    "agentSpawn": [
      {
        "command": "git status --porcelain",
        "timeout_ms": 10000
      }
    ]
  }
}
```

MCP サーバー統合によるカスタムエージェント

MCP サーバーを介して外部ツールを統合するカスタムエージェント：

```
{
"name": "custom-dev",
  "description": "Development custom agent with external tool integration",
  "mcpServers": {
    "git": {
      "command": "git-mcp-server",
```

```
      "args": [],
      "timeout": 30000
    },
    "fetch": {
      "command": "fetch-mcp-server",
      "args": ["--timeout", "10"]
    }
  },
  "tools": [
    "fs_read",
    "fs_write",
    "@git",
    "@fetch/fetch_url"
  ],
  "allowedTools": [
    "fs_read",
    "@git/git_status",
    "@fetch/fetch_url"
  ],
  "toolAliases": {
    "@git/git_status": "status",
    "@fetch/fetch_url": "get"
  }
}
```

例

このセクションでは、一般的な開発ワークフローとユースケースのカスタムエージェント設定の実例を示します。

AWS スペシャリストカスタムエージェント

このカスタムエージェントは、AWS インフラストラクチャの管理および開発タスク用に最適化されています。関連するドキュメントを含めながら、AWS ツールへの事前承認されたアクセスを提供します。

Example AWS スペシャリストカスタムエージェント設定

```
{
  "description": "Specialized custom agent for AWS infrastructure and development
 tasks",
  "tools": [
    "fs_read",
```

```
    "fs_write",
    "execute_bash",
    "use_aws"
  ],
  "allowedTools": [
    "fs_read",
    "use_aws"
  ],
  "toolsSettings": {
    "use_aws": {
      "allowedServices": [
        "s3",
        "lambda",
        "cloudformation",
        "ec2",
        "iam",
        "logs"
      ]
    },
    "fs_write": {
      "allowedPaths": [
        "infrastructure/**",
        "scripts/**",
        "*.yaml",
        "*.yml",
        "*.json"
      ]
    }
  },
  "resources": [
    "file://README.md",
    "file://infrastructure/**/*.yaml",
    "file://infrastructure/**/*.yml",
    "file://docs/aws-setup.md",
    "file://scripts/deploy.sh"
  ],
  "hooks": {
    "agentSpawn": [
      {
        "command": "aws sts get-caller-identity",
        "timeout_ms": 10000,
        "cache_ttl_seconds": 300
      }
    ]
```

```
  }
}
```

このカスタムエージェントのユースケース：

- CloudFormation スタックのデプロイ
- S3 バケットと Lambda 関数の管理
- AWS サービスの問題のトラブルシューティング
- Infrastructure as Code の確認と更新

開発ワークフローのカスタムエージェント

このカスタムエージェントは、コードレビュー、テスト、Git オペレーションなどの一般的なソフトウェア開発タスク用に設計されています。

Example 開発ワークフローのカスタムエージェント設定

```
{
  "description": "General development workflow custom agent with Git integration",
  "mcpServers": {
    "git": {
      "command": "git-mcp-server",
      "args": [],
      "timeout": 30000
    }
  },
  "tools": [
    "fs_read",
    "fs_write",
    "execute_bash",
    "@git"
  ],
  "allowedTools": [
    "fs_read",
    "@git/git_status",
    "@git/git_log",
    "@git/git_diff"
  ],
  "toolAliases": {
    "@git/git_status": "status",
    "@git/git_log": "log",
```

```
    "@git/git_diff": "diff"
  },
  "toolsSettings": {
    "fs_write": {
      "allowedPaths": [
        "src/**",
        "tests/**",
        "docs/**",
        "*.md",
        "*.json",
        "package.json",
        "requirements.txt"
      ]
    }
  },
  "resources": [
    "file://README.md",
    "file://CONTRIBUTING.md",
    "file://docs/**/*.md",
    "file://package.json",
    "file://requirements.txt"
  ],
  "hooks": {
    "agentSpawn": [
      {
        "command": "git status --porcelain",
        "timeout_ms": 5000
      },
      {
        "command": "git branch --show-current",
        "timeout_ms": 3000
      }
    ]
  }
}
```

このカスタムエージェントのユースケース :

- コードのレビューと分析
- テストの記述と更新
- Git ワークフロー管理
- ドキュメントの更新

- 依存関係管理

コードレビューカスタムエージェント

このカスタムエージェントは、コード品質、セキュリティ、ベストプラクティスを分析するために最適化されたツールとコンテキストを使用して、コードレビュータスクに特に焦点を当てています。

Example コードレビューのカスタムエージェント設定

```
{
  "description": "Specialized custom agent for code review and quality analysis",
  "tools": [
    "fs_read",
    "execute_bash"
  ],
  "allowedTools": [
    "fs_read",
    "execute_bash"
  ],
  "toolsSettings": {
    "execute_bash": {
      "allowedCommands": [
        "grep",
        "find",
        "wc",
        "head",
        "tail",
        "cat",
        "diff",
        "git diff",
        "git log",
        "eslint",
        "pylint",
        "rubocop"
      ]
    }
  },
  "resources": [
    "file://CONTRIBUTING.md",
    "file://docs/coding-standards.md",
    "file://docs/security-guidelines.md",
    "file://.eslintrc.json",
    "file://.pylintrc",
```

```
    "file://pyproject.toml"
  ],
  "hooks": {
    "agentSpawn": [
      {
        "command": "git diff --name-only HEAD~1",
        "timeout_ms": 5000,
        "max_output_size": 2048
      }
    ],
    "userPromptSubmit": [
      {
        "command": "find . -name '*.py' -o -name '*.js' -o -name '*.ts' | wc -l",
        "timeout_ms": 3000,
        "cache_ttl_seconds": 60
      }
    ]
  }
}
```

このカスタムエージェントのユースケース :

- コード品質のプルリクエストの確認
- セキュリティの脆弱性を特定する
- コーディング標準への準拠の確認
- コードの複雑さと保守性の分析
- 改善の提案とリファクタリングの機会

プロジェクト固有のカスタムエージェント

この例では、プロジェクト固有のツール、ドキュメント、ビルドプロセスなど、特定のプロジェクトに合わせたカスタムエージェントを作成する方法を示します。

Example プロジェクト固有のカスタムエージェント設定

```
{
  "description": "Custom agent for the mobile app backend project",
  "mcpServers": {
    "docker": {
      "command": "docker-mcp-server",
      "args": ["--socket", "/var/run/docker.sock"]
```

```
    },
    "database": {
      "command": "postgres-mcp-server",
      "args": ["--connection", "postgresql://localhost:5432/myapp"],
      "env": {
        "PGPASSWORD": "$DATABASE_PASSWORD"
      }
    }
  },
  "tools": [
    "fs_read",
    "fs_write",
    "execute_bash",
    "@docker",
    "@database"
  ],
  "allowedTools": [
    "fs_read",
    "@docker/docker_ps",
    "@docker/docker_logs",
    "@database/query_read_only"
  ],
  "toolAliases": {
    "@docker/docker_ps": "containers",
    "@docker/docker_logs": "logs",
    "@database/query_read_only": "query"
  },
  "toolsSettings": {
    "fs_write": {
      "allowedPaths": [
        "src/**",
        "tests/**",
        "migrations/**",
        "docker-compose.yml",
        "Dockerfile",
        "requirements.txt"
      ]
    },
    "execute_bash": {
      "allowedCommands": [
        "npm test",
        "npm run build",
        "python manage.py test",
        "docker-compose up",
```

```
        "docker-compose down"
      ]
    }
  },
  "resources": [
    "file://README.md",
    "file://docs/api-documentation.md",
    "file://docs/database-schema.md",
    "file://docker-compose.yml",
    "file://requirements.txt",
    "file://src/config/settings.py"
  ],
  "hooks": {
    "agentSpawn": [
      {
        "command": "docker-compose ps",
        "timeout_ms": 10000,
        "cache_ttl_seconds": 30
      },
      {
        "command": "git status --porcelain",
        "timeout_ms": 5000
      }
    ]
  }
}
```

このカスタムエージェントのユースケース :

- Docker コンテナとサービスの管理
- データベースクエリと移行の実行
- アプリケーションの構築とテスト
- 本番稼働に関する問題のデバッグ
- API ドキュメントの更新

効果的なカスタムエージェントを作成するためのヒント

- シンプルな開始 - 基本的なツール設定から開始し、必要に応じて複雑さを追加します
- わかりやすい名前を使用する - 目的を明確に示すカスタムエージェント名を選択する
- 関連するコンテキストを含める - プロジェクトドキュメントと設定ファイルをリソースに追加する

- 安全なツールの事前承認 - allowedTools に頻繁に使用される低リスクのツールを含める
- 動的コンテキストにフックを使用する - コマンドフックを使用して現在のシステム状態を含める
- ツールの範囲を制限する - toolsSettings を使用して、関連するパスとサービスへのツールアクセスを制限する
- 徹底的にテストする - カスタムエージェント設定が期待どおりに動作することを確認する
- カスタムエージェントを文書化する - チームメンバーがカスタムエージェントの目的を理解できるように、明確な説明を使用します。

トラブルシューティング

このセクションでは、カスタムエージェントを使用する際に発生する可能性がある一般的な問題とその解決方法について説明します。

設定エラー

無効な JSON 構文

問題： カスタムエージェントが JSON 解析エラーでロードできません。

症状：

- 「無効な JSON」または「構文エラー」に関するエラーメッセージ
- カスタムエージェントが に表示されない `/agent list`
- デフォルトのエージェント動作へのフォールバック

解決方法

- JSON バリデータまたは linter を使用して JSON を検証する
- 一般的な JSON エラーを確認します。
  - 配列要素またはオブジェクトプロパティ間にカンマがない
  - 最後の要素の後の末尾のカンマ
  - 一致しない括弧または中括弧
  - 文字列値のエスケープされていない引用符
- `/agent schema` を使用して設定構造を検証する

スキーマ検証エラー

問題： カスタムエージェント設定が想定されたスキーマと一致しません。

症状：

- 不明な設定フィールドに関する警告
- カスタムエージェントの動作が設定と一致しません
- 必須フィールドの欠落エラー

解決方法

- を使用してスキーマと設定を比較する `/agent schema`
- 入力ミスのフィールド名を確認する (例: `allowedTools`と `allowedTool`)
- データ型がスキーマ要件 (配列と文字列、ブール値と文字列) に一致することを確認する
- 補足の Amazon Q Developer CLI ドキュメントのエージェント形式のドキュメントで正しい構文を確認してください。

カスタムエージェントのロードの問題

カスタムエージェントが見つかりません

問題： カスタムエージェントがリストに表示されないか、使用できません。

症状：

- `/agent list` カスタムエージェントが表示されない
- `/agent use [name]` が「エージェントが見つかりません」で失敗する
- 警告なしでデフォルトエージェントにフォールバックする

解決方法

- カスタムエージェントファイルが正しい場所にあることを確認します。
  - グローバル: `~/.aws/amazonq/cli-agents/[name].json`
  - ローカル: `amazonq/cli-agents/[name].json`
- ファイルのアクセス許可を確認する - ファイルが読み取り可能であることを確認します
- ファイル名が、使用しようとしているカスタムエージェント名と一致することを確認します。

- ファイルに.json拡張子があることを確認する

カスタムエージェントバージョンのロードが正しくない

問題： カスタムエージェントの別のバージョンが想定以上にロードされています。

症状：

- カスタムエージェントの動作が最近の設定変更と一致しません
- カスタムエージェントの競合に関する警告メッセージ
- 予期しないツールの可用性またはアクセス許可

解決方法

- ローカルディレクトリとグローバルディレクトリ間のカスタムエージェント名の競合を確認する
- ローカルカスタムエージェントはグローバルカスタムエージェントよりも優先されることに注意してください。
- `/agent list` を使用して、ロードされているバージョンを確認します。
- 必要に応じて、競合するカスタムエージェントファイルを削除または名前を変更する

ツールのアクセス許可の問題

ツールは利用できません

問題： カスタムエージェントは、設定したツールにアクセスできません。

症状：

- 不明または使用できないツールに関するエラーメッセージ
- でツールのアクセス許可を要求するカスタムエージェント `allowedTools`
- MCP サーバーツールが機能しない

解決方法

- ツール名がtools配列で正しくスペルされていることを確認する
- MCP ツールの場合、サーバーが で正しく設定されていることを確認します。 `mcpServers`
- MCP サーバーが実行されていてアクセス可能であることを確認します。

- MCP ツールには正しい構文を使用します。 `@server_name/tool_name`
- 組み込みツール名を Amazon Q Developer CLI の補足ドキュメントの組み込みツールドキュメントと照合する

/tools コマンドは空のリストを返します

問題： `/tools` コマンドは、使用可能なツールがないか、想定よりも少ないツールを表示します。

症状：

- `/tools` が空のリストを返す
- 予想されるツールがツールリストにない
- カスタムエージェントには機能がないように見える

一般的な原因：

- カスタムエージェント設定の空のtools配列
- `tools` 配列内のツール名の入力ミス
- MCP サーバーツール名が正しくない (サーバープレフィックスがない)
- ツールのロードを妨げる MCP サーバー設定の問題

解決方法

- カスタムエージェント設定に有効なツール名のtools配列が含まれていることを確認します。
- ツール名が正しくスペルされていることを確認する (大文字と小文字が区別される)
- MCP ツールの場合は、正しいサーバープレフィックス形式を使用していることを確認してください。 `server-name___tool-name`
- デフォルトのエージェントでテストして、ツールが使用可能であることを確認します。 `q chat /tools`
- 外部ツールを使用している場合は MCP サーバーのステータスを確認する

予期しないアクセス許可プロンプト

問題： カスタムエージェントは、事前承認されたと思われるツールのアクセス許可を求めます。

症状：

- にリストされているツールのアクセス許可プロンプト allowedTools
- カスタムエージェント設定にもかかわらずワークフローが中断する

解決方法

- ツールが 配列toolsと allowedTools配列の両方にリストされていることを確認する
- 2 つの配列間のツール名のタイプミスをチェックする
- MCP ツールの場合は、 で完全なサーバープレフィックス名を使用します。 allowedTools
- が正しく適用toolAliasesされていることを確認する

カスタムエージェントの動作のデバッグ

コンテキストまたはリソースがない

問題： カスタムエージェントは、予想されるファイルやコンテキストにアクセスできないようです。

解決方法

- resources 配列内のファイルパスが正しく、ファイルが存在することを確認する
- リソースの glob パターンが目的のファイルと一致していることを確認します。
- フックコマンドが正常に実行され、出力が生成されていることを確認する
- フックコマンドを手動でテストして、環境で動作することを確認します。
- コマンドがカットオフされている場合はフックタイムアウト設定を確認する

MCP サーバーの問題

問題： MCP サーバーが機能していないか、ツールを使用できません。

解決方法

- MCP サーバーコマンドが正しく、実行可能ファイルが PATH にあることを確認する
- 必要な環境変数が設定されていることを確認します。
- MCP サーバーを個別にテストして動作していることを確認する
- MCP サーバーログでエラーメッセージを確認する
- サーバーの起動が遅い場合はタイムアウト値を増やす

- MCP のトラブルシューティングの詳細については、「Amazon Q Developer での MCP の使用」を参照してください。

カスタムエージェント設定のテスト

カスタムエージェント設定を体系的にテストするには：

1. JSON 検証ツールを使用して JSON 構文を検証する
2. を使用してスキーマに対して設定を確認する `/agent schema`
3. でカスタムエージェントのロードをテストする `/agent list`
4. でカスタムエージェントに切り替える `/agent use [name]`
5. 各ツールを個別にテストしてアクセスとアクセス許可を検証する
6. リソースとフックが予想されるコンテキストを提供していることを確認する
7. 一般的なワークフローをテストして、カスタムエージェントが期待どおりに動作することを確認する

追加のヘルプの取得

エージェントで引き続き問題が発生した場合：

- 補足の Amazon Q Developer CLI ドキュメントにある完全なエージェント設定ドキュメントを確認する
- ツール固有の設定については、組み込みツールリファレンスを確認してください。
- MCP 関連の問題については、MCP ドキュメントを参照してください。
- よりシンプルなエージェント設定から開始し、徐々に複雑さを追加する
- 設定を エージェントの例とユースケースの例と比較する
- エージェントの切り替えと編集には、セッション内コマンドを使用するのではなく、新しいチャットセッションを開始する必要があることに注意してください。

CLI でエディタコマンドを使用する

Amazon Q Developer CLI には、任意のテキストエディタを開いて複雑なプロンプトを作成する `/editor` コマンドが用意されています。これは、複数行のプロンプト、コード例、または質問を慎重に構造化する必要がある場合に特に便利です。

基本的な使用法

空のプロンプトでデフォルトのエディタを開くには：

```
Amazon Q> /editor
```

最初のテキストでエディタを開くには：

```
Amazon Q> /editor Write a Python function that calculates Fibonacci numbers
```

/editor コマンドを使用すると、Amazon Q は .md拡張子付きの一時ファイルを作成し、このファイルで指定されたエディタを開き、エディタを保存して閉じるときにコンテンツを読み取り、プロンプトとして送信します。

任意のエディタの設定

Amazon Q は、システムの$EDITOR環境変数を使用して、開くエディタを決定します。設定されていない場合、デフォルトで になりますvi。

一時的な設定 (現在のセッションのみ)

エディタを現在のターミナルセッションのみに設定するには：

```
$ export EDITOR=nano
```

永続設定

エディタの設定をセッション間で永続化するには、シェル設定ファイルに export コマンドを追加します。

```
# For bash (add to ~/.bashrc)
export EDITOR=nano

# For zsh (add to ~/.zshrc)
export EDITOR=nano

# For fish shell (add to ~/.config/fish/config.fish)
set -x EDITOR nano
```

設定ファイルを編集したら、ターミナルを再起動するか、ファイルをソースにします。

```
$ source ~/.bashrc  # or ~/.zshrc
```

一般的なエディタオプション

使用できる一般的なエディタオプションを以下に示します。

- `vi` または `vim` - Vi/Vim テキストエディタ
- `nano` - Nano テキストエディタ (初心者向け)
- `emacs` - Emacs テキストエディタ
- `code -w` - Visual Studio Code (VS Code CLI をインストールする必要があります)
- `subl -w` - Sublime Text (Sublime CLI をインストールする必要があります)

> **Note**
>
> GUI エディタの `-w`フラグは、ファイルが閉じられるまでターミナルを待機させるため、重要です。

仕組み

`/editor` コマンドは次のワークフローに従います。

1. `/editor` コマンドを使用すると、Amazon Q は `.md`拡張子付きの一時ファイルを作成します。
2. 指定されたエディタがこのファイルで開きます
3. エディタにプロンプトを書き込んでファイルを保存する
4. エディタを閉じるとき
5. Amazon Q はコンテンツを読み取り、プロンプトとして送信します
6. 一時ファイルは自動的にクリーンアップされます

エディタでのコードの使用

エディタでコードを記述すると、エディタを閉じるとコンテンツ全体がプロンプトとして Amazon Q に送信されます。コードはローカルで実行されず、AI のテキスト入力として扱われます。

例: コードの記述と送信

1. と入力/editorして エディタを開きます
2. エディタで Python スクリプトを記述します。

```
def fibonacci(n):
    if n <= 1:
        return n
    else:
        return fibonacci(n-1) + fibonacci(n-2)

# This function seems inefficient
# How can I improve it?
```

3. エディタを保存して閉じる
4. Amazon Q はこのテキスト全体をプロンプトとして受け取り、コードを改善するための提案で応答します

このアプローチは、以下に役立ちます。

- コードレビューの取得
- 最適化を求める
- 複雑なコード構造の説明
- デバッグヘルプのコンテキストの提供

を他のコマンドと組み合わせる

/editor コマンドは、他の Amazon Q CLI コマンドと組み合わせるとさらに強力になります。ワークフローを強化するための実用的な組み合わせをいくつか紹介します。

/compact での /editor の使用

/compact コマンドを使用すると、Amazon Q レスポンスがより簡潔になります。この組み合わせは、効率的なコードレビューに最適です。

```
Amazon Q> /editor
# Write in the editor:
Please review this Python function that calculates prime numbers:
```

```
def is_prime(n):
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False
    i = 5
    while i * i <= n:
        if n % i == 0 or n % (i + 2) == 0:
            return False
        i += 6
    return True

# Save and close

Amazon Q> /compact
# This makes Amazon Q provide a concise code review
```

/context で /editor を使用する

/context コマンドは、会話コンテキストにファイルを追加します。この組み合わせは、他のファイルを参照するコードについて説明するのに役立ちます。

```
Amazon Q> /context path/to/config.json
Amazon Q> /editor
# Write in the editor:
Given the config.json file I just shared, please help me write a Python function that:
1. Loads the configuration
2. Validates all required fields are present
3. Returns a validated config object

# Save and close
```

/clear での /editor の使用

/clear コマンドは新しい会話を開始します。この組み合わせは、トピックを切り替える際に役立ちます。

```
Amazon Q> /clear
Amazon Q> /editor
# Write in the editor:
```

```
I want to start a new discussion about AWS Lambda cold starts.
What are the best practices for minimizing cold start times for Python Lambda
 functions?

# Save and close
```

複数ステップの会話に /editor を使用する

/editor コマンドは、使用するたびに新しい一時ファイルを作成します。会話で複数回使用して、以前のレスポンスに基づいて構築できます。

```
# First use of editor for initial complex question
Amazon Q> /editor
# Write in editor:
I need to design a database schema for a library management system.
Requirements:
- Track books, authors, publishers
- Handle member checkouts and returns
- Support reservations and waiting lists
- Generate overdue notices

# After getting Amazon Q's response with initial schema design

# Second use of editor for follow-up with specific implementation details
Amazon Q> /editor
# Write in editor:
Based on your proposed schema, I have some follow-up questions:
1. How would you modify the Member table to support different membership tiers?
2. What indexes would you recommend for optimizing checkout queries?
3. Can you show me SQL to create the Books and Authors tables with proper
 relationships?
```

このアプローチの利点は、コマンドラインにすべてを入力することなく、前の会話を参照する複雑なフォローアップ質問を慎重に作成できることです。各エディタセッションでは、Amazon Q の以前のレスポンスに基づいて詳細な質問を構成するためのスペースとフォーマットの制御が提供されます。

/profile で /editor を使用する

特殊な質問にエディタを使用する前に、別のコンテキストプロファイルに切り替えます。

```
Amazon Q> /profile set aws-developer
Amazon Q> /editor
```

```
# Write detailed AWS-specific questions that benefit from the AWS developer profile
 context
```

/help での /editor の使用

コマンドオプションが不明な場合は、 /helpの前に を使用できます/editor。

```
Amazon Q> /help editor
# Review the help information
Amazon Q> /editor
# Use the editor with better understanding of available options
```

コマンドの組み合わせのベストプラクティス

1. 特定のファイルを参照する必要がある場合は、 /editor /contextより前に使用してください。
2. 複雑な質問への簡潔な回答が必要な場合は/compact、 /editorより前に使用してください。
3. まったく新しいトピックを開始する/editorときに より/clear前に を使用する
4. フォローアップ質問を慎重に作成する必要がある複雑なマルチパート会話には、複数の/editorセッションを使用します。
5. を使用する前に現在のプロファイルを検討/editorし、適切なコンテキストであることを確認します。

効果的な使用のためのヒント

- エディタを使用して、慎重な構造化のメリットを享受する複雑なプロンプトを作成する
- 適切なインデントを含むコード例を含める
- 明確なセクションを使用してマルチパート質問を整理する
- Markdown 形式を使用して構造を改善する
- 空のファイルを保存すると、プロンプトは送信されません。

トラブルシューティング

- エディタが開いていない: $EDITOR 環境変数が正しく設定されていることを確認します
- 「そのようなファイルやディレクトリがない」エラー: エディタコマンドが PATH に と がインストールされていることを確認します。

- ターミナルのハングアップ: GUI エディタの場合は、必ず待機フラグ (例: -w) を使用してください。
- コンテンツが送信されていない: エディタを閉じる前にファイルを保存していることを確認します

Git 対応ファイル選択の使用

Git リポジトリを使用する場合、Q CLI のあいまいファインダーは Git 対応であるため、コンテキストに関連するファイルを簡単に選択して追加できます。この機能は、Git リポジトリの一部であるファイルをすばやく識別して含めるのに役立ちます。

Git 対応の選択の仕組み

Git 対応のあいまいファインダーは、リポジトリの Git 情報と自動的に統合され、拡張されたファイル選択機能を提供します。

- リポジトリ内の Git 追跡ファイルを認識する
- ファイルとともに Git ステータスインジケータを表示する
- Git 履歴に基づいて関連ファイルを優先順位付けする

Git 対応ファイル選択の使用

git 対応ファイル選択を使用するには

1. ターミナルで Git リポジトリに移動します。
2. コンテキスト add コマンドを実行します。

   ```
   /context add
   ```

3. あいまいファインダーインターフェイスには、Git ステータスインジケータを含むリポジトリのファイルが表示されます。

   - M – 変更されたファイル
   - A – ファイルの追加
   - ? – 追跡されていないファイル
4. と入力して、検索条件の一部として Git ステータスを使用してファイルをフィルタリングします。

5. 矢印キーを使用して移動し、Enter キーを押してコンテキストに追加するファイルを選択します。

Git 対応の選択に関するヒント

- 検索で Git ステータスインジケータを使用して、変更されたファイルまたは追跡されていないファイルをすばやく検索する
- あいまいファインダーは、Git 履歴で最近変更されたファイルを優先します
- Git で無視されたファイル (.gitignore 経由) は引き続き使用できますが、結果では優先順位は解除されます。

コンテキスト管理

コンテキストの管理

コンテキストファイルには、会話中に Amazon Q で考慮すべき情報が含まれています。これには、プロジェクト要件、コーディング標準、開発ルール、または Amazon Q がより関連性の高いレスポンスを提供するのに役立つその他の情報が含まれます。

コンテキストを設定する方法は 2 つあります。

- エージェントリソース (推奨): エージェント設定ファイルで定義された永続コンテキスト
- セッションコンテキスト: `/context` コマンドを使用してチャットセッション中に追加された一時的なコンテキスト

エージェントリソースを使用した永続コンテキストの設定

コンテキストを設定する推奨方法は、エージェント設定ファイルの `resources`フィールドを使用することです。これにより、エージェントを使用するたびに使用できる永続コンテキストが作成されます。

エージェント設定の`resources`配列にファイルパスまたは glob パターンを追加します。

```
{
  "name": "my-agent",
  "description": "My development agent",
  "resources": [
```

```
    "file://README.md",
    "file://.amazonq/rules/**/*.md",
    "file://docs/**/*.md",
    "file://src/config.py"
  ]
}
```

リソースをコンテキストファイルとして含めfile://るには、 にプレフィックスを付ける必要があります。これらのファイルは、このエージェントを使用するすべてのチャットセッションで自動的に使用できます。

一時的なセッションコンテキストの追加

/context add コマンドを使用して、現在のチャットセッションにファイルを一時的に追加できます。これらの追加は現在のセッションでのみ使用でき、新しいチャットセッションを開始すると保持されません。

```
q chat
> /context add README.md
Added 1 path(s) to context.
Note: Context modifications via slash command is temporary.
```

glob パターンを使用して、一度に複数のファイルを追加することもできます。

```
q chat
> /context add docs/*.md
Added 3 path(s) to context.
Note: Context modifications via slash command is temporary.
```

コンテキストの変更を永続的にするには、代わりに ファイルをエージェントの resourcesフィールドに追加します。

コンテキストの表示

現在のコンテキストを表示するには、 /context show コマンドを使用します。

```
q chat
> /context show
# Agent (my-agent):
    README.md (1 match)
    .amazonq/rules/**/*.md (3 matches)
    docs/**/*.md (5 matches)
```

```
# Session (temporary):
    <none>

5 matched files in use:
# README.md (~250 tkns)
# .amazonq/rules/security.md (~180 tkns)
# .amazonq/rules/coding-standards.md (~320 tkns)
# docs/architecture.md (~150 tkns)
# docs/best-practices.md (~200 tkns)

Total: ~1100 tokens
```

出力には以下が表示されます。

- # エージェント: エージェントのresourcesフィールドからの永続コンテキスト
- # セッション: 現在のセッション中に追加された一時的なコンテキスト

コンテキストの削除

現在のセッションコンテキストからファイルを削除するには、 `/context rm` コマンドを使用します。

```
q chat
> /context rm src/temp-file.py
Removed 1 path(s) from context.
Note: Context modifications via slash command is temporary.
```

すべてのセッションコンテキストをクリアするには、 `/context clear` コマンドを使用します。

```
q chat
> /context clear
Cleared context
Note: Context modifications via slash command is temporary.
```

注： `/context` コマンドを使用してエージェント定義のコンテキストを削除することはできません。コンテキストを完全に削除するには、エージェントの `resources`フィールドを編集します。

一般的なユースケース

コンテキスト管理の一般的なユースケースは次のとおりです。

プロジェクトルールの使用

Amazon Q は、セキュリティガイドラインと制限を定義できるプロジェクトレベルのルールをサポートしています。これらのルールは、プロジェクトの `.amazonq/rules` ディレクトリにある Markdown ファイルで定義されます。

たとえば、以下を指定するルールを作成できます。

- Amazon Q がアクセスを避けるディレクトリ
- 生成されたコードのセキュリティ要件
- コーディング標準とベストプラクティス

プロジェクトルールを含めるには、エージェント設定を使用することをお勧めします。

```
{
  "name": "my-project-agent",
  "resources": [
    "file://.amazonq/rules/**/*.md",
    "file://README.md",
    "file://docs/architecture.md"
  ]
}
```

現在のセッションにプロジェクトルールを一時的に追加することもできます。

```
q chat
> /context add .amazonq/rules/*.md
Added 3 path(s) to context.
Note: Context modifications via slash command is temporary.
```

プロジェクトルールの作成と使用の詳細については、IDE ドキュメントの「Amazon Q Developer チャットで使用するプロジェクトルールの作成」を参照してください。

セッションコンテキストからエージェントリソースへの移行

`/context add` コマンドを使用して同じコンテキストファイルを繰り返し追加する場合は、永続化のためにエージェントの `resources`フィールドに移動することを検討してください。

1. で頻繁に追加するファイルを書き留めます。 `/context add`

2. エージェント設定ファイルを編集する

3. `file://` プレフィックス付きの`resources`配列にファイルパスを追加する

4. エージェント設定を保存する

移行の例：

```
# Instead of running these commands every session:
> /context add README.md
> /context add docs/*.md
> /context add .amazonq/rules/*.md

# Add them to your agent config once:
{
  "resources": [
    "file://README.md",
    "file://docs/**/*.md",
    "file://.amazonq/rules/**/*.md"
  ]
}
```

コンテキストフックの使用

コンテキストフックは Amazon Q Developer CLI の機能で、Q Developer との会話にコンテキストを自動的に挿入するために使用できます。`/context` コマンドでコンテキストを手動で追加する代わりに、コンテキストフックはコマンドを実行し、出力をコンテキストとして含めます。

コンテキストフックのタイプ

Q Developer CLI は、次の 2 種類のコンテキストフックをサポートしています。

会話開始フック

会話の開始時に を 1 回実行します。出力は会話コンテキストに追加され、セッション全体で保持されます。

プロンプトごとのフック

各ユーザーメッセージで を実行します。出力は現在のプロンプトにのみ追加されます。

コンテキストフックの管理

Q Developer CLI の `/context hooks` コマンドを使用して、コンテキストフックを管理できます。

フックを表示する

グローバルレベルとプロファイルレベルの両方ですべての設定済みフックを表示するには：

```
/context hooks
```

このコマンドは、現在のプロファイルとグローバルレベルで設定されているすべてのフックを、そのステータス (有効または無効) とともに表示します。

フックを追加する

新しいコンテキストフックを追加するには：

```
/context hooks add [--global] <name> --trigger <trigger> --command <command>
```

<name>

フックの一意の名前

< トリガー >

`per_prompt` または `conversation_start`。

< コマンド >

実行するシェルコマンド

--グローバル

(オプション) 現在のプロファイルの代わりにフックをグローバル設定に追加する

Example git ステータスフックの追加

```
/context hooks add git-status --trigger per_prompt --command "git status --short"
```

フックを削除する

既存のコンテキストフックを削除するには：

```
/context hooks rm <name> [--global]
```

フックを有効または無効にする

特定のフックを有効にするには：

```
/context hooks enable [--global] <name>
```

特定のフックを無効にするには：

```
/context hooks disable [--global] <name>
```

すべてのフックを有効にするには：

```
/context hooks enable-all [--global]
```

すべてのフックを無効にするには：

```
/context hooks disable-all [--global]
```

フックに関するヘルプを取得する

フックコマンドの詳細なヘルプを表示するには：

```
/context hooks help
```

設定

フックの設定は、次の場所で確認できます。

フックがグローバルの場合：

```
~/.aws/amazonq/global_context.json
```

フックがプロファイルベースの場合：

```
~/.aws/amazonq/profiles/profile-name/context.json
```

ユースケースの例

次の例は、コンテキストフックの一般的なユースケースを示しています。

Git ステータス

各プロンプトで git ステータスを表示するフックを追加します。

```
/context hooks add git-status --trigger per_prompt --command "git status --short"
```

このフックは、各プロンプトの前に `git status --short` コマンドを実行し、Q Developer との会話に出力を含めます。これにより、バージョン管理について質問したり、git コマンドに関するヘルプが必要な場合に、Q Developer が git リポジトリの現在の状態を把握できます。

プロジェクト情報

会話の開始時にプロジェクト情報を表示するフックを追加します。

```
/context hooks add project-info --trigger conversation_start --command "echo 'Project:
 '$(basename $(pwd))"
```

このフックは会話の開始時に 1 回実行され、コンテキストに現在のプロジェクト名が含まれます。これは、Q Developer がプロジェクトに固有のより関連性の高いレスポンスを提供するのに役立ちます。

> **Tip**
>
> このフックを展開して、プログラミング言語、フレームワーク、プロジェクト構造など、より多くのプロジェクト情報を含めることができます。

動作と制限

コンテキストフックを使用する場合は、次の動作と制限に注意してください。

- フックは並行して実行され、応答時間への影響を最小限に抑える
- フック出力はフォーマットされ、コンテキスト内で明確にマークされます

- `/clear` コマンドは会話開始フックを再評価します
- フックは Q Developer CLI 設定に保存されます。
- 現在、インラインコマンドフックのみがサポートされています
- フック出力はフックあたり 10KB に制限されます
- デフォルトでは、フックは 5 秒後にタイムアウトします

セキュリティに関する考慮事項

コンテキストフックを使用する場合は、次のセキュリティのベストプラクティスを考慮してください。

- コンテキストフックは、現在のユーザーアクセス許可でシェルコマンドを実行します。フックに含めるコマンドには、ユーザーアカウントと同じアクセス権限があるため、注意してください。
- 認証情報や個人データなどの機密情報を公開する可能性のあるフックを追加するときは注意してください。フックの出力を確認して、共有したくない情報が含まれていないことを確認します。
- フックが必要ない場合は、無効にすることを検討してください。これにより、誤って機密情報を公開するリスクが軽減されます。
- フックを使用して、システムまたはデータを変更する可能性のあるコマンドを実行しないでください。フックは、変更を行うためではなく、情報の読み取りに使用する必要があります。
- 共有環境でフックを使用する場合は注意してください。Q Developer CLI セッションにアクセスできる他のユーザーは、フックの出力を表示できる場合があります。

> ⚠ Warning
>
> パスワード、アクセスキー、またはその他の認証情報を公開するコマンドをフックに含めないでください。この情報は会話履歴に保存され、他のユーザーに公開される可能性があります。

トラブルシューティング

コンテキストフックで問題が発生した場合は、次のソリューションを試してください。

コンテキストフックのトラブルシューティング

問題	ソリューション
フックが実行されない	フックが で有効になっていることを確認する `/context hooks`
コマンドが失敗する	ターミナルでコマンドを直接テストして、動作することを確認します。
出力が大きすぎる	コマンドを変更して出力サイズを制限する
フックのタイムアウト	コマンドを最適化して実行を高速化する

Amazon Q の設定

Amazon Q には、設定を通じて動作をカスタマイズするさまざまな方法が用意されています。これらの設定には、グラフィカルインターフェイスとコマンドラインオプションの両方からアクセスできます。

設定へのアクセス

Amazon Q 設定には、次の 2 つの方法でアクセスできます。

- 設定 GUI: `q settings`を実行してグラフィカル設定インターフェイスを開きます
- コマンドライン: さまざまなコマンドを使用して設定を直接表示および変更する

コマンドライン設定の管理

次のコマンドを使用して、コマンドラインから直接 Amazon Q 設定を管理できます。

基本設定コマンド

コマンド	説明
`q settings`	設定 GUI インターフェイスを開きます
`q settings all`	現在の設定をすべて一覧表示します

コマンド	説明
`q settings all -f json-pretty`	フォーマットされた JSON のすべての設定を一覧表示します
`q settings open`	デフォルトのエディタで設定ファイルを開きます
`q settings [KEY] [VALUE]`	特定の設定を表示または設定する
`q settings -d [KEY]`	特定の設定を削除します。
`q settings chat.defaultAgent [agent name]`	デフォルトエージェントを指定します。

`q settings` コマンドを使用する場合は、出力形式を指定できます。

```
q settings -f [FORMAT]
```

使用可能な形式 :

- `plain`: 結果をマークダウンとして出力します (デフォルト)
- `json`: 結果を JSON として出力します
- `json-pretty`: 結果をフォーマットされた JSON として出力します

その他の Amazon Q CLI コマンド

Amazon Q には、設定管理以外の多くのコマンドライン機能が用意されています。使用可能なすべてのコマンドとその引数の包括的なリファレンスについては、「」を参照してくださいAmazon Q CLI コマンドリファレンス。

コマンドに関するヘルプについては、 `--help`フラグを使用します。

```
q [COMMAND] --help
```

Amazon Q CLI コマンドリファレンス

このセクションでは、すべての Amazon Q Developer CLI コマンドとその引数の包括的なリファレンスを提供します。このリファレンスを使用して、各コマンドで使用できるオプションとその適切な構文を理解します。

コマンドに関するヘルプについては、 `--help`フラグを使用できます。

```
q [COMMAND] --help
```

グローバル引数

次の引数は、Amazon Q CLI コマンドで使用できます。

グローバル引数

引数	ショートフォーム	説明
`--verbose`	`-v`	ログ記録の詳細度を高めます。より詳細な出力のために繰り返すことができます: `-v`、`-vv`、`-vvv`、 `-vvvv`
`--help`	`-h`	コマンドのヘルプ情報を表示する
`--version`	`-V`	バージョン情報を表示する
`--help-all`		すべてのサブコマンドのヘルプを印刷する

コマンド

以下のセクションでは、各 Amazon Q CLI コマンドとその使用可能な引数について説明します。

Q チャット

Amazon Q でインタラクティブチャットセッションを開始します。サブコマンドが指定されていない場合、 はqデフォルトで になりますq chat。

構文:

```
q chat [OPTIONS] [INPUT]
```

Q チャット引数

引数	ショートフォーム	説明
`--no-interactive`		インタラクティブモードなしで STDOUT への最初のレスポンスを出力します。これは、 も使用されていない限り、プロンプトがツールを使用するためのアクセス許可をリクエストすると失敗`--trust-all-tools` します。
`--resume`	`-r`	このディレクトリから前の会話を再開する
`--agent`		使用するエージェント
`--trust-all-tools`		確認を求めることなく、モデルが任意のツールを使用してコマンドを実行することを許可する
`--trust-tools`		指定されたツールセットのみを信頼します。ツール名のカンマ区切りリストを指定します。例えば、`--trust-tools=fs_read,fs_write` などです。ツールを信頼しない場合は、 を使用します。 `--trust-tools=`
`INPUT`		最初の質問 (位置引数)

例:

```
q chat
q chat "How do I list files in Linux?"
q chat --no-interactive --trust-all-tools "Show me the current directory"
q chat --resume
q chat --agent my-agent "Help me with AWS CLI"
```

Q 翻訳

AI を使用して自然言語の手順を実行可能シェルコマンドに変換します。

構文:

```
q translate [OPTIONS] [INPUT...]
```

Q 引数を変換する

引数	ショートフォーム	説明
--n	-n	生成する完了の数 (≤5 である必要があります)
INPUT		生成するコマンドの自然言語の説明 (位置引数)

例:

```
q translate "list all files in the current directory"
q translate "find all Python files modified in the last week"
q translate "compress all log files older than 30 days"
```

Q ドクター

Amazon Q の一般的なインストールと設定の問題を修正して診断します。

構文:

```
q doctor [OPTIONS]
```

Q 引数

引数	ショートフォーム	説明
--all	-a	修正なしですべての医師テストを実行する
--strict	-s	警告のエラー

例:

```
q doctor
q doctor --all
q doctor --strict
```

Q 更新

Amazon Q アプリケーションを最新バージョンに更新します。

構文:

```
q update [OPTIONS]
```

Q 更新引数

引数	ショートフォーム	説明
`--non-interactive`	`-y`	確認を求めない
`--relaunch-dashboard`		更新後にダッシュボードに再起動します。デフォルトは true
`--rollout`		ロールアウトを使用する

例:

```
q update
q update --non-interactive
q update --rollout
```

Q テーマ

オートコンプリートドロップダウンメニューのビジュアルテーマを取得または設定します。これは、コマンドの完了を示すポップアップウィンドウの外観に影響します。

構文:

```
q theme [OPTIONS] [THEME]
```

Q テーマ引数

引数	説明
`--list`	使用可能なすべてのテーマを一覧表示する

引数	説明
--folder	テーマディレクトリパスを表示する
THEME	設定するテーマの名前。組み込みテーマにはdark、、light、system (位置引数) が含まれます。

例:

```
q theme --list
q theme --folder
q theme dark
q theme light
q theme system
```

注： テーマの変更は、オートコンプリートポップアップウィンドウに影響します。視覚的な変更を確認するには、オートコンプリート提案をトリガーする必要がある場合があります。

Q 統合

Amazon Q のシステム統合を管理します。

構文:

```
q integrations [SUBCOMMAND] [OPTIONS]
```

Q 統合サブコマンド

Q 統合サブコマンド

サブコマンド	説明
install	統合をインストールします。--silent (-s) をサポートしてステータスメッセージを抑制
uninstall	統合をアンインストールします。--silent (-s) をサポートしてステータスメッセージを抑制
reinstall	統合を再インストールします。--silent (-s) をサポートしてステータスメッセージを抑制

サブコマンド	説明
status	統合のステータスを確認します。出力形式の --format (-f) をサポート

Q インライン

入力時にコマンドラインに直接表示されるインライン提案 (ゴーストテキスト) を管理します。詳細については、コマンドライン支援機能を参照してください。

構文:

```
q inline [SUBCOMMAND] [OPTIONS]
```

Q インラインサブコマンド

Q インラインサブコマンド

サブコマンド	説明
enable	入力時に表示されるインライン提案を有効にする
disable	インライン提案を無効にする
status	インライン提案が有効か無効かを表示する
set-customization	使用するカスタマイズモデルを選択します。必要に応じて、カスタマイズの ARN を指定します。
show-customizations	使用可能なカスタマイズモデルを表示します。出力形式の --format (-f) をサポート

例:

```
q inline enable
q inline disable
q inline status
q inline set-customization
q inline set-customization arn:aws:codewhisperer:us-east-1:123456789012:customization/
example
```

```
q inline show-customizations
q inline show-customizations --format json
```

Q ログイン

AWS Builder ID (無料) または IAM Identity Center (Pro) を使用して Amazon Q で認証します。

構文:

```
q login [OPTIONS]
```

Q ログイン引数

引数	説明
`--license`	ライセンスタイプ。オプション: `free` (ビルダー ID の場合)、`pro` (アイデンティティセンターの場合)
`--identity-provider`	ID プロバイダー URL (アイデンティティセンター用)
`--region`	AWS リージョン (アイデンティティセンター用)
`--use-device-flow`	認証には必ず OAuth デバイスフローを使用してください。ブラウザのリダイレクトを処理できない場合に便利です

例:

```
q login
q login --license free
q login --license pro --identity-provider https://my-company.awsapps.com/start --region
 us-east-1
q login --use-device-flow
```

Q ログアウト

Amazon Q セッションからサインアウトします。

構文:

```
q logout
```

このコマンドは追加の引数を取りません。

Q whoami

現在のユーザーと認証ステータスに関する情報を表示します。

構文:

```
q whoami [OPTIONS]
```

Q whoami 引数

引数	ショートフォーム	説明
`--format`	`-f`	出力形式。オプション: `plain` (デフォルト) `json`、、 `json-pretty`

Q プロファイル

現在の IAM Identity Center ユーザーに関連付けられているプロファイルを表示します。このコマンドは Pro ユーザーのみが使用できます。

構文:

```
q profile
```

このコマンドは追加の引数を取りません。

Q 設定

Amazon Q の設定を管理します。設定管理の詳細については、Amazon Q の設定「」を参照してください。

構文:

```
q settings [SUBCOMMAND] [OPTIONS] [KEY] [VALUE]
```

Q 設定引数

引数	ショートフォーム	説明
--delete	-d	設定を削除する
--format	-f	出力形式。オプション: plain (デフォルト) json、、 json-pretty
KEY		キーの設定 (位置引数)
VALUE		設定値 (位置引数)

Q 設定サブコマンド

Q 設定サブコマンド

サブコマンド	説明
open	デフォルトのエディタで設定ファイルを開く
all	現在の設定をすべて一覧表示します。--format オプションをサポート

Q 診断

診断テストを実行して、Amazon Q のインストールと設定の問題をトラブルシューティングします。

構文:

```
q diagnostic [OPTIONS]
```

Q 診断引数

引数	ショートフォーム	説明
--format	-f	出力形式。オプション: plain (デフォルト) json、、 json-pretty
--force		強制制限された診断出力

Q 問題

Amazon Q フィードバックまたはバグレポート用の新しい GitHub 問題を作成します。

構文:

```
q issue [OPTIONS] [DESCRIPTION...]
```

Q 問題引数

引数	ショートフォーム	説明
`--force`	`-f`	問題の強制作成
DESCRIPTION		問題の説明 (位置引数)

Q バージョン

バージョン情報を表示し、オプションで変更ログを表示します。

構文:

```
q version [OPTIONS]
```

Q バージョン引数

引数	説明
`--changelog[=VERSION]`	changelog を表示します。現在のバージョン、`--changelog=all` すべてのバージョン、または特定のバージョン`--changelog=x.x.x` で `--changelog` を使用する

Q MCP

モデルコンテキストプロトコル (MCP) サーバーを管理します。MCP の詳細については、Amazon Q Developer での MCP の使用「」を参照してください。

構文:

```
q mcp [SUBCOMMAND] [OPTIONS]
```

Q mcp 追加

設定された MCP サーバーを追加または置き換えます。

構文:

```
q mcp add [OPTIONS]
```

Q mcp 引数を追加する

引数	説明
`--name`	サーバーの名前 (必須)
`--command`	サーバーを起動するために使用されるコマンド (必須)
`--scope`	サーバーを追加する場所。オプション: `workspace` 、 `global`
`--env`	サーバーを起動するときに使用する環境変数。形式: `key1=value1,key2=value2`
`--timeout`	ミリ秒単位のサーバー起動タイムアウト
`--force`	既存のサーバーを同じ名前で上書きする

Q mcp 削除

MCP 設定からサーバーを削除します。

構文:

```
q mcp remove [OPTIONS]
```

Q mcp 引数を削除する

引数	説明
`--name`	削除するサーバーの名前 (必須)

引数	説明
--scope	削除するスコープ。オプション: workspace 、 global

Q mcp リスト

設定済みの MCP サーバーを一覧表示します。

構文:

```
q mcp list [SCOPE]
```

Q MCP リスト引数

引数	説明
SCOPE	一覧表示するスコープ。オプション: workspace 、 global (位置引数)

Q MCP インポート

別のファイルからサーバー設定をインポートします。

構文:

```
q mcp import [OPTIONS] [SCOPE]
```

Q MCP インポート引数

引数	説明
--file	サーバー設定をインポートするファイル (必須)
--force	既存のサーバーを同じ名前で上書きする
SCOPE	インポート先のスコープ。オプション: workspace 、 global (位置引数)

Q MCP ステータス

設定された MCP サーバーのステータスを取得します。

構文:

```
q mcp status [OPTIONS]
```

Q MCP ステータス引数

引数	説明
`--name`	ステータスを取得するサーバーの名前 (必須)

ログファイル

Amazon Q Developer CLI は、トラブルシューティングに役立つログファイルを保持します。これらのログはマシンにローカルに保存され、AWS には送信されません。

ログファイルの場所 :

- macOS: `$TMPDIR/qlog/`
- Linux: `$XDG_RUNTIME_DIR`または `TMPDIR`または `/tmp`

ログレベルは、 `Q_LOG_LEVEL`環境変数を設定することで制御できます。次の値を指定できます:

- `error`: エラーメッセージのみ (デフォルト)
- `warn`: 警告とエラーメッセージ
- `info`: 情報、警告、エラーメッセージ
- `debug`: デバッグ、情報、警告、エラーメッセージ
- `trace`: 詳細なトレース情報を含むすべてのメッセージ

> **Warning**
> ログファイルには、ファイルパス、コードスニペット、コマンド出力など、Amazon Q との会話ややり取りからの機密情報が含まれている場合があります。これらのログはローカルマ

シンにのみ保存され、AWS には送信されませんが、ログファイルを他のユーザーと共有する場合は注意が必要です。

ログレベルの設定例 (デバッグ目的):

```
# For bash/zsh
export Q_LOG_LEVEL=debug
q chat

# For fish
set -x Q_LOG_LEVEL debug
q chat
```

コマンドライン支援機能

Amazon Q CLI には、ターミナルでより効率的に作業できるように、2 つの異なる AI を活用した支援機能が用意されています。

- オートコンプリートドロップダウンメニュー: カーソルの右側に表示されるグラフィカルメニュー。矢印キーで選択できる使用可能なコマンドオプションが表示されます。
- インライン提案: 入力時にコマンドラインに直接表示される灰色の「ゴーストテキスト」

これらの機能は独立して動作します。どちらか一方をもう一方なしで使用でき、個別の設定オプションがあります。

オートコンプリートドロップダウンメニュー

オートコンプリートドロップダウンメニューは、コマンドを入力するときにカーソルの右側に表示され、矢印キーを使用して選択できる使用可能なオプション、サブコマンド、引数が表示されます。このグラフィカルメニューはテーマ化をサポートし、git、npm、docker、aws など、何百もの一般的なコマンドラインツールで動作します。

オートコンプリートドロップダウンの使用

オートコンプリートドロップダウンは、Amazon Q CLI のインストール後に自動的に有効になります。

オートコンプリートドロップダウンを使用するには

1. Amazon Q CLI をインストールします。
2. ターミナルまたはコマンドプロントを開きます。
3. コマンドの入力を開始すると、カーソルの右側に使用可能なオプションを示すグラフィカルメニューが表示されます。
4. 矢印キーを使用して提案をナビゲートし、Tab または Enter を押してオプションを選択します。

ドロップダウンメニューは数百のコマンドラインツールで動作するため、コマンドオプションと構文を覚えやすくなります。

オートコンプリートドロップダウンの設定

オートコンプリートドロップダウンの動作と外観をカスタマイズできます。

- 有効化/無効化: `q settings autocomplete.disable false` (有効化) または `q settings autocomplete.disable true` (無効化)
- 変更テーマ: `q theme dark`、`q theme light`、または `q theme system`
- 現在のテーマを表示します。 `q theme`
- 利用可能なテーマを一覧表示します。 `q theme --list`

オートコンプリート設定オプションの完全なリストについては、Amazon Q の設定「」を参照してください。

インライン提案

入力すると、インライン候補はコマンドラインに直接「ゴーストテキスト」として表示されます。この機能はオートコンプリートドロップダウンメニューとは別に、個別に有効または無効にできます。

インライン提案の使用

インライン提案は、インストール後にデフォルトで有効になります。コマンドを入力すると、グレーのゴーストテキストが表示され、完了の可能性が表示されます。

インライン提案を使用するには

1. ???をインストールします。

2. ターミナルまたはコマンドプロントを開きます。
3. コマンドの入力を開始すると、グレーのゴーストテキストが表示され、推奨される完了が表示されます。
4. 右矢印キーまたは Tab を押して提案を受け入れるか、入力を続けて無視します。

インライン提案の管理

`q inline` コマンドを使用してインライン提案を制御できます。

- 有効: `q inline enable`
- 無効: `q inline disable`
- ステータスを確認します。 `q inline status`
- カスタマイズの設定: `q inline set-customization [ARN]`
- 使用可能なカスタマイズを表示します。 `q inline show-customizations`

コマンドの完全なリファレンスについては、Q インライン「」を参照してください。

サポートされているツール

オートコンプリートドロップダウンメニューは、次のような幅広いコマンドラインツールをサポートしています。

- AWS CLI
- Git
- Docker
- npm
- kubectl
- terraform
- さらに多くの標準の Unix/Linux コマンド

コマンドライン用の Amazon Q への寄稿

フィードバックの提供、問題の報告、新機能の提案を行うことで、Amazon Q for コマンドラインの改善に役立てることができます。

GitHub リポジトリ

Amazon Q for コマンドラインはオープンソースプロジェクトです。ソースコードを検索し、GitHub でプロジェクトに貢献できます。

Amazon Q Developer CLI GitHub リポジトリにアクセスして、次の操作を行います。

- ソースコードを表示する
- 問題の報告
- プルリクエストを送信する
- ディスカッションに参加する

レポートに関する問題

Amazon Q for コマンドラインに関する問題は、コマンドラインから直接、または GitHub を通じて報告できます。

コマンドラインを使用して問題をレポートするには

1. ターミナルまたはコマンドプロントを開きます。
2. 次のコマンドを実行してください。

```
q issue
```

3. プロンプトに従って、発生している問題を記述します。
4. システム情報やログなど、レポートに含まれる情報を確認します。
5. 問題レポートを送信することを確認します。

問題レポートは、調査のために Amazon Q チームに送信されます。

GitHub で問題をレポートするには

1. GitHub リポジトリの「問題」ページにアクセスします。
2. 「新しい問題」をクリックします。
3. 問題テンプレートに、発生している問題の詳細を入力します。
4. 問題を送信します。

フィードバックの提供

Amazon Q のコマンドラインに関するフィードバックを提供して、製品の改善に役立てることができます。

フィードバックを提供するには

1. ターミナルまたはコマンドプロントを開きます。
2. 次のコマンドを実行してください。

```
q feedback
```

3. プロンプトに従ってフィードバックを提供します。

お客様のフィードバックは Amazon Q チームに送信され、製品の将来のバージョンを改善するために使用されます。

RFCs

RFC (コメントのリクエスト) プロセスを通じて、コマンドライン用の Amazon Q の新機能と改善点に関するディスカッションに参加できます。

GitHub リポジトリの「ディスカッション」ページにアクセスして、次の操作を行います。

- 既存の RFCs
- 提案された機能に関するコメント
- 新機能の独自の RFC を送信する

テレメトリデータ

Amazon Q for コマンドラインは、製品の改善に役立つテレメトリデータを収集します。このデータには、実行するコマンドやさまざまな機能を使用する頻度など、製品の使用方法に関する情報が含まれます。

テレメトリデータ収集はいつでもオプトアウトできます。

テレメトリデータ収集をオプトアウトするには

1. ターミナルまたはコマンドプロントを開きます。

2. 次のコマンドを実行してください。

```
q telemetry disable
```

テレメトリデータ収集を再度有効にするには、次のコマンドを使用します。

```
q telemetry enable
```

Amazon Q Developer での MCP の使用

Model Context Protocol (MCP) は、AI アシスタントが外部ツールやサービスとやり取りできるようにするオープンスタンダードです。Amazon Q Developer CLI で MCP がサポートされるようになりました。これにより、カスタムツールやサービスに接続して Q の機能を拡張できます。

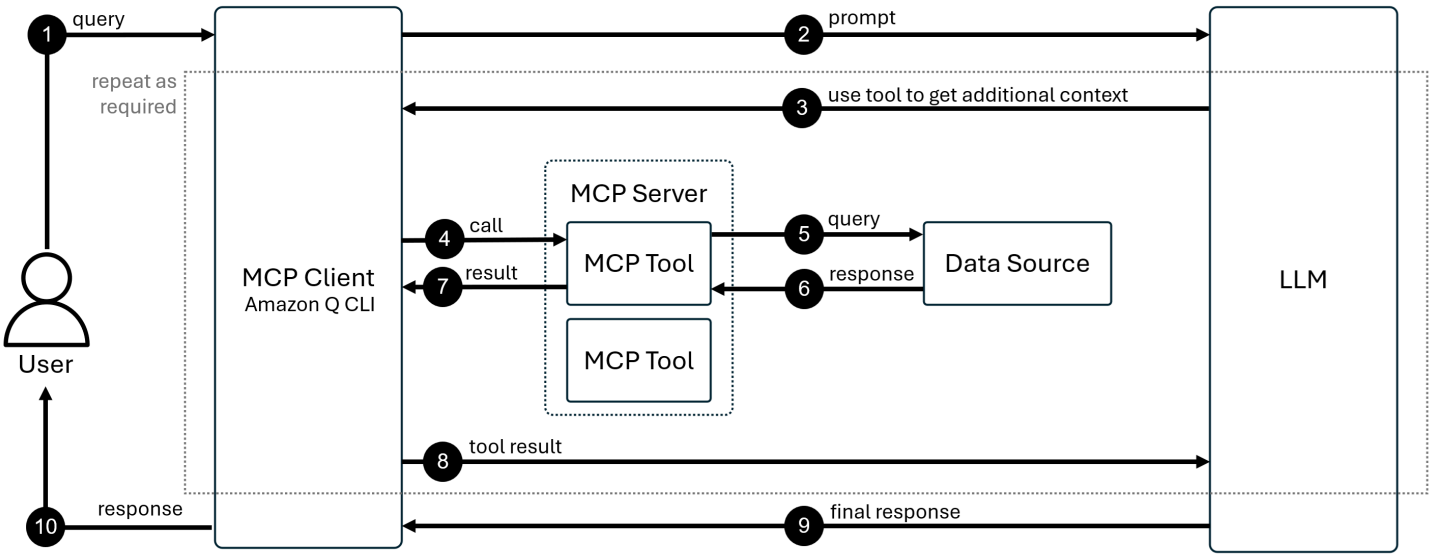
トピック

- MCP の概要
- CLI での MCP 設定
- MCP サーバーのロード
- ツールとプロンプト
- IDE での Q Developer の MCP 設定
- 主な利点
- MCP アーキテクチャ
- コア MCP の概念
- MCP セキュリティ

MCP の概要

MCP (モデルコンテキストプロトコル) は、AI アシスタントが外部ツールと通信する方法を標準化するオープンプロトコルです。AI モデルが利用可能なツールを検出し、特定のパラメータを使用してツールの実行をリクエストし、ツールの結果を受信して処理するための構造化された方法を定義します。

MCP は AI モデル用のユニバーサルコネクタであり、外部システムとやり取りしたり、ライブデータを取得したり、さまざまなツールとシームレスに統合したりできます。これにより、Amazon Q は必要な情報にリアルタイムでアクセスすることで、よりコンテキストに関連する支援を提供できます。



CLI での MCP 設定

このページでは、MCP サーバーを設定するための CLI 固有のオプションについて説明します。

設定コマンド

使用法: `qchat mcp [OPTIONS]` *`COMMAND`*

MCP 設定コマンド

コマンド	説明
`qchat mcp add`	設定済みのサーバーを追加または置き換える
`qchat mcp remove`	MCP 設定からサーバーを削除する
`qchat mcp list`	設定済みサーバーを一覧表示する
`qchat mcp import`	別のファイルからサーバー設定をインポートする
`qchat mcp status`	設定済みサーバーのステータスを取得する
`qchat mcp help`	このコマンドのリストまたは特定のサブコマンドのヘルプを出力する (複数可)

MCP サーバーのロード

Amazon Q は MCP サーバーをバックグラウンドでロードするため、すべてのサーバーの初期化を待たずにすぐに操作を開始できます。各サーバーのロードが完了すると、ツールは徐々に利用可能になります。

サーバーステータスの確認

`/tools` コマンドを使用して、ロード中のサーバーと使用可能なツールを確認できます。

サーバー初期化の設定

以下を使用して、サーバーの初期化タイムアウトをカスタマイズできます。

```
$ q settings mcp.initTimeout [value]
```

ここで、`[value]`はミリ秒単位のタイムアウトです。この設定は、Amazon Q がサーバーの初期化を待機してからインタラクションを開始するまでの時間を制御します。

ツールとプロンプト

このセクションでは、Amazon Q Developer CLI で MCP ツールとプロンプトを使用する方法について説明します。

MCP ツールについて

MCP ツールは、MCP サーバーが Amazon Q Developer CLI に公開する実行可能関数です。これにより、Amazon Q Developer はユーザーに代わってアクションを実行し、データを処理して、外部システムとやり取りできます。

MCP の各ツールには、次のものがあります。

- 名前: ツールの一意の識別子
- 説明: ツールの動作に関する人間が読める説明
- 入力スキーマ: ツールが受け入れるパラメータを定義する JSON スキーマ
- 注釈: ツールの動作と効果に関するオプションのヒント

使用可能なツールの検出

Q CLI セッションで使用できるツールを確認するには：

```
/tools
```

このコマンドは、組み込みツールと MCP サーバーが提供するツールの両方を含む、使用可能なすべてのツールを表示します。

ツールには、使用方法を決定するさまざまなアクセス許可レベルがあります。

- 自動承認: これらのツールは、呼び出しごとに明示的なアクセス許可なしで使用できます
- 承認が必要: これらのツールを使用するたびに明示的なアクセス許可が必要です
- 危険: これらのツールは潜在的にリスクがあり、承認前に慎重に検討する必要があります

ツールの使用方法

MCP ツールは、次の 2 つの方法で使用できます。

1. 自然言語リクエスト: 何をするかを説明するだけで、Q が使用するツールを決定します。
2. 直接ツール呼び出し: 特定のツールを使用するように Q に明示的にリクエストすることもできます。

プロンプトの使用

MCP サーバーは、特定のタスクで Q をガイドするのに役立つ事前定義されたプロンプトを提供できます。

- 使用可能なプロンプトを一覧表示します。 `/prompts`
- プロンプトを使用します。
  - `@ prompt-name arg1 arg2`

引数でプロンプトを使用する例：

```
@fetch https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/command-line-mcp-configuration.html
```

IDE での Q Developer の MCP 設定

このページでは、MCP サーバーを設定するための IDE 固有のオプションについて説明します。

MCP 設定 UI へのアクセス

次の図は、IDE で Q Developer 用の MCP サーバーを設定するために使用される GUI を示しています。

IDE の Q Developer で MCP 設定 UI にアクセスするには：

1. IDE (VS Code、JetBrains など) を開きます。
2. Q Developer パネルを開きます。
3. チャットパネルを開きます。

4. ツールアイコンを選択します。

MCP サーバーの追加

MCP サーバーを IDE に追加するには：

1. MCP 設定 UI にアクセスします。
2. プラス記号 (+) を選択します。
3. グローバルまたはローカルのスコープを選択します。

   グローバルスコープを選択した場合、MCP サーバー設定は ~/.aws/amazonq/mcp.json に保存され、すべてのプロジェクトで使用できます。ローカルスコープを選択した場合、設定は現在のプロジェクト内の .amazonq/mcp.json に保存されます。
4. Name フィールドに、MCP サーバーの名前を入力します。

   たとえば、AWS Documentation MCP サーバーをインストールしていた場合、名前は *AWS DocMCPServer* である可能性があります。
5. トランスポートプロトコルを選択します。

   現在は、`stdio` のみがサポートされます。
6. コマンドフィールドに、初期化時に MCP サーバーが実行するシェルコマンドを入力します。

   AWS ドキュメント MCP サーバーの場合、コマンドは ですuvx。これは のエイリアスでありuv `tool run`、一時的な Python 環境を作成します。
7. Arguments フィールドに、必要に応じてシェルコマンドに渡す引数を入力します。

   AWS ドキュメント MCP サーバーの場合、引数は *awslabs.aws-documentation-mcp-server@latest* です。これは、PyPI (Python Package Index) でホストされているパッケージを指す Python パッケージ識別子です。

   必要に応じて引数を追加します。
8. 必要に応じて環境変数を入力します。

   この例では、最初に名前: *FASTMCP_LOG_LEVEL* と値: *ERROR* を入力します。

また、*AWS_DOCUMENTATION_PARTITION* という名前と *aws* 値を使用して、作業するパーティションを示します。

9. 必要に応じてタイムアウト値を入力します。

   この例では、推奨値を 60 (秒) に維持します。

10. [保存] を選択します。

    設定パネルはツールのアクセス許可パネルに置き換えられます。

11. ツールのアクセス許可の確認と調整 「」の手順に従います。

MCP 設定のトラブルシューティング

IDE に MCP サーバーを追加すると、Amazon Q はそのサーバーへの接続を試みます。

接続に問題がある場合、パネルの上部にアラートが表示されます。アラートが解決されるまで、その MCP サーバーからのツールが正常に機能することは想定しないでください。

設定の修正を選択して MCP 設定画面に戻って、適切な変更を行うことができます。

MCP サーバーの有効化

次の手順では、問題の MCP サーバーがまだ有効になっていないことを前提としています。

IDE で MCP サーバーを有効にするには：

1. MCP サーバーパネルを開きます。
2. 有効にするサーバーの横にある、有効化 を選択します。

MCP サーバーの無効化

IDE で MCP サーバーを無効にするには：

1. MCP サーバーパネルを開きます。
2. 無効にするサーバーを選択します。
3. セットアップの編集の横にある 3 つのドットを選択します。
4. MCP サーバーを無効にするを選択します。

現在有効になっている MCP サーバーの削除

現在有効になっている MCP サーバーを IDE から削除するには：

1. MCP サーバーパネルを開きます。
2. 削除するサーバーを選択します。

   パネルが開き、そのサーバーの詳細が表示されます。
3. セットアップの編集の横にある 3 つのドットを選択します。
4. MCP サーバーの削除を選択します。
5. プロンプトが表示されたら、削除を確認します。

現在無効になっている MCP サーバーの削除

現在無効になっている MCP サーバーを IDE から削除するには：

1. MCP サーバーパネルを開きます。
2. 削除するサーバーの横にある「削除」を選択します。
3. プロンプトが表示されたら、削除を確認します。

ツールのアクセス許可の確認と調整

ツールのアクセス許可を確認して調整するには：

1. MCP サーバーパネルを開きます。
2. アクセス許可を確認して調整する MCP サーバーを選択します。
3. ツールごとに、次のいずれかのアクセス許可レベルを設定できます。

   - 質問: ツールを使用するたびにアクセス許可を求めます。
   - 常に許可する: プロンプトを表示せずにツールを実行できるようにします。
   - 拒否: このツールを使用しないでください。

主な利点

- 拡張性: Amazon Q を特定のドメインまたはワークフロー専用のツールに接続する

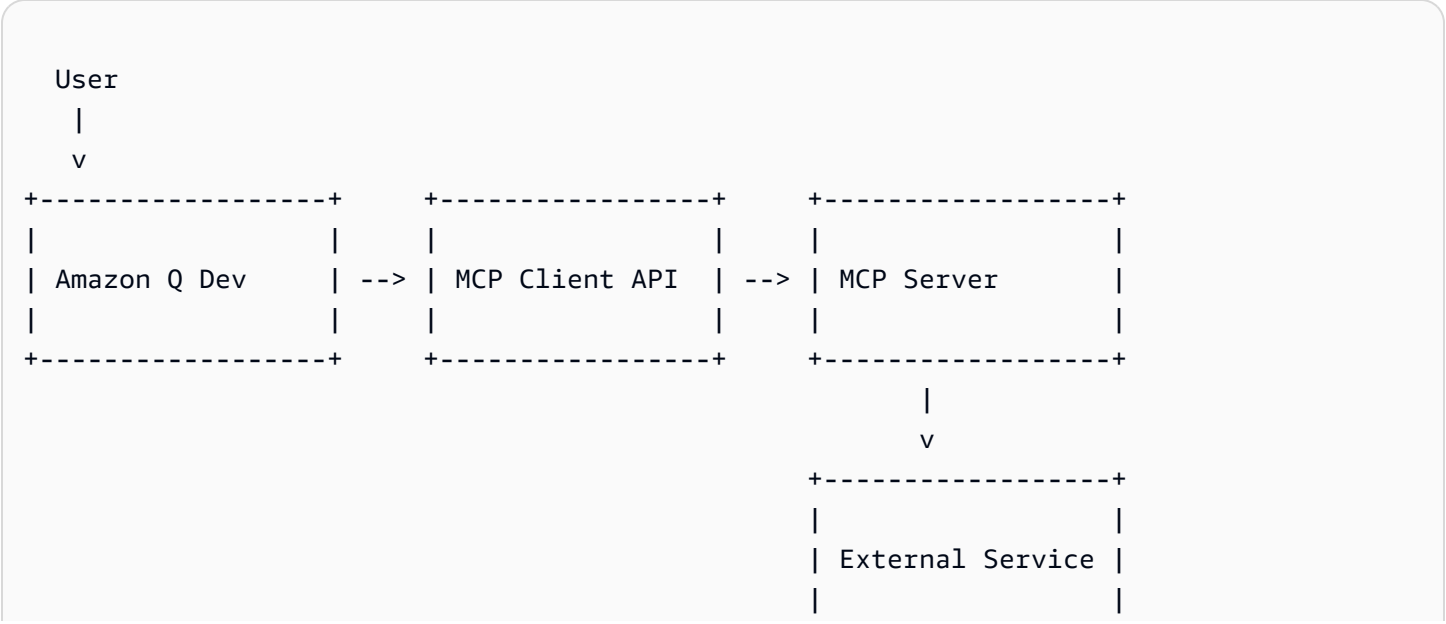
- カスタマイズ: 特定のニーズに合わせてカスタムツールを作成する
- エコシステムの統合: MCP 互換ツールの増大するエコシステムを活用する
- 標準化: 複数の AI アシスタントでサポートされている一貫したプロトコルを使用する
- 柔軟性: MCP では、同じツール統合を維持しながら、異なる LLM プロバイダーを切り替えることができます。
- セキュリティ: ローカル MCP サーバーを使用してデータをインフラストラクチャ内に保持する

MCP アーキテクチャ

MCP は、クライアント/サーバーアーキテクチャに従います。

- MCP ホスト: MCP を介してデータにアクセスする Amazon Q Developer CLI などのプログラム
- MCP クライアント: サーバーとの 1:1 接続を維持するプロトコルクライアント
- MCP サーバー: 標準化されたモデルコンテキストプロトコルを通じて、それぞれが特定の機能を公開する軽量プログラム
- ローカルデータソース: MCP サーバーが安全にアクセスできるコンピュータのファイル、データベース、サービス
- リモートサービス: MCP サーバーが接続できるインターネット (APIs など) 経由で利用可能な外部システム

Example MCP 通信フロー



```
                                                    +------------------+
```

<caption>

MCP を介したユーザー、Amazon Q Developer CLI、外部サービス間の通信フロー
</caption>

コア MCP の概念

ツール

ツールとは、MCP サーバーがクライアントに公開する実行可能関数です。これにより、Amazon Q は次のことを実行できます。

- 外部システムでアクションを実行する
- 特殊な方法でデータを処理する
- APIsとサービスとやり取りする
- ユーザーに代わってコマンドを実行する

ツールは、一意の名前、説明、入力スキーマ (JSON スキーマを使用) 、およびツールの動作に関するオプションの注釈で定義されます。

プロンプト

プロンプトは、特定のタスクで Amazon Q をガイドするのに役立つ事前定義されたテンプレートです。次の操作が可能です。

- 動的引数を受け入れる
- リソースからコンテキストを含める
- 複数のインタラクションを連鎖する
- 特定のワークフローのガイド
- UI 要素として表面化 (スラッシュコマンドなど)

リソース

リソースは、次のような MCP サーバーが Amazon Q に提供できるデータを表します。

- ファイルの内容
- データベースレコード
- API レスポンス
- ドキュメント
- 設定データ

MCP セキュリティ

Amazon Q Developer CLI で MCP サーバーを使用する場合は、セキュリティへの影響とベストプラクティスを理解することが重要です。

セキュリティモデル

Amazon Q Developer CLI の MCP セキュリティモデルは、次の原則に基づいて設計されています。

1. 明示的なアクセス許可: ツールを実行する前に明示的なユーザーアクセス許可が必要です
2. ローカル実行: MCP サーバーはマシン上でローカルに実行されます
3. 分離: 各 MCP サーバーは個別のプロセスとして実行されます
4. 透明性: ユーザーは利用可能なツールとその動作を確認できます。

セキュリティに関する考慮事項

MCP を使用する際の主なセキュリティ上の考慮事項：

- 信頼できるソースからのみサーバーをインストールする
- 承認する前にツールの説明と注釈を確認する
- 機密設定に環境変数を使用する
- MCP サーバーと Q CLI を最新の状態に保つ
- 予期しないアクティビティがないか MCP ログをモニタリングする

Amazon Q Developer とのサードパーティー統合

Amazon Q Developer は一般的な開発プラットフォームと統合され、特殊な人工知能 (AI) 機能を通じてソフトウェア開発ワークフローを強化します。サポートされる統合には、 および GitLab Duoが含まれGitHub、開発ライフサイクルを通じて AI を活用したサポートを提供します。これらの統合は、日常的なタスクを自動化し、コード品質を向上させ、モダナイゼーションの取り組みを加速することで、開発を合理化するのに役立ちます。

GitLab Duo Amazon Q Developer を使用する

GitLab Duo と Amazon Q Developer は、GitLab ワークフローに直接統合された包括的な AI エクスペリエンススイートを提供します。GitLab 自己管理型サービスおよび Ultimate 階層のサブスクライバーが利用できる統合により、GitLab の問題に対するクイックアクションと AI 機能をトリガーするためのマージリクエストが可能になります。統合には、Amazon Q を搭載した GitLab Duo Chat も含まれており、開発プロセス全体でコンテキストに応じたサポートを提供します。

GitLab Duo Amazon Q の では、以下が提供されます。

- GitLab の問題での迅速なアクションによる高レベルのアイデアの特徴量開発
- マージリクエストでの迅速なアクションによるコード品質、問題、セキュリティ上の懸念に関するコードレビュー
- GitLab の問題でクイックアクションを使用して、サポートされている最新の Java バージョンへのコードモダナイゼーション
- マージリクエストでクイックアクションを使用したユニットテスト生成
- 開発タスクの統合チャットサポート

開始するには、「Amazon Q GitLab Duoでのセットアップ」を参照してください。

Amazon Q Developer for GitHub (プレビュー)

Amazon Q Developer と の統合GitHubにより、特殊な AI エージェントによる自動機能開発、コードレビュー、Java モダナイゼーションが可能になります。Amazon Q Developer GitHubに問題を割り当てると、問題とプロジェクトコードをコンテキストとして使用して新しいコードを生成し、プルリクエストを作成します。開発プロセス中にフィードバックを提供すると、Amazon Q Developer は提案されたコードを繰り返して、共同開発ワークフローを作成できます。

Amazon Q Developer は、 で次の主要な機能を提供しますGitHub。

- アイデアからプルリクエストまで、新機能とバグ修正を自動的に実装する機能開発ラベル
- コードの品質、問題、セキュリティ上の懸念に関する新規または再開されたプルリクエストの自動コードレビュー
- コードベースをサポートされている Java バージョンに自動的にアップグレードするコード変換ラベル
- コマンドをスラッシュして、問題から機能開発とコード変換を開始し、最初の自動レビュー後にコードレビューを開始するための代替方法を提供します。
- 生成されたコードに関するフィードバックを提供し、実装することで開発を反復する
- 機能開発タスクとコード変換タスクを Amazon Q Developer にすばやく割り当てるためのブラウザ拡張機能

開始するには、「クイックスタート: のインストール、 の機能の使用GitHub、使用制限の引き上げ」を参照してください。

Amazon Q Developer のプロジェクトルール

Amazon Q Developer では、GitLab または でプロジェクト固有のルールを作成および維持できます。これによりGitHub、チームのコーディング標準とベストプラクティスが定義されます (Python コードで型ヒントが必要な場合や、Java コードで Javadoc コメントが必要な場合など)。これらのルールは、 *project-root*`/.amazonq/rules` ディレクトリに Markdown ファイルとして保存され、経験レベルに関係なくすべてのデベロッパー間で一貫性が確保され、デベロッパーがプロジェクト内で Amazon Q Developer とやり取りするときに Amazon Q Developer のコンテキストに自動的に組み込まれ、生成されたすべてのレスポンスが確立されたガイドラインに従っていることを確認します。詳細については、「サードパーティープラットフォームでの Amazon Q Developer のプロジェクトルールの作成」を参照してください。

GitLab Duo Amazon Q を使用した

GitLab Duo Amazon Q では、アイデアのためのコード実装の提案、コードを Java 17 に変換する、品質と脆弱性に関するマージリクエストを確認する、ユニットテストを提案するなど、人工知能 (AI) エクスペリエンスのスイートを提供します。さらに、Amazon Q をサポートするGitLab Duoチャット機能を使用して、脆弱性の説明、失敗したパイプラインのトラブルシューティング、コードのリファクタリングなどの開発タスクに対処できます。GitLab セルフマネージド型サービスおよび

Ultimate 階層サブスクリプション (GitLab デュオと Amazon Q サブスクリプションアドオン) で利用できます。詳細については、GitLab ドキュメントの GitLab プランを参照してください。 GitLab

Amazon Q GitLab Duoで を設定したら、GitLab の問題でクイックアクションを使用し、リクエストコメントをマージして AI 機能をトリガーできます。詳細については、「GitLab Duo の概念」および「Amazon Q GitLab Duoの開始方法」を参照してください。Amazon Q GitLab Duoで で使用できるすべての機能については、GitLab ドキュメントの「サポートされているその他の機能」を参照してください。

トピック

- GitLab Duo の概念
- Amazon Q GitLab Duoの開始方法
- コード変換用の CI/CD パイプラインのカスタマイズ
- Amazon Q GitLab Duoでの の問題のトラブルシューティング

GitLab Duo の概念

GitLab Duo Amazon Q で を使用する際に知っておくべき概念と用語をいくつか紹介します。

トピック

- Amazon Q GitLab Duoを使用した の設定
- AWS ID プロバイダーによるオンボーディングと IAM ロールの作成
- GitLab クイックアクション

Amazon Q GitLab Duoを使用した の設定

GitLab で Amazon Q 人工知能 (AI) 機能を使用するには、まず以下を実行する必要があります。

- GitLab 17.11.0 以降のセルフマネージドインスタンスがある。
- Amazon Q で GitLab Ultimate サブスクリプションを持っている (トライアルアクセスなし) 。
- GitLab Duo 機能を有効にします (実験機能とベータ機能はデフォルトでオフになっています) 。詳細については、「ベータおよび実験機能を有効にする」を参照してください。
- GitLab 用の IAM ID プロバイダーを作成します。詳細については、GitLab ドキュメントの「IAM ID プロバイダーを作成する」を参照してください。

- IAM ID プロバイダーが GitLab で Amazon Q にアクセスできると信頼する IAM ロールを作成します。これには、特定のアクセス許可を使用する必要があります。詳細については、GitLab ドキュメントの「IAM ロールの作成」を参照してください。

AWS ID プロバイダーによるオンボーディングと IAM ロールの作成

GitLab Duo オンボーディングプロセスの一環として、Amazon Q Developer コンソールを使用して Amazon Q Developer プロファイルを作成する必要があります。プロファイルを使用すると、ID プロバイダーのユーザー全体またはサブセットのカスタマイズおよび制御設定を作成できます。プロファイルを作成したら、OpenID Connect (OIDC) ID プロバイダー (IdP) と IAM サービスロールを使用して、 GitLab Duoと AWS アカウント間の信頼を確立する必要があります。必要なリソースを作成し、Amazon Q GitLab Duoでセットアップする方法については、GitLab ドキュメントの「Amazon Q GitLab Duoでセットアップする」を参照してください。

新しい IAM ロールが作成されると、必要なアクセス許可を持つ必要な信頼ポリシーも作成されます。ロール信頼ポリシーは、IAMのロールに関連付けられている必須のリソースベースのポリシーです。

アクセス許可ポリシーを追加する必要があります。これにより、Amazon Q に接続し、Amazon Q との統合GitLab Duoで の機能を利用できるようになります。IAM ロールを作成するときは、ポリシーを追加する必要があります。アクセス許可ポリシーによって提供されるアクセス許可の詳細については、「」を参照してくださいGitLabDuoWithAmazonQPermissionsPolicy。

または、インラインポリシーを作成し、必要なアクセス許可を追加することもできます。アクセスコントロールをカスタマイズする場合は、インラインポリシーの作成を選択できます。詳細については、IAM ユーザーガイドの「管理ポリシーとインラインポリシー」およびAWS Identity and Access Managementアクセス許可」を参照してください。

信頼ポリシー

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Principal": {
```

```
                "Federated": "arn:aws:iam::111122223333:oidc-provider/
auth.token.gitlab.com/cc/oidc/{{Instance_ID}}"
            },
            "Condition": {
                "StringEquals": {
                    "auth.token.gitlab.com/cc/oidc/{{Instance_ID}}:aud": "gitlab-
cc-{{Instance_ID}}"
                }
            }
        }
    ]
}
```

アクセス許可ポリシー

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GitLabDuoUsagePermissions",
      "Effect": "Allow",
      "Action": [
        "q:SendEvent",
        "q:CreateAuthGrant",
        "q:UpdateAuthGrant",
        "q:GenerateCodeRecommendations",
        "q:SendMessage",
        "q:ListPlugins",
        "q:VerifyOAuthAppConnection"
      ],
      "Resource": "*"
    },
    {
      "Sid": "GitLabDuoManagementPermissions",
      "Effect": "Allow",
      "Action": [
        "q:CreateOAuthAppConnection",
        "q:DeleteOAuthAppConnection"
      ],
      "Resource": "*"
```

```
    },
    {
      "Sid": "GitLabDuoPluginPermissions",
      "Effect": "Allow",
      "Action": [
        "q:CreatePlugin",
        "q:DeletePlugin",
        "q:GetPlugin"
      ],
      "Resource": "arn:aws:qdeveloper:*:*:plugin/GitLabDuoWithAmazonQ/*"
    }
  ]
}
```

オプションで、キーのライフサイクルと使用状況を完全に制御したい場合は、カスタマーマネージドキー (CMK) を使用してリソースを暗号化することもできます。コンテンツの暗号化と復号に CMK を使用できるユーザーを制限するkms:ViaService条件キー。詳細については、「サードパーティー統合のための Amazon Q Developer へのアクセスを管理する」を参照してください。

GitLab クイックアクション

呼び出されると、クイックアクションは GitLab の問題でタスクを実行し、リクエストをマージします。GitLab でクイックアクションを呼び出す方法については、GitLab ドキュメントを参照してください。

リクエストの生成と反復のマージ

- /q dev – GitLab の問題でキャプチャされた大まかなアイデアから、提案されたコード実装で Amazon Q ready-to-reviewマージリクエストを生成させることができます。これにより、概念を作業コードに変換するプロセスを合理化できます。マージリクエストは新しいブランチに作成され、Amazon Q は問題作成者をマージリクエストレビューワーとして割り当てます。また、マージリクエストの概要も表示されます。詳細については、「アイデアをマージリクエストに変換する」を参照してください。
- /q dev (改訂) – 問題からやり直すのではなく、Amazon Q が提供する提案されたコード実装を繰り返し実行できます。Amazon Q はフィードバックを確認し、最初に生成されたコードを更新します。また、変更が行われるたびにコミットメッセージも提供されます。各イテレーション後の説明が更新され、フィードバックを説明するコメントがイテレーションに組み込まれます。その後、提案を確認してコードにマージできます。詳細については、「フィードバックに基づいてコードを変更する」を参照してください。

コード変換

- `/q transform` – Java Maven 8 または Java Maven 11 から Java Maven 17 プロジェクトへのアップグレードプロセスを開始できます。GitLab の問題から、Amazon Q はコードを分析して必要な Java のアップグレードまたはモダナイゼーションを決定し、問題を更新し、提案された変更で新しいマージリクエストを自動的に開き、問題作成者をレビューワーとして割り当てます。構築するには GitLab Runner のセットアップが必要であり、コード変換用にカスタマイズする必要があります。詳細については、コード変換用の CI/CD パイプラインのカスタマイズ「Java のアップグレード」を参照してください。

  > **Note**
  >
  > Maven プロジェクトのソースバージョンは、コードを変換する前に特定する必要があるため、コンパイラの設定は `pom.xml` ファイル内で設定する必要があります。したがって、`pom.xml`ファイルにはソースとターゲットが必要です。

ユニットテスト生成

- `/q test` – マージリクエストで新しく追加されたソースコード行のユニットテストを生成できます。テストファイルに追加できるユニットテスト提案を含む Amazon Q コメント。生成されたテストを一度に適用することも、適用する前に各テストを個別に確認することもできます。マージリクエストにテストファイルが見つからない場合、Amazon Q はテストファイルに手動で追加できるユニットテストを提供します。詳細については、「テストカバレッジの作成」を参照してください。

コードレビュー

- `/q review` – Amazon Q GitLab Duoを使用して でマージリクエストのレビューを開始できます。新しいマージリクエストに対して自動コードレビューが開始されます。GitLab 管理者は、自動レビューを無効にするように Amazon Q を設定することもできます。自動コードレビューは、Amazon Q がマージリクエストを生成してコード修正を提案する際に発生する可能性のある問題を特定して修正します。品質チェック、問題の分析、論理エラー、アンチパターン、コードの重複などを提供します。

  Amazon Q は、コメントを含むコード分析を提供し、各コメントは個別の検出結果を提供します。このクイックアクションは、すべての言語で使用できます。自動コードレビューは、新しいマージリクエストを開くか、以前に閉じたマージリクエストを再度開くと開始されます。ただし、自動

コードレビューは、既存のマージリクエスト内で行われた後続のコミットによってトリガーされません。`/q review` クイックアクションを使用して、コードレビューを手動でトリガーできます。

GitLab インスタンスまたはグループ内の新しいマージリクエストごとに自動的に実行されるようにコードレビューを設定できます。詳細については、「マージリクエストの確認」を参照してください。

ウェブ UI と IDEs でのチャットセッション

- GitLab デュオチャットとコード提案は Amazon Q と連携して、CI/CD 設定、エラーの説明、質問への対処をサポートします。チャットセッションでスラッシュコマンドを使用して、Amazon Q チャット機能で GitLab デュオ を呼び出すことができます。詳細については、GitLab デュオチャットをリクエストする」を参照してください。

Amazon Q GitLab Duoの開始方法

GitLab Duo Amazon Q を使用すると、人工知能 (AI) 機能がソフトウェア開発オペレーションとソースコード管理ワークフローに直接組み込まれます。セルフマネージド GitLab インスタンスと GitLab と同期された GitLab Ultimate サブスクリプションを使用して、GitLab DuoAmazon Q の使用を開始できます。 GitLab Amazon Q Developer プロファイルを作成し、OpenID Connect (OIDC) ID プロバイダーとの接続を追加し、GitLab から Amazon Q にアクセスするための IAM ロールを作成する必要があります。詳細については、「AWS ID プロバイダーによるオンボーディングと IAM ロールの作成」を参照してください。必要なリソースを作成し、Amazon Q GitLab Duoでセットアップする方法については、GitLab ドキュメントの「Amazon Q GitLab Duoでセットアップする」を参照してください。

インスタンス、グループ、またはプロジェクトの をオンまたはオフにすることで、Amazon Q GitLab Duoで の可用性をいつでも設定できます。詳細については、「Amazon Q GitLab Duoでオフにする」を参照してください。

Amazon Q GitLab Duoをセットアップしたら、GitLab で Amazon Q の AI 機能を使用してコードを Java 17 に変換し、マージリクエストの品質と脆弱性を確認し、ユニットテストを提案できます。Amazon Q をサポートするGitLab Duoチャット機能を使用して、脆弱性の説明、失敗したパイプラインのトラブルシューティング、コードリファクタリングなどの開発タスクに対応することもできます。

コード変換を呼び出す前に、プロジェクトで少なくとも 1 つの GitLab Runner をカスタマイズする必要があります。詳細については、「コード変換用の CI/CD パイプラインのカスタマイズ」を参照

してください。GitLab の問題でクイックアクションを呼び出し、リクエストをマージする方法の詳細については、GitLab DuoAmazon Q の「」を参照してください。

コード変換用の CI/CD パイプラインのカスタマイズ

Amazon Q for コード変換は、静的分析を使用してその機能の一部を実行します。そのためには、プロジェクトのソースコードに加えてコンパイルおよびテストスコープの依存関係を指定する必要があります。GitLab のコード変換では、GitLab CI/CD ジョブを使用してこれらの依存関係へのアクセスを提供します。

プロジェクトのコード変換を呼び出す前に、以下が必要です。

- 少なくとも 1 つの GitLab Runner。
- プロジェクトで CI/CD 機能を有効にする必要があります。
- プロジェクトのデフォルトブランチに.gitlab-ci.ymlコミットされた 。

コード変換用に CI/CD パイプラインをカスタマイズするには

1. プロジェクトに GitLab CI/CD パイプラインがまだない場合は、GitLab が提供するMaven.gitlab-ci.ymlテンプレートを使用して作成します。詳細については、「プロジェクトパイプラインを作成する」を参照してください。
2. 次のジョブで .gitlab-ci.yml ファイルを更新します。

```
q-code-transformation:
  stage: build
  script:
    - 'mvn $MAVEN_CLI_OPTS test-compile'
    - 'mvn $MAVEN_CLI_OPTS dependency:copy-dependencies -
DoutputDirectory=dependencies -Dmdep.useRepositoryLayout=true -Dmdep.copyPom=true -
Dmdep.addParentPoms=true'
  artifacts:
    name: q-code-transformation-dependencies
    paths:
      - dependencies/*
  rules:
    - if: $CI_COMMIT_REF_NAME =~ /^q\/transform-/ && $CI_PIPELINE_SOURCE == 'push'
      when: always
```

- 最初のmvn呼び出しでは、Amazon Q コード変換が処理を試みる前にプロジェクトがコンパイルされていることを検証します。目標は、テストコンパイル、テスト、統合テスト、検証のいずれかです。
- 2 番目のmvn呼び出しでは、プロジェクトの依存関係をステージングディレクトリにコピーして、ジョブアーティファクトとして含めます。
- このセクションでは、Amazon Q コード変換によってアクセスできるように、コピーされた依存関係artifactsをアップロードします。
- このセクションでは、新しいコミットがプッシュされたq/transform-*ときに で始まるブランチ名でのみ実行されるようにこのジョブrulesを設定します。これは、マージリクエストが開かれた場合には当てはまりません。

Java バージョンを動的に選択する

完了後、Amazon Q コード変換が GitLab でマージリクエストを開くと、プロジェクトパイプラインはマージリクエストを実行するように設定されたジョブを実行します。更新されたコードは Java 17 を対象とするため、ジョブが Java 8 または Java 11 を使用してビルドしようとすると、これらのジョブでビルドエラーが発生します。

以下は、Docker .gitlab-ci.ymlを使用し、 で始まるブランチ名を持つマージリクエストでジョブが実行されているときに Java 17 を動的に選択する高度な ですq/transform-*。開いたリクエストをデフォルトブランチにマージすることを決定したら、デフォルトで Java 17 を使用する.gitlab-ci.ymlように を変更する必要があります。

```
variables:
  MAVEN_OPTS: >-
    -Dhttps.protocols=TLSv1.2
    -Dmaven.repo.local=$CI_PROJECT_DIR/.m2/repository
    -Dorg.slf4j.simpleLogger.showDateTime=true
    -Djava.awt.headless=true
    -Dmaven.install.skip=true

  MAVEN_CLI_OPTS: >-
    --batch-mode
    --errors
    --fail-at-end
    --show-version
    --no-transfer-progress
    -DinstallAtEnd=true
```

```
    -DdeployAtEnd=true

  BUILD_IMAGE: maven:3-openjdk-8

workflow:
  rules:
    - if: $CI_COMMIT_REF_NAME =~ /^q\/transform-/ && $CI_PIPELINE_SOURCE ==
 'merge_request_event'
      variables:
        BUILD_IMAGE: maven:3-openjdk-17
    - when: always

image: $BUILD_IMAGE

cache:
  paths:
    - .m2/repository

compile:
  stage: build
  script:
    - 'mvn $MAVEN_CLI_OPTS compile'

verify:
  stage: test
  script:
    - 'mvn $MAVEN_CLI_OPTS verify'

q-code-transformation:
  stage: build
  script:
    - 'mvn $MAVEN_CLI_OPTS test-compile'
    - 'mvn $MAVEN_CLI_OPTS dependency:copy-dependencies -DoutputDirectory=dependencies
 -Dmdep.useRepositoryLayout=true -Dmdep.copyPom=true -Dmdep.addParentPoms=true'
  artifacts:
    name: q-code-transformation-dependencies
    paths:
      - dependencies/*
  rules:
    - if: $CI_COMMIT_REF_NAME =~ /^q\/transform-/ && $CI_PIPELINE_SOURCE == 'push'
      when: always
```

Amazon Q GitLab Duoでの の問題のトラブルシューティング

Amazon Q GitLab Duoで を使用する場合の一般的なオンボーディングの問題をトラブルシューティングするには、次のセクションを参照してください。

GitLab インスタンスがインバウンドリクエストをブロックする

問題: GitLab インスタンスがインバウンドリクエストをブロックし、Amazon Q が GitLab インスタンスにコールバックできない。

解決策: インバウンドリクエストをブロックしているものを特定し、Amazon Q からのインバウンドリクエストを受け入れるように変更します。次のいずれかの形式になります。

- プロキシ
- ファイアウォールレイヤー
- 任意のインフラストラクチャレイヤーの拒否リストまたは許可リスト

再同期するには、GitLab インスタンスに再オンボードする必要があります。詳細については、GitLab ドキュメントの「Amazon Q GitLab Duoでのセットアップ」を参照してください。

Gitlab と Amazon Q の接続が同期していない

問題: Amazon Q アプリケーションGitLab Duoを変更し、GitLab と Amazon Q 間の接続が同期しなくなりました。

解決策: GitLab と Amazon Q が同期していない場合、Amazon Q が GitLab を呼び出すと、認証情報が無効になり、認証情報を更新できず、GitLab からの不正な応答が発生する可能性があります。

インスタンスドメイン名を解決できません

問題: オンボーディング後に GitLab インスタンス URL を変更しましたが、GitLab と Amazon Q 間の接続が同期しなくなりました。Amazon Q は GitLab インスタンスを正常に呼び出すことができなくなります。

解決策: ドメイン名が解決可能であることを確認する必要があります。GitLab インスタンスに再オンボードします。詳細については、GitLab ドキュメントの「Amazon Q GitLab Duoでのセットアップ」を参照してください。

IAM ロールと ID プロバイダー (IdP) が正しくない

問題: IAM ロールが Amazon Q GitLab Duoで に必要な APIs に正しいアクセス許可を提供していません。

解決策: ID プロバイダー (IdP) と IAM ロールが正しく設定されていることを確認します。詳細については、GitLab ドキュメントの「Amazon Q GitLab Duoでのセットアップ」を参照してください。

Amazon Q Developer プロファイルが存在しません

問題: Amazon Q GitLab Duoで にオンボードしようとしていますが、次の問題が発生します。 `Application could not be craeted by the AI Gateway: Error 400 - {"detail":"An error occured (ValidationException) when calling the CreateOAuthAppConnection operation: ProfileDoesNotExist"}`

解決策: まず Amazon Q Developer コンソールを使用して Amazon Q Developer プロファイルを作成する必要があります。詳細については、GitLab ドキュメントの「Amazon Q GitLab Duoでのセットアップ」を参照してください。

Amazon Q Developer for GitHub (プレビュー)

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

Amazon Q Developer for GitHubまたは GitHub Enterprise Cloud を使用すると、ソフトウェア開発ワークフローに Amazon Q Developer の機能を活用できます。特殊な開発エージェントを使用すると、新しいアイデアの実装、品質問題のコードのレビュー、ユニットテストによる脆弱性への対応、レガシー Java アプリケーションのモダナイズを行うことができます。エージェントがタスクを完了すると、フィードバックを提供でき、エージェントは前のソリューションを繰り返し実行します。詳細については、「Amazon Q Developer エージェント」を参照してください。

から Amazon Q Developer 統合にアクセスしGitHub、組織のリポジトリへのアクセスを許可できます。Amazon Q Developer for の使用を開始するにはGitHub、「」を参照してくださいクイックスタート: のインストール、 の機能の使用GitHub、使用制限の引き上げ。

> **Important**
>
> Amazon Q Developer アプリをインストールしてGitHubリポジトリへのアクセスを許可するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「組織内に GitHub アプリとロールをインストールするための要件https://docs.github.com/en/organizations/managing-peoples-access-to-your-organization-with-roles/roles-in-an-organization」を参照してください。 GitHub

> **Note**
>
> Amazon Q Developer と の統合は、米国のデータGitHubを処理します。詳細については、「Amazon Q Developer でのクロスリージョン処理」を参照してください。

> **Note**
>
> Amazon Q Developer for GitHub (プレビュー) は現在、サービスの改善のためにコンテンツを使用しません。今後これを有効にすると、適切な通知と、そのような使用をオプトアウトする方法が提供されます。

トピック

- Amazon Q Developer アプリのインストールとアクセスの承認
- Amazon Q Developer エージェント
- アプリのインストールの登録
- でのブラウザ拡張機能の使用 GitHub
- GitHub 問題とプルリクエストでのスラッシュコマンドの使用
- クイックスタート: のインストール、 の機能の使用GitHub、使用制限の引き上げ
- での Amazon Q Developer による機能の開発と反復処理 GitHub
- での Amazon Q Developer によるコードの確認 GitHub
- での Amazon Q Developer によるコードの変換 GitHub
- コード変換のワークフローのカスタマイズ
- Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定

- 登録済みインストールの詳細の設定
- の Amazon Q Developer の問題のトラブルシューティング GitHub

Amazon Q Developer アプリのインストールとアクセスの承認

GitHub 組織管理者は、 から Amazon Q Developer アプリGitHubを無料でインストールして設定できます。開始するために AWS アカウントを設定する必要はありません。インストールプロセス中に、GitHub組織内のすべてのリポジトリまたは選択したリポジトリへのアクセスを提供することを選択します。をインストールして承認すると、 の Amazon Q Developer 機能を無料で使用できますGitHub。Amazon Q Developer コンソールでアプリのインストールを登録することで、無料利用を増やすことができます。詳細については、「クイックスタート: のインストール、 の機能の使用 GitHub、使用制限の引き上げ」を参照してください。

> **Important**
>
> Amazon Q Developer アプリをインストールしてGitHubリポジトリへのアクセスを許可するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「組織内に GitHub アプリとロールをインストールするための要件https://docs.github.com/en/organizations/managing-peoples-access-to-your-organization-with-roles/roles-in-an-organization」を参照してください。 GitHub

> **Note**
>
> GitHub エンタープライズ組織が IP 許可リストを有効にしている場合は、GitHub アプリで許可された IP アドレスを受け入れる必要があります。IP アドレスを許可リストに自動的に追加することもできます。詳細については、GitHub ドキュメントの「GitHub Apps によるアクセスの許可」および「許可された IP アドレスの有効化GitHub 」を参照してください。GitHub リソースへのアクセスには、次の IP アドレスが使用されます。
>
> ```
> 34.228.181.128
> 44.219.176.187
> 54.226.244.221
> ```

Amazon Q Developer エージェント

Amazon Q Developer エージェントは、コーディング、テスト、デプロイからアプリケーションのトラブルシューティングとモダナイズまで、ソフトウェア開発ライフサイクル全体でサポートを提供します。

- Amazon Q 開発エージェント – 問題を作成し、機能開発ラベルを追加すると、Amazon Q Developer は新機能とバグ修正を自動的に実装します。Amazon Q Developer は、変更と変更の概要を含むプルリクエストを作成します。ラベルを適用する代わりに、問題のコメントで`/q dev`スラッシュコマンドを使用して機能開発を開始することもできます。詳細については、「での Amazon Q Developer による機能の開発と反復処理 GitHub」を参照してください。
- Amazon Q コードレビューエージェント – 新しいプルリクエストが作成されるか、クローズドプルリクエストが再開されると、Amazon Q Developer は自動的にコードレビューを実行し、コードの品質、潜在的な問題、セキュリティ上の懸念に関するフィードバックを提供します。Amazon Q Developer は、特定された問題の修正も生成します。これは、確認してプルリクエストにコミットすることを選択できます。自動コードレビューは、既存のプルリクエスト内で行われた後続のコミットによってトリガーされません。`/q review` スラッシュコマンドを使用して、プルリクエスト内で追加のコードレビューを開始できます。詳細については、「での Amazon Q Developer によるコードの確認 GitHub」を参照してください。
- Amazon Q transform エージェント – 問題を作成してコード変換ラベルを追加すると、Amazon Q Developer はコードを Java バージョン 8 または 11 からバージョン 17 に変換します。Amazon Q Developer は、変更と変更の概要を含むプルリクエストを作成します。ラベルを適用する代わりに、問題のコメントで`/q transform`スラッシュコマンドを使用してコード変換を開始することもできます。詳細については、「での Amazon Q Developer によるコードの変換 GitHub」を参照してください。

> **Important**
>
> Amazon Q Developer アプリは、アクセスを許可する GitHub リポジトリに Amazon Q 開発エージェントと Amazon Q Transform エージェントラベルを自動的に作成しようとします。ラベルが自動的に作成されない場合や、意図せずに削除された場合は、 で手動で作成できますGitHub。ラベルを Amazon Q Developer ラベルとして認識して処理するには、ラベルに Amazon Q 開発エージェントと Amazon Q 変換エージェントという名前を付ける必要があります。詳細については、GitHub ドキュメントの「ラベルの作成」を参照してください。

アプリのインストールの登録

の Amazon Q Developer 統合GitHubは無料で利用できます。開始するためにアカウント AWS を設定する必要はありません。機能開発とコード変換では 1 か月あたりの呼び出しが制限され、コードレビューでは 1 か月あたりの行数が制限されます。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、無料利用を増やすことができます。詳細については、「Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定」を参照してください。

> **Important**
>
> Amazon Q Developer コンソールでアプリのインストールを登録するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「GitHub アプリと OAuth アプリと組織をインストールするための要件GitHub」を参照してください。OAuth

でのブラウザ拡張機能の使用 GitHub

サポートされているブラウザで Amazon Q Developer 拡張機能を使用すると、ラベルメニューを検索しなくても、GitHub問題で機能開発やコード変換のラベルをすばやく追加できます。

Amazon Q Developer 拡張機能は、次のブラウザで使用できます。

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

GitHub 問題とプルリクエストでのスラッシュコマンドの使用

GitHub 問題内のコメントでスラッシュコマンドを使用したり、リクエストをプルしたりして、Amazon Q Developer を呼び出して開発タスクを実行したり、サポートを提供したりできます。

- `/q dev` - GitHub問題で Amazon Q Developer を呼び出して、新機能とバグ修正を自動的に実装します。Amazon Q Developer は、変更と変更の概要を含むプルリクエストを作成します。
- `/q review` - Amazon Q Developer を呼び出して、プルリクエストが作成または再開されたときにコードレビューを自動的に実行し、コードの品質、潜在的な問題、セキュリティ上の懸念、および推奨される修正に関するフィードバックを提供します。自動レビューは、既存のプルリクエストへの後続のコミットによってトリガーされません。

- `/q transform` - コード変換ラベルで問題が作成されると、Amazon Q Developer を呼び出して Java 8 または 11 コードを Java 17 に自動的に変換します。Amazon Q Developer は、変更と変更の概要を含むプルリクエストを作成します。
- `/q help` - スラッシュコマンドGitHub、機能、カスタマイズの詳細、Amazon Q Developer Guide のAmazon Q Developer for GitHub (プレビュー) ドキュメントへのリンクなど、Amazon Q Developer for に関する情報を提供します。

クイックスタート: のインストール、 の機能の使用GitHub、使用制限の引き上げ

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

このチュートリアルでは、以下のタスクのチュートリアルを提供します。

1. GitHub Marketplace から Amazon Q Developer アプリをインストールし、リポジトリへのアクセスを提供します。
2. 問題が発生した場合は、機能開発またはコード変換用のラベルを追加するか、コードレビュー用の新しいプルリクエストを作成して、Amazon Q Developer の使用を開始します。または、問題でスラッシュコマンドを使用して、機能開発またはコード変換を開始することもできます。スラッシュコマンドを使用して、プルリクエスト内で追加のコードレビューを開始することもできます。
3. (オプション) Amazon Q Developer アプリのインストールを AWS アカウントに登録して、使用制限を増やします。

ステップ 1: に Amazon Q Developer をインストールGitHubしてアクセスを許可する

Amazon Q Developer は、 AWS アカウントをセットアップしなくてもGitHub無料で使用できます。で Amazon Q Developer を使用する最初のステップGitHubは、 GitHub からアプリケーションをインストールすることです。このプロセス中に、Amazon Q Developer にすべてのGitHubリポジトリまたは選択したリポジトリへのアクセスを提供できます。

> **Important**
>
> Amazon Q Developer アプリをインストールしてGitHubリポジトリへのアクセスを許可するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「組織内に GitHub アプリとロールをインストールするための要件https://docs.github.com/en/organizations/managing-peoples-access-to-your-organization-with-roles/roles-in-an-organization」を参照してください。 GitHub

Amazon Q Developer をインストールしてアクセスを許可するには

1. Amazon Q Developer for GitHub app ページに移動します。
2. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。
3. Amazon Q Developer アプリの概要と機能を確認し、インストールを選択します。
4. GitHub リポジトリへのアクセスを設定するには、次のいずれかを実行します。
   a. 現在および今後のすべてのリポジトリへのアクセスを許可するには、[すべてのリポジトリ]を選択します。
   b. 特定のリポジトリへのアクセスを許可するには、リポジトリの選択のみを選択し、リポジトリの選択ドロップダウンを選択し、アクセスを許可するリポジトリを選択します。
5. インストールを選択して、 への Amazon Q Developer のインストールを完了GitHubし、リポジトリへのアクセスを承認します。

にアプリをインストールしてアクセスを承認するGitHubと、 の Amazon Q Developer の概要ページにリダイレクトされますGitHub。GitHub リポジトリに移動して、Amazon Q Developer 機能の使用を開始できます。

> **Note**
>
> GitHub エンタープライズ組織が IP 許可リストを有効にしている場合は、GitHub アプリで許可された IP アドレスを受け入れる必要があります。IP アドレスを許可リストに自動的に追加することもできます。詳細については、GitHub ドキュメントの「GitHub Apps によるアクセスの許可」および「許可された IP アドレスの有効化GitHub 」を参照してください。GitHub リソースへのアクセスには、次の IP アドレスが使用されます。
>
> ```
> 34.228.181.128
> 44.219.176.187
> ```

```
54.226.244.221
```

ステップ 2: で Amazon Q Developer 機能を使用する GitHub

Amazon Q Developer アプリを にインストールGitHubし、リポジトリへのアクセスを承認すると、アプリケーションのコーディング、テスト、デプロイからトラブルシューティング、モダナイズまで、ソフトウェア開発ライフサイクル全体で Amazon Q Developer エージェントのサポートを開始できます。詳細については、以下のいずれかを参照してください。

> **Important**
>
> Amazon Q Developer アプリは、アクセスを許可する GitHub リポジトリに Amazon Q 開発エージェントと Amazon Q Transform エージェントラベルを自動的に作成しようとします。ラベルが自動的に作成されない場合や、意図せずに削除された場合は、 で手動で作成できますGitHub。ラベルを Amazon Q Developer ラベルとして認識して処理するには、ラベルに Amazon Q 開発エージェントと Amazon Q 変換エージェントという名前を付ける必要があります。詳細については、GitHub ドキュメントの「ラベルの作成」を参照してください。

- での Amazon Q Developer による機能の開発と反復処理 GitHub
- での Amazon Q Developer によるコードの確認 GitHub
- での Amazon Q Developer によるコードの変換 GitHub

ステップ 3: 無料使用制限を増やし、詳細を設定する

で Amazon Q Developer エージェントGitHubを無料で使用できます。開始するために AWS アカウントを設定する必要はありません。機能開発とコードレビューのための 1 か月あたりの呼び出し数には制限があり、1 か月あたりのコード変換行数には制限があります。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、いつでも無料利用を増やすことができます。を登録すると、コードレビューの無効化や、検索とフィルタリングのためのタグの追加などの詳細を設定することもできます。詳細については、「Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定」を参照してください。

> **Important**
>
> Amazon Q Developer コンソールでアプリのインストールを登録するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「GitHub アプリと OAuth アプリと組織をインストールするための要件GitHub」を参照してください。OAuth

での Amazon Q Developer による機能の開発と反復処理 GitHub

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

で Amazon Q Developer を使用するとGitHub、新機能とバグ修正を自動的に実装し、タスクをアイデアから完成したプルリクエストに取り込むことで、開発を合理化できます。機能開発ラベルを問題に追加するか/q dev、スラッシュコマンドを使用すると、Amazon Q Developer はタイトルや説明、リポジトリコードをコンテキストとして使用して、新しいコード修正を生成し、プルリクエストを作成します。プルリクエストでは、フィードバックを提供でき、Amazon Q Developer は提案されたコードを繰り返し実行します。

Amazon Q Developer で機能開発を 1 か月あたり限られた回数で実行できます。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、いつでも無料利用を増やすことができます。詳細については、「Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定」を参照してください。

> **Important**
>
> Amazon Q Developer アプリは、アクセスを許可する GitHub リポジトリに Amazon Q 開発エージェントと Amazon Q Transform エージェントラベルを自動的に作成しようとします。ラベルが自動的に作成されない場合や、意図せずに削除された場合は、 で手動で作成できますGitHub。ラベルを Amazon Q Developer ラベルとして認識して処理するには、ラベルに Amazon Q 開発エージェントと Amazon Q 変換エージェントという名前を付ける必要があります。詳細については、GitHub ドキュメントの「ラベルの作成」を参照してください。

機能開発に Amazon Q Developer を使用するには

1. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。
2. GitHub 組織に移動し、Amazon Q Developer で新機能を実装するリポジトリに移動します。
3. 問題を選択し、新しい問題を作成するか、既存の問題を選択します。詳細については、GitHub ドキュメントの「問題の作成」を参照してください。

   - 新しい問題については、「タイトルテキスト入力の追加」フィールドに、機能開発のために Amazon Q Developer にコンテキストを提供するタイトルを入力します (例：「画像認識アプリの作成」)。問題の説明はコンテキストも提供するため、含める必要があります。

     既存の問題については、問題のタイトルと説明を編集して、機能開発のコンテキストを Amazon Q Developer に提供できます。詳細については、 GitHubドキュメントの「問題の編集」を参照してください。
4. 問題の作成時または既存の問題の設定時に、機能開発 Amazon Q Developer `/q dev` ラベルを適用するか、スラッシュコマンドを使用できます。次のいずれかを行います：

   - 問題にラベルを適用するには、次のいずれかを実行します。
     - ブラウザ拡張機能として提供されている Amazon Q に割り当てるドロップダウンメニューを選択し、Amazon Q 開発エージェントラベルを選択します。
     - 右側のメニューで、ラベルを選択し、Amazon Q 開発エージェントラベルを選択します。
   - コメントで`/q dev`スラッシュコマンドを使用するには：

     1. 問題内で、コメントの追加に移動し、コメントテキスト入力フィールドに「」と入力します`/q dev`。
     2. [Comment] (コメント) を選択します。
5. 新しい問題の場合は、問題の作成を選択して、Amazon Q Developer が機能を開発するために必要な詳細を含む問題の作成を完了します。既存の問題を設定する場合は、必ず変更を保存してください。詳細については、 GitHubドキュメントの「問題の編集」を参照してください。

   Amazon Q Developer が機能開発のコード変更の生成を完了すると、問題にコメントし、プルリクエストを開きます。
6. Amazon Q Developer が残したコメント (例：`I finished generating the proposed code changes and opened a pull request: Resolve Create an image recognition app`「」) に移動し、プルリクエストリンクを選択します。

プルリクエストタブに移動し、Amazon Q Developer によって作成されたプルリクエストを選択することもできます。

7. ファイル変更タブを選択して、コードの変更を表示します。
8. 提案されたコード変更に満足したら、プルリクエストをマージできます。詳細については、「プルリクエストをマージする」を参照してください。

Amazon Q Developer にフィードバックを提供することで、機能開発のプルリクエストを確認し、推奨されるコード変更を反復処理することもできます。

Amazon Q Developer 機能開発コードで反復処理するには

1. Amazon Q Developer によって作成されたプルリクエストを選択し、ファイルの変更タブを選択してコードの変更を表示します。
2. フィードバックを残して Amazon Q Developer に繰り返しさせるコードの行については、+ を選択してフィードバックを含むコメントを追加し、レビューの開始を選択します。

   レビューを送信すると、Amazon Q Developer はフィードバックに基づいてコードのレビューを開始します。Amazon Q Developer がフィードバックの修正を完了すると、プルリクエストに変更がコミットされ、コメントで更新されます。
3. 「レビューの完了」を選択し、「変更のリクエスト」ラジオボタンを選択し、「レビューの送信」を選択してフィードバックをレビューします。
4. 更新されたコードの変更に満足したら、プルリクエストをマージするか、新しいフィードバックでコードを再度繰り返すことができます。詳細については、「プルリクエストをマージする」を参照してください。

Amazon Q Developer は GitHub Actions ワークフローと統合して、インテリジェントなフィードバックループシステムを作成します。プルリクエストが作成されると、ユニットテストなどの設定されたワークフローが自動的に実行されます。ワークフローが失敗した場合、Amazon Q Developer は出力を分析し、結果と失敗情報に基づいてコードの変更を反復します。コードの変更がプッシュされ、ワークフローが実行されると、Amazon Q Developer は結果を解釈し、この情報を使用してアプローチを絞り込みます。これにより、各反復が以前の試行に基づいて構築され、Amazon Q Developer がそのソリューションを進化できる継続的な学習ループが作成されます。

フィードバックループの有効性は、Amazon Q Developer が解釈して行動するための有意義なフィードバックを提供する、明確に定義されたテストとワークフローがあることに依存します。リポジトリ内の GitHub Actions YAML ファイルを使用してワークフローを設定できるため、Amazon Q

Developer が応答できるカスタマイズされたテストおよび検証プロセスが可能になります。詳細については、 GitHubドキュメントのGitHub「アクションについて」を参照してください。

での Amazon Q Developer によるコードの確認 GitHub

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

Amazon Q Developer は、 内で自動コードレビューを有効にしますGitHub。新しいプルリクエストを作成するか、クローズドプルリクエストを再開すると、Amazon Q Developer は自動的にコードレビューを実行し、コードの品質、潜在的な問題、重要度の高い検出結果に関するフィードバックを提供します。Amazon Q Developer は、特定された問題の修正も生成します。この修正は、確認してプルリクエストにコミットすることを選択できます。自動コードレビューは、既存のプルリクエスト内で行われた後続のコミットによってトリガーされません。

`/q review` スラッシュコマンドを使用して、プルリクエスト内でコードレビューを開始することもできます。スラッシュコマンドを新しいプルリクエストコメントに追加できます。これにより、コメントや新しいコミットなど、現在の状態でプルリクエストの新しいコードレビューが開始されます。詳細については、「GitHub プルリクエストで Amazon Q Developer を使用してコードレビューを開始する」を参照してください。

Amazon Q Developer は、1 か月あたり限られた数の行に対してコードレビューを実行できます。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、いつでも無料利用を増やすことができます。詳細については、「Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定」を参照してください。

> **Note**
>
> コードレビュー機能が無効になっている場合は、 でラベルを適用する前に、Amazon Q Developer コンソールで有効にする必要がありますGitHub。詳細については、「での Amazon Q Developer の機能の編集 GitHub」を参照してください。

Amazon Q Developer をコードレビューに使用して修正を適用するには

レビューを開始する前に、 `project-root/.amazonq/rules` ディレクトリのシンプルな Markdown ファイルでカスタムコーディング標準を定義することで、コード品質レビューをカスタマイズできます。Amazon Q は自動的にガイドラインに従い、プロジェクト全体で一貫したコード品質を確保します。詳細については、「サードパーティープラットフォームでの Amazon Q Developer のプロジェクトルールの作成」を参照してください。

1. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。
2. GitHub 組織に移動し、Amazon Q Developer でコードレビューを実行するリポジトリに移動します。
3. ソースコードに加えられた変更のプルリクエストを新しく作成します。詳細については、GitHub ドキュメントの「プルリクエストの作成」を参照してください。

   新しいプルリクエストを作成すると、Amazon Q Developer は自動的にコードレビューを開始し、潜在的な問題を見つけます。Amazon Q Developer がレビューを完了すると、各検出結果を個別のコメントとして送信し、プルリクエストにコミットできる修正案も送信します。
4. Amazon Q Developer によって提案されたコード変更を確認し、コミット提案を選択し、コミット変更を選択してプルリクエストを更新します。
5. 推奨されるコード修正に満足している場合は、プルリクエストをマージして、Amazon Q Developer によって提案されたコード変更を適用できます。詳細については、GitHub ドキュメントの「プルリクエストのマージ」を参照してください。

GitHub プルリクエストで Amazon Q Developer を使用してコードレビューを開始する

Amazon Q Developer が新規または再開されたGitHubプルリクエストに対して自動コードレビューを実行した後、`/q review`スラッシュコマンドを使用して追加のコードレビューを開始してコードを繰り返し実行できます。コードレビューは、プルリクエストの差分全体に対して実行されます。

> **Note**
>
> コードレビューは、新しいコメントを含むプルリクエスト内でのみ開始できます。`/q review` スラッシュコマンドは、既存のコメントスレッドでは機能しません。

プルリクエストでコードレビューの開始を使用するには

1. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。

2. GitHub 組織に移動し、Amazon Q Developer でコードレビューを実行するプルリクエストに移動します。詳細については、「プルリクエストについて」を参照してください。
3. プルリクエスト内でコメントの追加に移動し、コメントテキスト入力フィールドに と入力します`/q review`。
4. コメント を選択してコードレビューを開始します。

   Amazon Q Developer がプルリクエストコードの分析を完了するまでに数分かかる場合があります。Amazon Q Developer が分析を完了すると、重大度と重要度の検出結果、およびプルリクエストのコミットと更新を選択できる変更案とともにコメントに返信します。
5. 推奨されるコード修正に満足している場合は、プルリクエストをマージして、Amazon Q Developer によって提案されたコード変更を適用できます。詳細については、GitHub ドキュメントの「プルリクエストのマージ」を参照してください。

での Amazon Q Developer によるコードの変換 GitHub

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

コード変換機能を使用すると、既存のコードのコア機能を維持しながら、テクノロジースタックを更新し、パフォーマンスを向上させ、将来の互換性を確保できます。Amazon Q Developer を使用すると、技術的な負債を回避し、長期的な持続可能性とスケーラビリティのためのアプリケーションを構築できます。

コード変換ラベルを問題に追加するか、スラッシュコマンドを使用すると、Amazon Q Developer `/q transform` はコードを Java バージョン 8 または 11 からバージョン 17 に変換します。次に、Amazon Q Developer は、メインラインにマージできる変更と変更の概要を含むプルリクエストを作成します。

Amazon Q Developer は、コード変換を 1 か月あたり限られた回数だけ実行できます。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、いつでも無料利用を増やすことができます。詳細については、「Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定」を参照してください。

前提条件

Amazon Q Developer でコードベースを変換する前に、次の要件を考慮してください。

- GitHub リポジトリでGitHubアクションが有効になっていることを確認します。詳細については、GitHub ドキュメントの「GitHub アクションの理解」と「ワークフローの有効化」を参照してください。 GitHub
- `.github/workflows/` ディレクトリでワークフローファイルを作成して設定し、コード変換タスクを処理します。詳細については、「コード変換のワークフローのカスタマイズ」を参照してください。

前提条件を満たしたら、Amazon Q transform エージェントラベルを適用し、GitHub リポジトリのコードベースをアップグレードできます。

> **Important**
>
> Amazon Q Developer アプリは、アクセスを許可する GitHub リポジトリに Amazon Q 開発エージェントと Amazon Q Transform エージェントラベルを自動的に作成しようとします。ラベルが自動的に作成されない場合や、意図せずに削除された場合は、 で手動で作成できますGitHub。ラベルを Amazon Q Developer ラベルとして認識して処理するには、ラベルに Amazon Q 開発エージェントと Amazon Q 変換エージェントという名前を付ける必要があります。詳細については、GitHub ドキュメントの「ラベルの作成」を参照してください。

コードベースをアップグレードするには

1. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。
2. GitHub 組織に移動し、コードベースを変換するリポジトリに移動します。
3. 問題を選択し、新しい問題を作成します。詳細については、GitHub ドキュメントの「問題の作成」を参照してください。
4. タイトルテキスト入力の追加フィールドに、コード変換のために Amazon Q Developer にコンテキストを提供するタイトルを入力します (例：「コードベースのアップグレード」)。問題の説明はコンテキストも提供するため、含める必要があります。
5. 問題の作成時または既存の問題の設定時に、コード変換 Amazon Q Developer `/q transform` ラベルを適用するか、スラッシュコマンドを使用できます。次のいずれかを行います：

   - 問題にラベルを適用するには、次のいずれかを実行します。

- ブラウザ拡張機能として提供されている Amazon Q に割り当てるドロップダウンメニューを選択し、Amazon Q 変換エージェントラベルを選択します。
- 右側のメニューで、ラベルを選択し、Amazon Q 変換エージェントラベルを選択します。

- コメントで`/q transform`スラッシュコマンドを使用するには：

  1. 問題内で、コメントの追加に移動し、コメントテキスト入力フィールドに「」と入力します`/q transform`。
  2. [Comment] (コメント) を選択します。

6. 問題の作成を選択して、Amazon Q Developer がコードを変換するために必要な詳細を含む問題の作成を完了します。

   Amazon Q Developer が機能開発のコード変更の生成を完了すると、問題にコメントし、プルリクエストを開きます。

7. Amazon Q Developer が残したコメント (例：`I finished upgrading your code and opened a pull request: Resolve Upgrade codebase.`「」) に移動し、プルリクエストリンクを選択します。

   プルリクエストタブに移動し、Amazon Q Developer によって作成されたプルリクエストを選択することもできます。

8. ファイル変更タブを選択して変更を表示します。
9. 提案されたコード変更に満足したら、プルリクエストをマージできます。詳細については、「プルリクエストをマージする」を参照してください。

Amazon Q Developer は、更新したファイルに関する詳細を含む変換の概要を提供するコメントを残します。概要は、コメントにある `build_logs.txt` ファイルにあります。Amazon Q Developer が完全なコード変換を実行できない場合、完全な変換を妨げたエラーに関する概要も表示されます。

コード変換のワークフローのカスタマイズ

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

タスクの完了後に Amazon Q Developer コード変換エージェントが GitHub でプルリクエストを作成すると、プロジェクトパイプラインはプルリクエストに対して実行するように設定されたジョブを実行します。更新されたコードは Java 17 をターゲットとしているため、ジョブが Java 8 または Java 11 を使用してビルドしようとすると、ジョブにビルドエラーが発生します。

Amazon Q transform エージェントラベルをGitHub問題に適用する前に、コード変換タスクを処理するワークフローファイルを `.github/workflows/` ディレクトリに作成して設定する必要があります。

コード変換のワークフローをカスタマイズするには

1. リポジトリにワークフローがない場合は、GitHub Actions ワークフローファイルを作成します。詳細については、GitHub ドキュメントの「GitHub アクションのクイックスタート」および「ワークフロー例の作成GitHub」を参照してください。
2. 次のジョブを使用してワークフローファイルを更新します。

```
q-code-transformation:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-java@v3
        with:
          java-version: '17'
          distribution: 'adopt'

      - name: Build and copy dependencies
        run: |
          mvn ${{ env.MAVEN_CLI_OPTS }} verify
          mvn ${{ env.MAVEN_CLI_OPTS }} dependency:copy-dependencies -
DoutputDirectory=dependencies -Dmdep.useRepositoryLayout=true -Dmdep.copyPom=true -
Dmdep.addParentPoms=true

      - name: Upload artifacts
        uses: actions/upload-artifact@v4
        with:
          name: q-code-transformation-dependencies
          path: dependencies
```

GitHub ワークフロー用の Java バージョンを動的に選択する

次の動的コードスニペットは、Amazon Q コード変換エージェントによって作成されたプルリクエストでジョブが実行されているときに、Java 17 を動的に選択します。オープンプルリクエストをデフォルトブランチにマージする場合は、デフォルトで Java 17 を使用するように.github/workflows/ディレクトリのワークフローファイルを変更する必要があります。

```
name: Q Code Transformation

on:
  push:
    branches:
      - 'Q-TRANSFORM-issue-*'

env:
   MAVEN_CLI_OPTS: >-
     -Djava.version=${{ contains(github.event.head_commit.message, 'Code transformation
 completed') &&  '17' || '1.8' }}

jobs:
  q-code-transformation:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-java@v3
        with:
          java-version: ${{ contains(github.event.head_commit.message, 'Code
 transformation completed') && '17' || '8' }}

          distribution: 'adopt'

      - name: Build and copy dependencies
        run: |
          mvn ${{ env.MAVEN_CLI_OPTS }} verify
          mvn ${{ env.MAVEN_CLI_OPTS }} dependency:copy-dependencies -
DoutputDirectory=dependencies -Dmdep.useRepositoryLayout=true -Dmdep.copyPom=true -
Dmdep.addParentPoms=true

      - name: Upload artifacts
        uses: actions/upload-artifact@v4
        with:
          name: q-code-transformation-dependencies
```

```
          path: dependencies
```

GitHub ワークフローでのファーストパーティ依存関係の処理

Amazon Q Developer を使用してコード変換を行う場合、ファーストパーティの依存関係を処理するようにGitHubワークフローを強化できます。

標準の Amazon Q Developer ワークフローは、パブリック Maven 依存関係を持つプロジェクトを処理します。ただし、エンタープライズプロジェクトには、多くの場合、パブリックリポジトリでは利用できないプライベート Java ARchive (JAR ファイル) などのファーストパーティの依存関係が含まれます。この拡張ワークフローは、標準設定を拡張して、パブリック依存関係とプライベート依存関係の両方をサポートします。

拡張ワークフローのユースケース

プロジェクトに次のものがある場合は、拡張GitHubワークフロー設定を使用します。

- プライベートまたは内部 JAR の依存関係
- Maven Central では利用できないカスタムユーティリティライブラリ
- `Could not resolve dependencies` エラーによるビルドの失敗

サンプルの構成

次の例は、ファーストパーティの依存関係を処理する完全なワークフロー設定を示しています。

```
name: Q Code Transformation

on:
  push:
    branches:
      - 'Q-TRANSFORM-issue-*'

env:
   MAVEN_CLI_OPTS: >-
     -Djava.version=${{ contains(github.event.head_commit.message, 'Code transformation
 completed') && '17' || '1.8' }}
     -Dmaven.compiler.source=${{ contains(github.event.head_commit.message, 'Code
 transformation completed') && '17' || '1.8' }}
     -Dmaven.compiler.target=${{ contains(github.event.head_commit.message, 'Code
 transformation completed') && '17' || '1.8' }}
```

```
jobs:
  q-code-transformation:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-java@v4
        with:
          java-version: ${{ contains(github.event.head_commit.message, 'Code
 transformation completed') && '17' || '8' }}
          distribution: 'temurin'

      - name: Cache Maven dependencies
        uses: actions/cache@v4
        with:
          path: ~/.m2
          key: ${{ runner.os }}-m2-${{ hashFiles('**/pom.xml') }}
          restore-keys: ${{ runner.os }}-m2

      - name: Install First-Party Dependencies
        run: |
          mvn install:install-file \
            -Dfile=./your-library/built-library/1.0.0/your-library-1.0.0.jar \
            -DgroupId=com.yourcompany.samples \
            -DartifactId=your-library \
            -Dversion=1.0.0 \
            -Dpackaging=jar

      - name: Build and copy dependencies
        run: |
          mvn ${{ env.MAVEN_CLI_OPTS }} verify
          mvn ${{ env.MAVEN_CLI_OPTS }} dependency:copy-dependencies -
DoutputDirectory=dependencies -Dmdep.useRepositoryLayout=true -Dmdep.copyPom=true -
Dmdep.addParentPoms=true

      - name: Upload artifacts
        uses: actions/upload-artifact@v4
        with:
          name: q-code-transformation-dependencies
          path: dependencies

      - name: Debug information
        if: failure()
        run: |
          echo "Branch: ${{ github.ref_name }}"
```

```
          echo "Commit message: ${{ github.event.head_commit.message }}"
          echo "Java version logic result:
${{ contains(github.event.head_commit.message, 'Code transformation completed') &&
'17' || '8' }}"
          echo "Local Maven repository contents:"
          find ~/.m2/repository/com/yourcompany/samples -name "*.jar" 2>/dev/null ||
echo "No your-library found in local repo"
          java -version
          mvn -version
```

拡張ワークフローの主要なコンポーネント

拡張 GitHub ワークフローには、いくつかの主要なコンポーネントがあります。

- ファーストパーティ依存関係のインストール

  ワークフローには、Maven `install:install-file` コマンドを使用してプライベート依存関係をインストールするための重要なステップが含まれています。これにより、変換中にローカル Maven リポジトリでプライベート依存関係を使用できるようになります。

- 動的 Java バージョン処理

  ワークフローは、環境変数を使用してコンパイラのプロパティを管理することで、変換状態に基づいて Java 8 と Java 17 を自動的に切り替えます。

- Maven の設定

  拡張ワークフローには、以下を含む包括的な Maven 設定が含まれています。

  - パフォーマンスを向上させるための依存キャッシュ
  - コンパイラプロパティの完全な管理
  - リポジトリレイアウトの保存
  - POM ファイル処理

- 実装手順
  - でプライベート依存関係を特定する `pom.xml`
  - リポジトリ構造で JAR ファイルを整理する
  - 特定のパスと座標を使用してワークフローをカスタマイズする
  - パターンに一致するブランチでワークフローをテストする `Q-TRANSFORM-issue-*`

- トラブルシューティング

ワークフローには、障害時にアクティブ化するデバッグ情報ステップが含まれており、以下を提供します。

- ブランチ情報
- メッセージの詳細をコミットする
- Java バージョン設定
- ローカル Maven リポジトリの内容
- Java および Maven のバージョン情報

Amazon Q Developer コンソールの使用制限の引き上げと詳細の設定

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

で Amazon Q Developer エージェントGitHubを無料で使用できます。開始するために AWS アカウントを設定する必要はありません。機能開発およびコードレビュー機能では 1 か月あたりの呼び出し数が制限されており、コード変換では 1 か月あたりの行数が指定されています。Amazon Q Developer アプリのインストールを AWS アカウントに登録することで、いつでも無料利用を増やすことができます。

デフォルトでは、コードレビュー、機能開発、およびコード変換機能は、アプリのインストールGitHub時に で有効になります。登録時にこれらの機能を無効にすることができます。登録するには、コンソールから機能を管理する Amazon Q Developer プロファイルが必要です。プロファイルには、設定とコードレコメンデーションのカスタマイズが保存されます。

> **Important**
>
> Amazon Q Developer コンソールでアプリのインストールを登録するには、GitHub組織の要件を満たしている必要があります。詳細については、GitHub ドキュメントの「GitHub アプリと OAuth アプリおよび組織をインストールするための要件GitHub」を参照してください。OAuth

Amazon Q Developer のインストールを登録するには

1. Amazon Q Developer コンソールに移動し、ナビゲーションペインで で Amazon Q Developer GitHubを選択します。
2. 「インストールの登録」を選択し、「 に誘導する認可」を選択しますGitHub。

   以前に Amazon Q Developer にGitHub組織へのアクセスを許可した場合は、Amazon Q Developer コンソールにリダイレクトされます。この場合、ステップ 4 に進みます。
3. GitHub 認証情報を使用してGitHubアカウントにサインインします。複数のアカウントがある場合は、Amazon Q Developer へのアクセスを許可するアカウントを選択します。
4. Amazon Q Developer にGitHubアカウントへのアクセスを許可するを選択します。認可後、Amazon Q Developer コンソールにリダイレクトされます。
5. Amazon Q Developer プロファイルの詳細を入力します。

   Amazon Q Developer の使用を開始するときに以前にプロファイルを作成した場合は、ステップ 6 に進みます。プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。

   - Amazon Q Developer プロファイルで、プロファイルの詳細を入力します。

     i. プロファイル名テキスト入力フィールドに、認識および参照できる名前を入力します。プロファイル名を適切に識別できるように、リージョンを含めることを検討してください (例: "QDevProfile_us-east-1")。

     ii. （オプション) 追加設定を選択し、プロファイルの説明 - オプションのテキスト入力フィールドに、プロファイルの説明を入力してコンテキストを指定し、プロファイルを識別します。
6. 登録の詳細で、GitHub詳細を入力し、オプションでコードレビュー機能を設定し、タグを追加します。

   a. 登録名のテキスト入力ファイルに、アプリのインストールの名前を入力します。

   b. （オプション) Organization name - optional text input フィールドに、アプリのインストールに関連付けられた組織の名前を入力します。

   c. （オプション) 機能で、トグルを選択してコードレビュー機能を設定し、機能を有効または無効にします。機能開発とコード変換の設定は変更できません。この機能はデフォルトで有効になっています。

   d. （オプション) タグ - オプションで、新しいタグを追加 を選択して、 で AWS リソースを検索およびフィルタリングするのに役立つタグを追加しますGitHub。

7. Register を選択して Amazon Q Developer プロファイルを作成し、 GitHubでアプリのインストールを AWS アカウントに登録します。

Amazon Q Developer プロファイルを正常に作成し、アプリのインストールを に登録すると GitHub、登録の詳細を表示できます。引き続き機能を有効または無効にしたり、後でタグを追加したりできます。GitHub アプリのインストールを削除することもできます。詳細については、「登録済みインストールの詳細の設定」を参照してください。

登録済みインストールの詳細の設定

> **Note**
>
> Amazon Q Developer for GitHubはプレビューリリースであり、変更される可能性があります。

Amazon Q Developer プロファイルを作成し、アプリのインストールを に登録したら GitHub、Amazon Q Developer コンソールから以下を実行できます。

- コードレビューの使用を有効または無効にします。機能開発とコード変換の設定は現在変更できません。
- GitHub 問題に Amazon Q Developer 機能ラベルを追加する方法を提供するブラウザ拡張機能へのリンクを見つけます。
- タグを追加して、リソースを検索およびフィルタリングしたり、 AWS コストを追跡したりできます。
- GitHub アプリのインストール登録を削除します。

での Amazon Q Developer の機能の編集 GitHub

で Amazon Q Developer で使用できる機能はGitHub、アプリを にインストールGitHubし、組織へのアクセスを承認すると、デフォルトで有効になります。でこの機能を使用しなくなった場合は、機能を無効にすることができますGitHub。

で Amazon Q Developer の機能を有効化または無効化するには GitHub

1. Amazon Q Developer コンソールに移動します。

2. ナビゲーションペインの の Amazon Q Developer GitHubで、登録済みインストールを選択します。
3. 名前 列で、機能を有効または無効にするインストールの登録名を選択します。
4. 機能 で、編集 を選択して機能の可用性を設定します。
5. モーダルで、有効または無効にする機能のトグルを選択し、保存を選択して変更を確認します。

Amazon Q Developer コンソールで機能を有効または無効にすると、変更は に反映されます GitHub。機能を無効にした後に Amazon Q Developer ラベルに問題を割り当てると、エラーが発生します。この機能が無効になっている場合、コードレビューは新しいプルリクエストでは行われなくなります。

ブラウザ拡張機能のインストール

Amazon Q Developer 拡張機能は、サポートされているブラウザのいずれかにインストールできます。拡張機能を使用すると、ラベルメニューを検索しなくても、GitHub問題が発生した場合に機能開発タスクまたはコード変換タスクを Amazon Q Developer にすばやく割り当てることができます。

Amazon Q Developer 拡張機能は、次のブラウザで使用できます。

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

Amazon Q Developer コンソールの登録インストールの詳細ページで、サポートされているブラウザを表示することもできます。

Amazon Q Developer GitHubアプリのインストール登録の削除

Amazon Q Developer コンソールを使用して、1 つ以上のGitHubアプリインストールの登録を削除できます。登録を完全に削除すると、登録に関連付けられたすべてのデータも削除されます。アクションを元に戻すことはできません。

GitHub アプリのインストールを削除するには

1. Amazon Q Developer コンソールに移動します。
2. ナビゲーションペインの の Amazon Q Developer GitHubで、登録済みインストールを選択します。

3. 次のいずれかを行います：

   - Actions 列で、削除するインストールの登録の削除を選択します。
   - Name 列で、削除するインストールの登録名を選択し、Delete を選択します。

   Actions 列で、削除する登録済みインストールの登録の削除を選択します。
4. モーダルで、登録の削除の詳細を確認します。
5. テキスト入力フィールドに と入力し**confirm**、削除を選択して変更を確認します。

GitHub アプリのインストールを削除したら、新しいインストールの登録を選択できます。

の Amazon Q Developer の問題のトラブルシューティング GitHub

Amazon Q Developer for を使用する際の一般的な問題のトラブルシューティングについては、次のセクションを参照してくださいGitHub。

Amazon Q Developer がブランチ保護ルールを持つリポジトリでプルリクエストを生成しない

問題: Amazon Q Developer は、ブランチ保護ルールがあるGitHubリポジトリにプルリクエストを作成できません。

解決策: ブランチ保護ルールは、Amazon Q Developer がブランチを作成してプルリクエストを作成できないようにします。ブランチ保護ルールを持つリポジトリGitHubで Amazon Q Developer for を使用するには、Amazon Q Developer アプリを許可リストに追加する必要があります。

Amazon Q Developer アプリを許可リストに追加するには

1. 必要に応じて、認証情報を使用してGitHubアカウントにサインインしますGitHub。
2. GitHub 組織に移動し、許可するリポジトリに移動して、GitHubAmazon Q Developer for app を一覧表示します。
3. 設定を選択し、ナビゲーションペインからブランチを選択します。
4. ブランチ保護ルールで、編集を選択してブランチ保護ルールを変更します。
5. 一致するブランチを作成するプッシュを制限するを選択し、GitHubAmazon Q Developer for app を検索します。
6. 変更を保存を選択して、Amazon Q Developer が Amazon Q Developer ラベルに関する問題のプルリクエストを作成できるようにします。

でのブランチ保護ルールの変更の詳細についてはGitHub、「ブランチ保護ルールの作成」を参照してください。

GitHub 問題に Amazon Q Developer ラベルがない

問題: GitHub の問題に Amazon Q 開発エージェントまたは Amazon Q 変換エージェントのラベルが表示されません。

解決策: GitHubアプリケーション用の Amazon Q Developer をインストールしたときにラベルが自動的に作成されない場合、またはラベルが意図せずに検索された場合は、 でラベルを手動で作成できますGitHub。ラベルを Amazon Q Developer ラベルとして認識して処理するには、ラベルに Amazon Q 開発エージェントと Amazon Q 変換エージェントという名前を付ける必要があります。詳細については、GitHub ドキュメントの「ラベルの作成」を参照してください。

Amazon Q Developer がGitHub問題のコードを作成していない

問題: GitHub の問題を作成し、タスクを実行するために Amazon Q Developer を呼び出しましたが、技術的な問題に関する次の一連のメッセージを受け取りました。

1. # この問題のコードの生成に取り組んでいます。この問題はコメントで更新し、完了したらプルリクエストを開きます。
2. ## 技術的な問題が発生しています。問題を解決できませんでした。もう一度試します。これには時間がかかる場合があります。
3. ## 技術的な問題が発生しています。問題を解決できませんでした。もう一度試します。これには時間がかかる場合があります。
4. # 技術的な問題が発生しています。問題を解決できませんでした。

   この問題をユーザーに再割り当てすることを検討してください。これにより、生成 AI 機能の使用に関するクォータを維持できます。

解決策: Amazon Q Developer が問題を処理してコードを生成できない場合は、新しい問題を作成し、Amazon Q 開発エージェントまたは Amazon Q Transform エージェントラベルを新しい問題に適用します。問題の作成と Amazon Q Developer エージェントラベルの適用の詳細については、での Amazon Q Developer による機能の開発と反復処理 GitHub「」または「」を参照してくださいでの Amazon Q Developer によるコードの変換 GitHub。

サードパーティープラットフォームでの Amazon Q Developer のプロジェクトルールの作成

GitLab または GitHub GitHub の Amazon Q Developer で使用できるプロジェクトルールのライブラリを構築できます。これらのルールでは、チーム全体のコーディング標準とベストプラクティスについて説明します。たとえば、すべての Python コードがタイプヒントを使用する必要があること、またはすべての Java コードが Javadoc コメントを使用する必要があることを示すルールがあるとします。これらのルールをプロジェクトに保存することで、経験レベルに関係なく、開発者間の一貫性を確保できます。

プロジェクトルールは、プロジェクトの *project-root*/.amazonq/rulesフォルダの Markdown ファイルで定義されます。

プロジェクトルールを作成すると、Amazon Q Developer はそれらを自動的にプロジェクト内のコンテキストとして使用し、機能開発用のコードを生成するときにそれらに従うようにします。

ファイルシステムを使用してプロジェクトルールを作成するには

1. サードパーティーリポジトリで、プロジェクトのルートフォルダを開きます。
2. プロジェクトルートフォルダで、次のフォルダを作成します。

   project-root/.amazonq/rules

   このフォルダには、すべてのプロジェクトルールが格納されます。
3. で*project-root*/.amazonq/rules、プロジェクトルールファイルを作成します。Markdown ファイルである必要があります。例:

   cdk-rules.md
4. プロジェクトルールの Markdown ファイルを開きます。
5. ファイルに詳細なプロンプトを追加します。例:

   ```
   All Amazon S3 buckets must have encryption enabled, enforce SSL, and block public
    access.
   All Amazon DynamoDB Streams tables must have encryption enabled.
   All Amazon SNS topics must have encryption enabled and enforce SSL.
   All Amazon SNS queues must enforce SSL.
   ```
6. 変更をコミット、レビュー、マージします。
7. (オプション) プロジェクトルールのマークダウンファイルを追加します。

これで、1 つ以上のプロジェクトルールが作成されました。Amazon Q は、これらのルールをプロジェクト内のコンテキストとして自動的に使用します。

チャットアプリケーションでの Amazon Q Developer とのチャット

Microsoft Teams および チャットアプリケーションで Amazon Q Developer とSlackチャットできます。設定されたチャネルでは、Amazon Q はソリューションの構築、問題のトラブルシューティング、次のステップの特定に関するベストプラクティスに関する質問に回答できます。設定済みのチャットアプリケーションでは、次の Amazon Q チャット機能を使用できます。

- に関するチャット AWS
- Amazon Q Developer とのリソースに関するチャット
- リソースの問題のトラブルシューティング
- コストに関するチャット
- テレメトリとオペレーションに関するチャット
- Reachability Analyzer の Amazon Q ネットワークのトラブルシューティング

チャットアプリケーションで Amazon Q を使用する際に使用できる機能の完全なセットの詳細については、「Amazon Q Developer in chat applications Administrator Guide」の「What is Amazon Q Developer in chat applications?」を参照してください。

> **Note**
>
> チャットアプリケーションで Amazon Q Developer を使用する場合、アクセスは Amazon Q Developer 無料利用枠に制限されます。

チャネルで Amazon Q チャットを有効にする

Amazon Q Developer で既に設定されている Microsoft Teamsまたは Slackチャネルにチャット機能を追加するには、次の手順を実行します。チャットアプリケーションで初めて Amazon Q Developer をセットアップし、ユーザーが Amazon Q とチャットできるようにするには、「チャットアプリケーションの Amazon Q Developer Microsoft Teams管理者ガイド」の「 の開始方法」と「 の開始方法Slack」を参照してください。

Microsoft Teams またはSlackチャネルから Amazon Q に質問する前に、Amazon Q をチャネルに追加する必要があります。まず、 AWS Identity and Access Management (IAM) ロール設定を更新し

てAmazonQDeveloperAccess管理ポリシーを含め、次にそのポリシーをチャネルガードレールとして追加します。管理者アクセスが必要な場合は、代わりに AmazonQFullAccess ポリシーを追加します。

1. AmazonQDeveloperAccess マネージドポリシーを自分の IAM ロールに追加します。
   a. にサインイン AWS Management Console し、IAM コンソールを開きます。
   b. IAM コンソールのナビゲーションペインで [ロール] を選択します。
   c. 修正するロールの名前を選択します。
   d. [アクセス許可ポリシー] で、[アクセス許可を追加]、[ポリシーのアタッチ] の順に選択します。
   e. 検索バーに「AmazonQDeveloperAccess」と入力します。
   f. [AmazonQDeveloperAccess] を選択します。
   g. [Add permissions (許可の追加)] を選択します。
2. AmazonQDeveloperAccess マネージドポリシーをチャネルガードレールに追加します。
   a. チャットアプリケーションコンソールで Amazon Q Developer を開きます。
   b. 設定済みのクライアントを選択します。
   c. 設定済みのチャネルを選択します。
   d. [ガードレールを設定] を選択します。
   e. 検索バーに「AmazonQDeveloperAccess」と入力します。
   f. [AmazonQDeveloperAccess] を選択します。
   g. [保存] を選択します。

チャネルで Amazon Q に質問する

設定が成功したことを確認するには、Amazon Q に質問します。「@Amazon Q」と入力し、その後に質問を入力します。

以下は、設定済みのチャネルで Amazon Q に尋ねることができる質問の例です。

- `@Amazon Q how do I troubleshoot lambda concurrency issues?`
- `@Amazon Q what are the best practices for securing S3 buckets?`
- `@Amazon Q what is the maximum zipped file size for a lambda?`
- `@Amazon Q get the configuration for my lambda function name?`

- `@Amazon Q what is the size of the auto scaling group name in us-east-2?`
- `@Amazon Q can you show ec2 instances running in us-east-1?`

提案のカスタマイズ

> **Note**
>
> Amazon Q Developer の以下の機能では、カスタマイズがサポートされています。
>
> - インライン提案 – 「インライン提案の生成」を参照してください。
> - IDE でのチャット – 「コードに関するチャット」を参照してください。

カスタマイズを利用すると、Amazon Q Developer が、チームの内部ライブラリ、独自のアルゴリズム手法、企業のコードスタイルに準拠した方法でソフトウェア開発を支援できるようになります。

Amazon Q のカスタマイズは、Amazon Q が貴社のコードベースに基づいて提案を行えるようにする一連の要素です。コードベースを含むデータソースを接続すると、Amazon Q はコンテンツを使用して組織のデベロッパーのスタイルに合わせたサポートを提供するようになります。

カスタマイズは、アカウント内で相互に完全に分離されます。また、他のお客様のデータからも分離されます。Amazon Q Developer 管理者によって指定されたユーザーのみが、特定のカスタマイズにアクセスできます。Amazon Q 管理者がどのユーザーがどのカスタマイズにアクセスできるかを指定する前に、その管理者のアクセス許可にその権限を付与する必要があります。詳細については、「Amazon Q のカスタマイズの前提条件」を参照してください。

トピック

- Amazon Q のカスタマイズの前提条件
- カスタマイズの作成
- カスタマイズの削除
- カスタマイズの最適化
- ログ記録とトラブルシューティング
- Amazon Q カスタマイズのアクティブ化
- Amazon Q のカスタマイズの更新
- Amazon Q のカスタマイズにユーザーとグループを追加する
- Amazon Q のカスタマイズの使用

Amazon Q のカスタマイズの前提条件

Amazon Q の カスタマイズは Amazon Q Developer Pro の基盤の上に構築され、その機能を使用します。

Amazon Q のカスタマイズを使用するには、まず「Amazon Q Developer の使用開始」で説明されている Amazon Q Developer Pro のセットアップ手順に従う必要があります。これには、Amazon Q カスタマイズへのアクセスを許可するユーザーを Amazon Q Developer Pro プロファイルに追加することも含まれます。

管理者の認可

Amazon Q のカスタマイズを使用する場合、Amazon Q 管理者はコードベースへのアクセスを許可されている必要があります。コードベースは Amazon S3 に、または AWS CodeConnectionsを通じて保存できます。ただし、Amazon Q Developer Pro の標準セットアッププロセスでは、 AWS Organizations 管理者は Amazon Q 管理者にこれらのサービスへのアクセスを提供しません。

つまり、カスタマイズを作成するには、管理者に追加のアクセス許可が必要です。必要なアクセス許可を付与するポリシーの例については、「管理者にカスタマイズの作成を許可する」を参照してください。

> **Note**
>
> GitHub をデータソースとして使用している場合は、特定のリポジトリの使用を制限できます。詳細については、デベロッパーツールコンソールのユーザーガイドにある「GitHub への接続を作成する」を参照してください。

> **Note**
>
> Amazon Q Developer Pro 用に設定した暗号化キーは、カスタマイズにも使用されます。

データの準備

カスタマイズは、最適なソースマテリアルを使用して作成することが重要です。データソースを準備するときは、チームで推奨されているパターンを含むコードを追加します。アンチパターン、バグ、セキュリティの脆弱性、パフォーマンスの問題などを含むコードは避けてください。

データソースを準備するには、次のガイドラインに従います。

- データソースには、サポートされている言語のソースコードファイルが少なくとも 2 MB、最大 20 GB 含まれている必要があります。データソース内の 10 MB を超えるファイルは無視されます。
- データソース内のファイル数に制限はありませんが、カスタマイズでサポートする言語ごとに少なくとも 10 個のファイルを含める必要があります。
- ファイル名と個々のディレクトリ名は 255 文字以下にする必要があります。(累積で 255 文字を超える場合があります)。これらの制限を超えると、カスタマイズの作成が失敗します。
- Amazon S3 データ内
  source、すべてのソースコードはルートではなくディレクトリに配置する必要があります
  レベル。ルートレベルのファイルは無視されます。

> **Note**
>
> カスタマイズでサポートされている言語と、カスタマイズの作成に使用されるファイルタイプについては、「」を参照してくださいカスタマイズの言語サポート。

カスタマイズの作成に関する情報は、Amazon CloudWatch Logs に保存できます。詳細については、「Amazon CloudWatch Logs でのカスタマイズ関連のメッセージへのアクセス」を参照してください。

カスタマイズの作成

このセクションでは、Amazon Q でカスタマイズを作成する方法について説明します。

カスタマイズを作成する手順は次のとおりです。

1. 「」の手順に従って、Amazon Q Developer Pro にユーザーをサブスクライブしますIAM Identity Center の開始方法。サブスクリプションプロセスの一環として、カスタマイズの作成に必要な Amazon Q Developer プロファイルをインストールします。
2. AWS Management Consoleにサインインします。
3. Amazon Q Developer コンソールに切り替えます。
4. 左のナビゲーションペインから、[カスタマイズ] を選択します。
5. カスタマイズのページが表示されます。

6. [カスタマイズの作成] を選択します。
7. カスタマイズの名前とオプションで説明を入力します。

> **Note**
>
> 名前と説明のどちらもデベロッパーにとって役に立つものにします。Amazon Q Developer Pro の使用が許可されている組織の開発者は、 AWS プラグインを通じて IDE でそれらを表示できます。

データソースへの接続

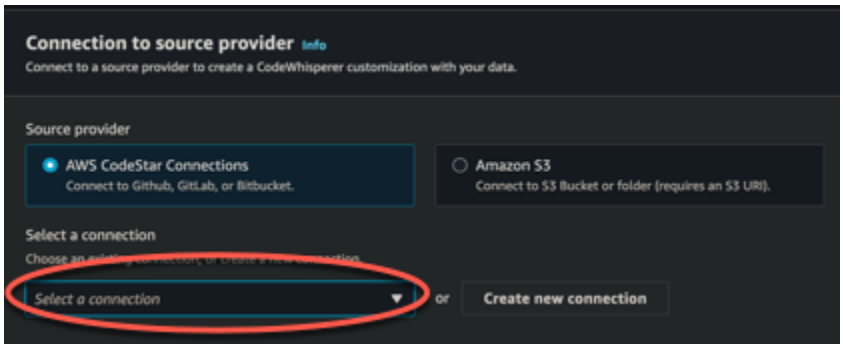
カスタマイズを作成する前に、コードベースを含むデータソースに接続する必要があります。これを行う方法は、データソースがどこにあるかによって異なります。

データソースが Github、GitLab、または Bitbucket にある場合は、 を使用してデータソースに接続できます AWS CodeConnections。それ以外の場合は、Amazon S3 バケット内のフォルダにデータを配置します。

CodeConnections の詳細については、「 デベロッパーツールコンソールユーザーガイド」の「接続とは」を参照してください。

CodeConnections を介してデータソースに接続するには

1. ソースプロバイダーへの接続で、AWS CodeStar CodeConnections を選択します。
2. 既存の接続を使用する場合は、接続の選択を選択します。



次に、「リポジトリ選択の選択」で、次のいずれかを実行します。

- 接続内のすべてのリポジトリを使用してカスタマイズを生成するには、この接続のすべてのリポジトリを使用するを選択します。

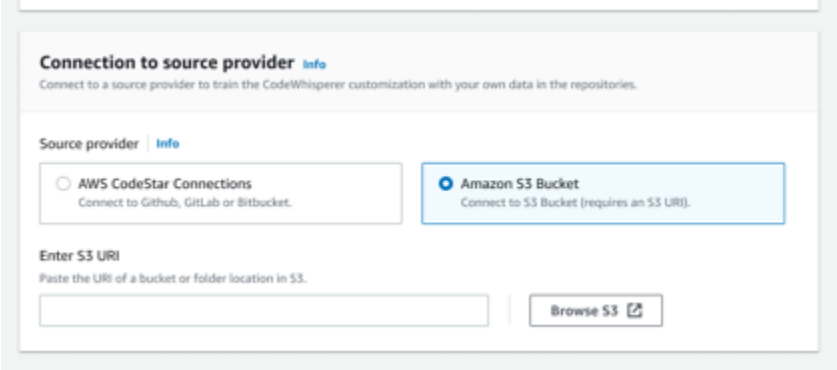
- カスタマイズを生成する特定のリポジトリを選択するには、特定のリポジトリを選択を選択し、リポジトリを選択を選択します。ポップアップウィンドウで、使用するリポジトリを見つけ、追加を選択します。

> **Note**
>
> カスタマイズに含めることができるリポジトリの数に制限はありませんが、個別に選択すると 100 に制限されます。100 を超えるリポジトリを使用する場合は、「すべてのリポジトリを使用する」オプションを選択するか、Amazon S3 にリポジトリを配置し、Amazon S3 経由でデータソースを接続する手順に従います。

3. 新しい接続を作成する場合は、新しい接続を作成し、この手順の残りのステップに従います。
4. ポップアップウィンドウでデータソースに移動し、コンソールの指示に従います。
5. データソースを作成したら、[カスタマイズの作成] ページに戻ります。
6. [接続の選択] で、ドロップダウンから接続を選択します。

Amazon S3 を介してデータソースに接続するには

1. [ソースプロバイダーへの接続] で、[Amazon S3] を選択します。



2. [Amazon S3 を参照] を選択します。
3. コードベースに移動し、URI をメモします。コードベースは、バケットのルートではなく、Amazon S3 バケット内のフォルダにある必要があります。

   詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 バケットの作成、設定、操作」および「Amazon S3 でのアクセスコントロール」を参照してください。
4. URL を[Amazon S3 URI を入力] というラベルが付いたフィールドに貼り付けます。

カスタマイズを作成する前に、タグを追加するオプションがあります。

タグの詳細については、AWS リソースのタグ付けに関するユーザーガイドを参照してください。

上記の手順に従って、[カスタマイズの作成] を選択します。

カスタマイズとデータ

Amazon Q のカスタマイズでは、お客様のコンテンツを使用して、お客様の組織のデベロッパーのスタイルで提案を表示します。

ただし、 AWS は、エンタープライズに直接サービスを提供しないコンテキストではコンテンツを保存または使用しません。

AWS は、お客様のコンテンツを使用して他のお客様にコード提案を行いません。

Amazon Q は、他のお客様 (またはお客様) のコードレビューを参照しません。

詳細については、「Amazon Q Developer のサービスの改善」を参照してください。

カスタマイズの作成のトラブルシューティング

- エラー `Total size of the provided repositories exceeds the maximum allowed size of number for a customization.` が発生した場合:

  その場合は、データソースからリポジトリを削除し、もう一度試してください。

- エラー `Insufficient data to create a customization. Add more files from supported languages and retry.` が発生した場合:

  特定の言語で書かれたコードを使用してカスタマイズを作成するには
  、データソースにその言語のコードを含むファイルが少なくとも 10 個必要です。データソースには少なくとも 2 MB を含める必要がありますが、サポートされている言語のソースコードファイルは 20 MB 近く、最大 20 GB にすることをお勧めします。

  一部のファイルは、関連する言語であってもカウントされません。
  2 MB に。例えば、重複ファイルやサポートされていない形式のファイルはカウントされません。

  このエラーが表示された場合は、カスタマイズの焦点であるプログラミング言語を含むファイルを追加し、もう一度試してください。

- エラー `Encountered an issue when retrieving some of the selected repositories from CodeConnections. Check the customization's log deliveries for details.` が発生した場合:

  このエラーが表示された場合は、接続がアクセスできる有効なリポジトリを使用して、カスタマイズの作成または更新を再試行してください。

カスタマイズの削除

このセクションでは、Amazon Q でカスタマイズを削除する方法について説明します。

> **Warning**
> カスタマイズを削除すると、リソースに関連付けられているすべてのバージョンが削除されます。

カスタマイズを削除する手順は次のとおりです。

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. 左のナビゲーションペインから、[カスタマイズ] を選択します。
4. カスタマイズのページが表示されます。
5. 削除するカスタマイズがまだアクティブな場合は、[非アクティブ化] を選択します。
6. [削除] を選択します。

> **Note**
> カスタマイズの詳細を表示するページからカスタマイズを削除することもできます。これを行うには、カスタマイズの詳細ページの右上隅から [削除] を選択します。

カスタマイズの最適化

このページでは、カスタマイズを最適化する方法について説明します。

カスタマイズの最適化

このセクションには、提案を最適化するための提案が含まれています。

- データソースを拡張し、含まれるコードリポジトリを増やすことを検討する。
- 限られたプログラミング言語のデータを主に含めた場合は、言語を増やすための拡張を検討してください。
- 自動生成されたファイルとリポジトリ、またはテンプレートから生成されたファイルとリポジトリを削除する。このようなファイルを含めることは通常重要ではなく、ノイズを追加する傾向があります。
- コードベースが内部ライブラリを頻繁に使用していない可能性がある。これが事実であることがわかっている場合は、コア Amazon Q モデルが既に実行されている可能性があります。

使用する言語に合わせて最適化する

特定の言語のコードをカスタマイズで使用するには、その言語を含むデータファイルを少なくとも 20 個含め、すべてのソースファイルをまとめて 2 MB 以上にする必要があります。カスタマイズでサポートされていない言語でコードを記述する場合、その言語での Amazon Q からの提案は Amazon Q ベースモデル (カスタマイズではない) を基に提供されます。つまり、カスタマイズがない場合に提供されるのと同じ提案になります。これは、ダッシュボードのメトリクスに影響する可能性があります。例えば、「Amazon Q によって生成されたコードのライン」は、デベロッパーがよく使用する言語がカスタマイズに含まれていた場合に比べて少なくなる可能性があります。

ログ記録とトラブルシューティング

ログ配信の設定

Amazon Q は、カスタマイズに関する問題の理解とトラブルシューティングに役立つログファイルを提供します。

ログファイルは、Amazon CloudWatch Logs グループ、Amazon S3 バケット、Amazon Data Firehose、または任意の組み合わせに送信できます。

ログ配信を設定するには、コンソールページの [ログ配信] タブを選択してカスタマイズします。インターフェイスの指示に従って、ログ配信を設定します。その後、[ログ配信を作成] を選択します。

Amazon S3 バケットに配信されるログのプレフィックスは AWSLogs/*account_id*/codeWhispererCustomizationLogs/*region*/*customization_id*/*year*/*month*/*day*/*hour*/ です。

ファイルは zip 圧縮され、*account_id*_codeWhispererCustomizationLogs_*customization_id*_*date*_*file_id*.log.g という命名形式で保存されます。

> **Warning**
>
> カスタマイズログを最大限に活用するには、カスタマイズの作成から 5 分以内にログ配信を設定することをお勧めします。

複数のリソースにログを配信するために必要なアクセス許可の詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「追加のアクセス許可が必要なロギング [V2]」を参照してください。

カスタマイズ関連のログメッセージを理解する

次の表は、カスタマイズに関する問題を理解するのに役立つログメッセージをまとめています。

ログメッセージ	ログレベル
`Starting to ingest number repos from source source`	情報
`Downloading data from repo: repo name`	情報
`Received amount MB of supported data. amount MB required. Add more data and retry.`	エラー
`The provided CodeStar Connection ARN: Arn is invalid.`	エラー

ログメッセージ	ログレベル
Access denied when attempting to reach the provided CodeStar Connection: *Arn*	エラー
Failed to download with AWS CodeStar Connection: *Arn* probably deleted by customer	エラー
ProviderThrottlingException from CodeStar Connection: *Arn* while cloning repository: *repository*	エラー
Processing data from S3: *S3 URI*	情報
Invalid S3 path specified: *S3 Directory*	エラー
Unable to access the provided S3 bucket: *bucket name*	エラー
The provided S3 bucket: *bucket name* does not exist.	エラー
The provided S3 key *S3 URI* does not exist.	エラー
Failed to ingest *number of failed repos / total number of repos* repositories	エラー
Unable to process repository: *repo name*, with a size of *repo size* GB, exceeds the limit of *max size* GB.	警告

ログメッセージ	ログレベル
`Unable to process file: file name, with a size of file size, which exceeds the limit of max file size MB`	エラー
`Unable to process collection: collection name , with total size of total repo size MB, which exceeds the limit of max total repo size MB`	エラー
`The following languages will be used for customization: list of languages . Languages may be excluded from customization if they are not sufficiently represented in your files.`	情報

コンソールでのカスタマイズ関連のエラーメッセージを理解する

次の表は、Amazon Q コンソールのカスタマイズ関連のメッセージを理解するのに役立ちます。

エラーメッセージ	推奨されるアクション
You have activated the maximum number of customizations.	アクティブなカスタマイズを無効にして、もう一度試す。
You have exceeded the maximum number of group permissions limit of *limit*.	グループを削除して再試行する。
You have exceeded the maximum number of user permissions limit of *limit*.	ユーザーを削除して再試行する。
Maximum active jobs reached.	同じアカウントの進行中のジョブが完了するまで待ってから、オペレーションを再試行する。

エラーメッセージ	推奨されるアクション
Encountered an unexpected error when processing the request.	オペレーションを再試行する。引き続き失敗する場合は、カスタマーサポートに問い合わせる。
CodeConnections から選択したリポジトリの一部を取得するときに問題が発生しました。詳細については、カスタマイズの口グ配信を確認してください。	接続がアクセスできる有効なリポジトリを使用して、カスタマイズを再度作成または更新してみてください。
Access denied when attempting to reach the provided AWS CodeConnections connection.	接続とサードパーティープロバイダーに対するアクセス許可を確認してから、オペレーションを再試行する。
One or more repositories not found while accessing the provided AWS CodeConnections connection.	サードパーティープロバイダーからのアクセス許可とリポジトリのリストを確認してから、オペレーションを再試行する。
指定された AWS CodeConnections 接続 ARN が無効です。	正しい接続 ARN を使用してカスタマイズを更新する。
提供された AWS CodeConnections 接続に関連付けられているホストは利用できません。	5 分後にもう一度試す。
Invalid Amazon S3 path specified.	有効な Amazon S3 URI を使用してカスタマイズを更新する。
Unable to access the provided Amazon S3 bucket.	管理者ロールのアクセス許可を確認し、アクセス許可の問題を修正した後に再試行する。
The provided Amazon S3 bucket does not exist.	有効な Amazon S3 URI を使用してカスタマイズを更新する。
The provided Amazon S3 key does not exist.	有効な Amazon S3 URI を使用してカスタマイズを更新する。

エラーメッセージ	推奨されるアクション
Insufficient data to create a customization. Add more files from supported languages and retry.	同じデータソースにデータを追加し、同じリファレンスを使用してカスタマイズを更新する。
Total size of the provided repositories exceeds the maximum allowed size of *size* for a customization.	指定されたデータソースから一部のデータを削除し、同じリファレンスを使用してカスタマイズを更新する。
You have created the maximum number of customizations. Delete an existing customization and try again.	現在のカスタマイズを削除して再試行する。
Customizations exist within the account. You must delete all customizations prior to deleting the profile.	アカウントに関連付けられたすべてのカスタマイズを削除し、再試行する。

Amazon Q カスタマイズのアクティブ化

バージョンのアクティブ化

このセクションでは、カスタマイズのバージョンをアクティブ化および非アクティブ化する方法について説明します。

組織のデベロッパーが以前のバージョンを使用している場合でも、新しいバージョンのカスタマイズをアクティブ化できます。新しいバージョンをアクティブ化すると、デベロッパーはシームレスに使用を開始できます。開発側で調整する必要はありません。

また、カスタマイズを以前のアクティブな状態にロールバックすることもできます。ただし、以前にアクティブ化されたバージョンが再びアクティブ化されるわけではありません。代わりに、以前のバージョンをコピーしてからコピーをアクティブ化することで、新しいバージョンが作成されます。

例えば、1、2、3 の 3 つのバージョンがあるとします。アクティブなバージョンは 3 です。バージョン 1 に戻ることにしました。しかし、バージョン 1 の「再アクティブ化」は、実際にはバージョン 1 をコピーし、バージョン 4 を作成しています。古いバージョンの新しいコピーであるバージョン 4 を使用することになります。

カスタマイズのバージョンをアクティブ化する手順は次のとおりです。

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. 左のナビゲーションペインから、[カスタマイズ] を選択します。

   カスタマイズのページが表示されます。
4. バージョンをアクティブ化するカスタマイズを選択します。

   カスタマイズの詳細ページが表示されます。
5. [バージョン] テーブルからアクティブ化するバージョンを選択します。
6. [アクティブ化] を選択します。

カスタマイズを非アクティブ化するには、ドロップダウンから [非アクティブ化] を選択します。

Amazon Q のカスタマイズの更新

カスタマイズは、作成時にデータソースのスナップショットに基づいて作成されます。次の場合は、Amazon Q のカスタマイズを更新できます。

- データソース内のファイルを更新し、新しいファイルでカスタマイズを再作成します。
- データソースを から Amazon S3 AWS CodeConnections に切り替えるか、逆に切り替えます。
- CodeConnections データソースで参照されるリポジトリを変更します。

カスタマイズには複数のバージョンを持つことができます。

Amazon Q 管理者は、カスタマイズごとに最大 3 つのバージョンにアクセスできます。

- 最新バージョン
- 現在アクティブなバージョン
- 現在アクティブではない最新のアクティブバージョン

新しいバージョンを作成する

カスタマイズの新しいバージョンを作成する手順は次のとおりです。

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。

3. 左のナビゲーションペインから、[カスタマイズ] を選択します。

   カスタマイズのページが表示されます。

4. 新しいバージョンを作成するカスタマイズを選択します。

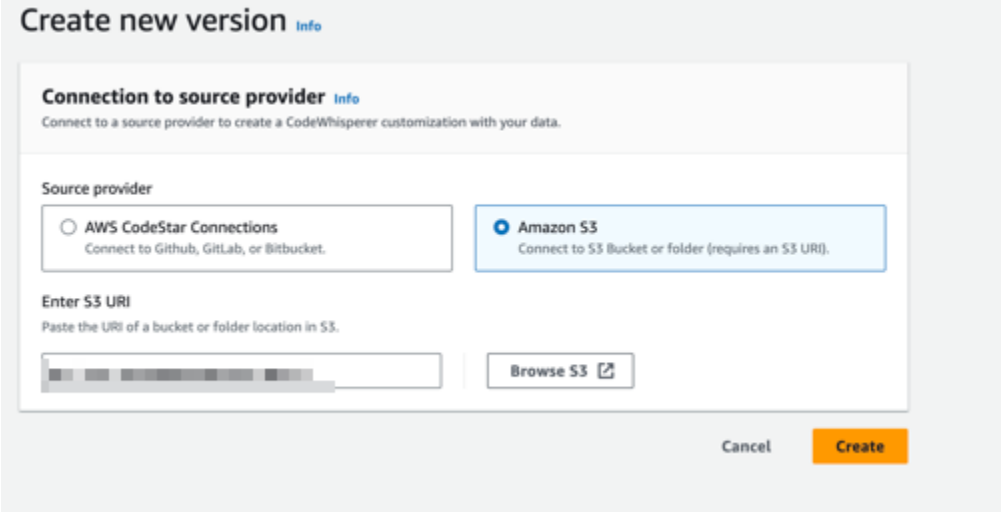
   カスタマイズの詳細ページが表示されます。

5. 次のいずれかを行います：

   - [アクション] ドロップダウンで、[新しいバージョンの作成] を選択します。
   - Sources タブを選択し、Update を選択します。

   カスタマイズの更新ページが表示されます。

6. [アクション] ドロップダウンで、[新しいバージョンの作成] を選択します。
7. （オプション) データソースを変更します。



8. （オプション) CodeConnections データソースを選択した場合は、接続に関連付けられたリポジトリを変更します。
9. [作成] を選択します。

エラーメッセージが表示された場合は、「カスタマイズの作成のトラブルシューティング」を参照してください。

Amazon Q のカスタマイズにユーザーとグループを追加する

このセクションでは、カスタマイズにユーザーとグループを追加する方法について説明します。

> **Note**
>
> ユーザーを追加する前に、カスタマイズを有効にする必要があります。

> **Note**
>
> ユーザーまたはグループをカスタマイズに追加できるのは、IAM Identity Center ワークフォースユーザーとして Amazon Q Developer Pro にサブスクライブしている場合のみです。詳細については、「ステップ 2: ワークフォースユーザーを Amazon Q Developer Pro にサブスクライブする」を参照してください。

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. ナビゲーションパネルから、カスタマイズを選択します。
4. ユーザーまたはグループを追加するカスタマイズの名前を選択します。
5. ウィンドウの下半分で、必要に応じて、ユーザーとグループタブを選択します。次に、[ユーザー] または [グループ] のサブタブを選択します。
6. カスタマイズへのアクセスを必要とするユーザーまたはグループを選択します。
7. [ユーザーの追加] または [グループの追加] を選択します。

Amazon Q のカスタマイズの使用

このセクションでは、デベロッパーとしてカスタマイズを使用する方法について説明します。

> **Note**
>
> Amazon Q Developer の以下の機能では、カスタマイズがサポートされています。
>
> - インライン提案 – 「インライン提案の生成」を参照してください。
> - IDE でのチャット – 「コードに関するチャット」を参照してください。

Visual Studio Code

でカスタマイズを使用するにはVisual Studio Code：

1. 「」のステップを使用して、IAM アイデンティティセンターで Amazon Q Developer Pro に認証しますIDE に Amazon Q Developer 拡張機能またはプラグインをインストールする。
2. Amazon Q の [デベロッパーツール] ペインで、[カスタマイズの選択] を選択します。
3. ウィンドウの上部にあるドロップダウンメニューから、適切なカスタマイズを選択します。

JetBrains

IDEs JetBrains でカスタマイズを使用するには：

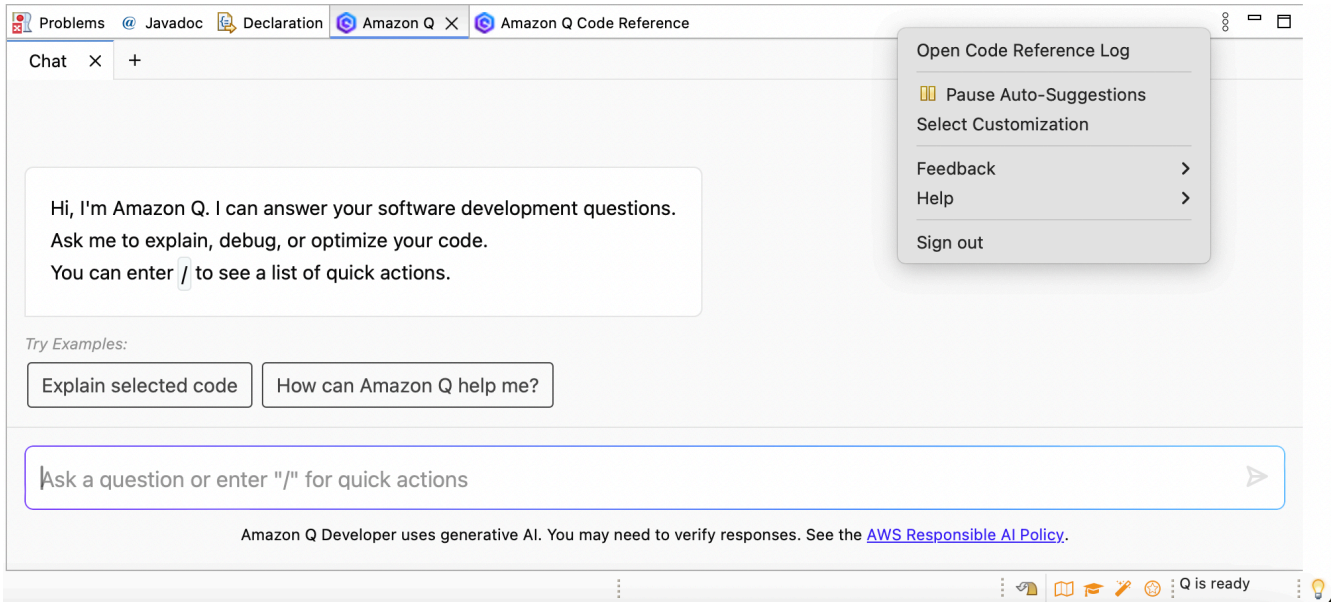
1. 「」のステップを使用して、IAM Identity Center で Amazon Q Developer Pro を認証しますIDE に Amazon Q Developer 拡張機能またはプラグインをインストールする。
2. Amazon Q の [デベロッパーツール] ペインで、[カスタマイズの選択] を選択します。
3. ポップアップウィンドウで、適切なカスタマイズを選択します。
4. [接続] を選択します。

Eclipse

IDEs Eclipse でカスタマイズを使用するには：

1. 「」のステップを使用して、IAM Identity Center で Amazon Q Developer Pro を認証しますIDE に Amazon Q Developer 拡張機能またはプラグインをインストールする。
2. IDE で、IDE Eclipse の右上隅にある Amazon Q アイコンを選択します。
3. Amazon Q チャットタブを開いた状態で、タブの右上隅にある省略記号アイコンを選択します。Amazon Q タスクバーが開きます。

   次の図は、IDE の Amazon Q Eclipse タスクバーを示しています。



4. カスタマイズの選択 を選択します。
5. ポップアップウィンドウで、適切なカスタマイズを選択します。
6. [選択] を選びます。

Amazon Q Developer のセキュリティ

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS 、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、 AWS お客様とお客様の間の責任共有です。責任共有モデルでは、これをクラウドのセキュリティおよびクラウド内のセキュリティとして説明しています。

- クラウドのセキュリティ – AWS は、 AWS のサービス で実行されるインフラストラクチャを保護する責任を担います AWS クラウド。 は、お客様が安全に使用できるサービス AWS も提供します。サードパーティーの監査人は、AWS コンプライアンスプログラム の一環として、セキュリティの有効性を定期的にテストおよび検証します。Amazon Q に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスプログラム対象範囲内のAWS サービス」を参照してください。
- クラウドのセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、Amazon Q Developer の使用時に責任共有モデルを適用する方法を理解するのに役立ちます。ここでは、セキュリティやコンプライアンスに関する目標を達成できるように Amazon Q を設定する方法について説明します。また、Amazon Q リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

トピック

- Amazon Q Developer でのデータ保護
- Amazon Q Developer における ID 管理とアクセス管理
- Amazon Q Developer のコンプライアンス検証
- Amazon Q Developer の耐障害性
- Amazon Q Developer のインフラストラクチャセキュリティ
- Amazon Q Developer のファイアウォール、プロキシサーバー、またはデータ境界の設定
- Amazon Q Developer とインターフェイスエンドポイント (AWS PrivateLink)

Amazon Q Developer でのデータ保護

Amazon Q Developer でのデータ保護に AWS 責任共有モデルがどのように適用されるかを学びます。このモデルで説明されているように、 AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、 AWS のサービス 使用する のセキュリティ設定および管理タスクについても責任を負います。データプライバシーの詳細については、「データプライバシーのよくある質問」を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された「AWS 責任共有モデルおよび GDPR」ブログを参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS Identity and Access Management () を使用して個々のユーザーを設定することをお勧めしますIAM。この方法により、それぞれのジョブを遂行するために必要な許可のみを各ユーザーに付与できます。また、次の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 以降が推奨されます。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- などの高度なマネージドセキュリティサービスを使用して Amazon Macie、 に保存されている機密データの検出と保護を支援します Amazon S3。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-2 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「連邦情報処理規格 (FIPS) 140-2」を参照してください。

顧客の E メールアドレスなどの機密情報や機密情報をタグや名前フィールドなどの自由形式のテキストフィールドに入力しないことを強くお勧めします。これは、、API AWS Management Console、 AWS Command Line Interface （ AWS CLI ）、または AWS SDK AWS のサービス を使用して Amazon Q または他の を操作する場合も同様です。 SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。Amazon Q Developer がコンテンツを使用する方法の詳細については、「」を参照してくださいAmazon Q Developer のサービスの改善。

トピック

- Amazon Q Developer のデータストレージ
- Amazon Q Developer でのデータ暗号化
- Amazon Q Developer のサービスの改善
- IDE とコマンドラインでデータ共有をオプトアウトする
- Amazon Q Developer でのクロスリージョン処理

Amazon Q Developer のデータストレージ

Amazon Q は、質問、そのレスポンス、コンソールメタデータやコードなどの追加のコンテキストを保存して、質問やリクエストに対するレスポンスを生成します。データの暗号化方法については、「」を参照してくださいAmazon Q Developer でのデータ暗号化。 AWS が Amazon Q に尋ねるいくつかの質問をどのように使用するか、およびその回答を使用してサービスを改善する方法については、「」を参照してくださいAmazon Q Developer のサービスの改善。

AWS コンテンツが処理および保存されるリージョン

IAM Identity Center ワークフォースユーザーの場合、Amazon Q Developer Pro 階層では、コンテンツは Amazon Q Developer プロファイルが作成された AWS リージョン に以下の機能でのみ保存されます。

- の「Amazon Q チャット AWS Management Console」
- Amazon Q による AWS コンソールエラーの診断
- Eclipse、IDEs、Visual Studio CodeVisual Studio JetBrains の Amazon Q
- コマンドライン上の Amazon Q

Amazon Q Developer Pro 階層で他の機能を使用する場合、コンテンツは米国リージョンに保存および処理される場合があります。米国以外のリージョンで Q Developer プロファイルを使用している場合は、米国でコンテンツを保存して推論を実行する機能へのアクセスをブロックするサービスコントロールポリシー (SCP) を作成できます。SCP の例については、「ポリシーを使用して Amazon Q Developer へのアクセスを管理する」を参照してください。

その他の Amazon Q の機能と統合、および Amazon Q Developer 無料利用枠を使用する場合、コンテンツは米国のリージョンに保存されます。コンソールエラーセッションの診断中に処理されたデータは、米国西部 (オレゴン) リージョンに保存されます。他のすべてのデータは、米国東部 (バージニア北部) リージョンに保存されます。データを異なる方法で保存する以下の機能に注意してください。

> **Note**
>
> Amazon Q で Console to Code を使用すると、コンテンツはコンソールリージョンに保存され、米国リージョンで処理されます。
> Amazon Redshift で Amazon Q 生成 SQL を使用すると、コンテンツはコンソールリージョンに保存され、処理されます。詳細については、「Amazon Redshift 管理ガイド」の「Amazon Q 生成 SQL の操作」を参照してください。
> Amazon CloudWatch の調査を使用して調査を作成すると、コンテンツが他のリージョンに保存され、処理される可能性があります。詳細については、「Amazon CloudWatch ユーザーガイド」の「Security in CloudWatch investigation」トピックを参照してください。
> Amazon CloudWatch

クロスリージョン推論では、Amazon Q Developer へのリクエストは、コンテンツが保存されている地域内の別のリージョンで処理される場合があります。詳細については、「クロスリージョン推論」を参照してください。

Amazon Q Developer でのデータ暗号化

このトピックでは、転送中の暗号化と保管中の暗号化について Amazon Q Developer に固有の情報を提供します。

転送中の暗号化

お客様と Amazon Q 間、および Amazon Q とそのダウンストリームの依存関係間のすべての通信は、TLS 1.2 以降の接続を使用して保護されます。

保管中の暗号化

Amazon Q は保管中のデータの保存に Amazon DynamoDB と Amazon Simple Storage Service (Amazon S3)を使用します。保管中のデータは、デフォルトで暗号化ソリューションを使用して AWS 暗号化されます。Amazon Q は、 AWS Key Management Service （ AWS KMS) AWS が所有する暗号化キーを使用してデータを暗号化します。データを暗号化する AWS マネージドキーを保護するためのアクションを実行する必要はありません。詳細については、AWS Key Management Service デベロッパーガイドの「AWS 所有キー」を参照してください。

Amazon Q Developer Pro をサブスクライブしている IAM Identity Center ワークフォースユーザーの場合、管理者は以下の機能について、保管中のデータのカスタマーマネージド KMS キーによる暗号化を設定できます。

- AWS コンソールでチャットする
- AWS コンソールエラーの診断
- カスタマイズ
- IDE のエージェント

AWS コンソールと IDE では、Amazon Q のリストされた機能のカスタマーマネージドキーでのみデータを暗号化できます。ウェブサイト、 AWS Documentation ページ、チャットアプリケーションでの Amazon Q AWS との会話は、 AWS所有キーでのみ暗号化されます。

カスタマーマネージドキーは、KMS キーへのアクセスを制御することでデータへのアクセスを直接制御するために作成、所有、管理する AWS アカウントの KMS キーです。対称キーのみがサポートされます。独自の KMS キーの作成については、AWS Key Management Service デベロッパーガイドの「キーの作成」を参照してください。 AWS Key Management Service

カスタマーマネージドキーを使用する場合、Amazon Q Developer は KMS 許可を使用し、認可されたユーザー、ロール、またはアプリケーションが KMS キーを使用できるようにします。Amazon Q Developer 管理者が設定中に暗号化にカスタマーマネージドキーを使用することを選択すると、その権限が作成されます。この権限により、エンドユーザーは保管中のデータ暗号化に暗号化キーを使用できます。グラントの詳細については、「」の「グラント AWS KMS」を参照してください。

AWS コンソールで Amazon Q とのチャットの暗号化に使用する KMS キーを変更する場合は、新しい会話を開始して、新しいキーを使用してデータを暗号化する必要があります。前のキーで暗号化された会話は保持されず、今後の会話のみが更新されたキーで暗号化されます。以前の暗号化方法からの会話を維持する場合は、それらの会話中に使用していたキーに戻すことができます。コンソールエラーセッションの診断の暗号化に使用される KMS キーを変更する場合は、新しい診断セッションを開始して、新しいキーを使用してデータを暗号化する必要があります。

カスタマーマネージド KMS キーの使用

カスタマーマネージド KMS キーを作成した後、Amazon Q Developer 管理者は Amazon Q Developer コンソールでキーを指定して、それを使用してデータを暗号化する必要があります。Amazon Q Developer コンソールでキーを追加する方法については、「」を参照してくださいAmazon Q Developer での暗号化方法の管理。

Amazon Q Developer でデータを暗号化するようにカスタマーマネージドキーを設定するには、管理者が使用するアクセス許可が必要です AWS KMS。必要な KMS アクセス許可は、IAM ポリシーの例に含まれています管理者に Amazon Q Developer コンソールの使用を許可する。

カスタマーマネージドキーで暗号化された機能を使用するには、Amazon Q がカスタマーマネージドキーにアクセスできるようにするアクセス許可が必要です。必要なアクセス許可を付与するポリシーについては、「」を参照してくださいAmazon Q にカスタマーマネージドキーへのアクセスを許可する。

Amazon Q Developer の使用中に KMS 許可に関連するエラーが表示された場合は、Amazon Q が許可を作成できるように許可を更新する必要があります。必要なアクセス許可を自動的に設定するには、Amazon Q Developer コンソールに移動し、ページ上部のバナーでアクセス許可の更新を選択します。

Amazon Q Developer のサービスの改善

Amazon Q Developer が関連性の高い情報を提供するために、AWS は Amazon Q からの特定のコンテンツを使用する場合があります。これには、Amazon Q へのお客様の質問とその回答などがあり、サービスの向上に役立てられます。このページでは、AWS がどのようなコンテントを使用するのか、またオプトアウトの方法について説明します。

サービスの改善に使用される Amazon Q Developer 無料利用枠コンテンツ

AWS では、Amazon Q Developer 無料利用枠の特定のコンテンツをサービスの改善に使用する場合があります。Amazon Q は、よくある質問への回答の改善、Amazon Q の運用上の問題の修正、デバッグ、モデルトレーニングなどのために、このコンテンツを使用する場合があります。

がサービスの改善に使用する AWS 可能性のあるコンテンツには、Amazon Q への質問や、Amazon Q が生成するレスポンスとコードが含まれます。

Amazon Q Developer Pro または Amazon Q Business のコンテンツは、サービスの改善には使用されません。

> **Note**
>
> Amazon Q Developer for GitHub (プレビュー) は現在、サービスの改善のためにコンテンツを使用しません。今後これを有効にすると、適切な通知と使用をオプトアウトする方法が提供されます。

オプトアウトの方法

Amazon Q Developer 無料利用枠でサービス改善のためのコンテンツ使用からオプトアウトする方法は、Amazon Q を利用する環境によって異なります。

、 AWS Management Console AWS Console Mobile Application、 AWS ウェブサイト、チャットアプリケーションで、AI サービスのオプトアウトポリシーを設定します AWS Organizations。詳細については、「AWS Organizations ユーザーガイド」の「AI サービスオプトアウトポリシー」を参照してください。

IDE で Amazon Q Developer 無料利用枠を使用している場合は、IDE の設定を調整します。詳細については、「IDE とコマンドラインでデータ共有をオプトアウトする」を参照してください。

IDE とコマンドラインでデータ共有をオプトアウトする

このページでは、サードパーティーの IDE やコーディング環境など、Amazon Q AWS を使用する IDEs またはコマンドラインでデータの共有をオプトアウトする方法について説明します。Amazon Q がこのデータを使用する方法については、「Amazon Q Developer のサービスの改善」を参照してください。

クライアント側のテレメトリの共有からオプトアウトする

クライアント側のテレメトリは、サービスの使用状況を数値で表したものです。例えば、 AWS は、提案が受け入れられるか拒否されるかを追跡する可能性があります。クライアント側のテレメトリに実際のコードが含まれていません。

IDEs で収集されたテレメトリ

IDE で Amazon Q によって収集されたテレメトリデータの詳細については、`aws-toolkit-common`Github リポジトリの commonDefinitions.json ドキュメントを参照してください。

Amazon Q を使用する各 IDE によって収集されたテレメトリデータの詳細については、次の GitHub リポジトリのリソースドキュメントを参照してください。

- VS Code の Amazon Q 拡張機能
- JetBrains 用 Amazon Q プラグイン
- Eclipse 用 Amazon Q プラグイン
- AWS Amazon Q を使用した Visual Studio Toolkit

Q CLI で収集されたテレメトリ

Q CLI によって収集されたテレメトリデータの詳細については、`amazon-q-developer-cli`Github リポジトリのテレメトリ_definitions.json ドキュメントを参照してください。

変換用にコマンドラインツールで収集されたテレメトリ

テレメトリコレクションは、Q コマンドライン変換ツールのパフォーマンス AWS を理解し、機能の使用方法を理解し、サービスを改善するのに役立ちます。コマンドラインでの変換の場合、ツールバージョンと Maven プラグインバージョンでテレメトリが収集されます。

> **Note**
>
> 個人を特定できる情報 (PII) やその他の機密情報や機密情報をフリーテキストフィールドに追加しないでください。

IDE を選択すると、クライアント側のテレメトリの共有をオプトアウトする手順が表示されます。

Visual Studio Code

VS Code でテレメトリデータの共有からオプトアウトする手順は次のとおりです。

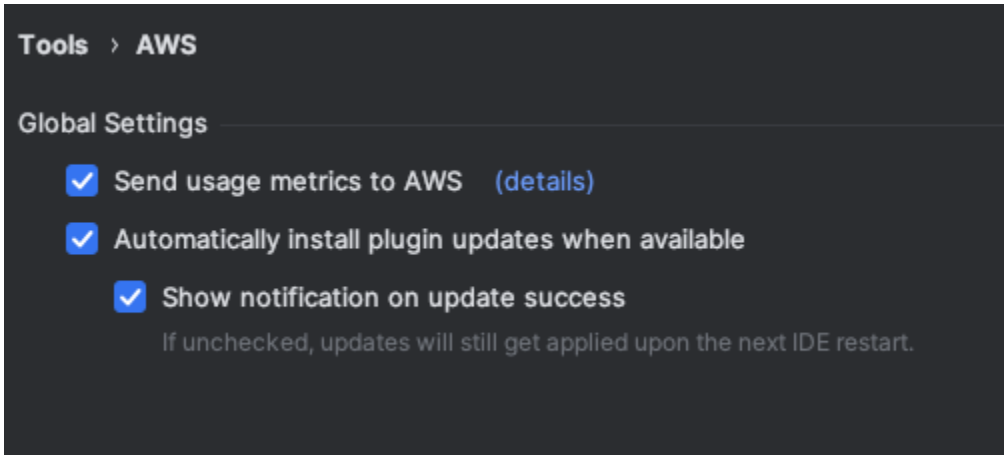
1. VS Code で [設定] を開きます。
2. VS Code ワークスペースを使用している場合は、[ワークスペース] サブタブに切り替えます。VS Code では、ワークスペース設定がユーザー設定より優先されます。
3. [検索] バーに「`Amazon Q: Telemetry`」と入力します。
4. チェックボックスをオフにします。

> **Note**
>
> これは、デベロッパーが各自の IDE 内で行う必要があります。エンタープライズの一部として Amazon Q を使用している場合、管理者がこの設定を変更することはできません。

JetBrains

JetBrains でテレメトリデータの共有からオプトアウトする手順は次のとおりです。

1. JetBrains IDE で、[Preferences] を開きます (Mac では、これは [Settings] の下になります)。
2. 左側のナビゲーションバーで、[Tools] を選択し、[AWS] を選択します。
3. 使用状況メトリクスの送信の選択を解除します AWS。



> **Note**
>
> これは、デベロッパーが各自の IDE 内で行う必要があります。エンタープライズの一部として Amazon Q を使用している場合、管理者がこの設定を変更することはできません。

Eclipse

Eclipse IDEs でのテレメトリデータの共有をオプトアウトするには、次の手順を使用します。

1. IDE Eclipse で設定を開きます。
2. 左側のナビゲーションバーから Amazon Q を選択します。
3. 使用状況メトリクスを に送信するの横にあるチェックボックスをオフにします AWS。
4. [Apply (適用)] を選択して変更を保存します。

> **Note**
>
> これは、デベロッパーが各自の IDE 内で行う必要があります。エンタープライズの一部として Amazon Q を使用している場合、管理者がこの設定を変更することはできません。

Visual Studio

AWS Toolkit for でのテレメトリデータの共有をオプトアウトするにはVisual Studio、次の手順を使用します。

1. [ツール] で [オプション] を選択します。
2. [オプション] ペインで、[AWS Toolkit] を選択し、[全般] を選択します。
3. AWS ツールキットによる使用状況情報の収集を許可するの選択を解除します。

> **Note**
>
> これは、デベロッパーが各自の IDE 内で行う必要があります。エンタープライズの一部として Amazon Q を使用している場合、管理者がこの設定を変更することはできません。

AWS Cloud9

1. IDE AWS Cloud9 内から、ウィンドウの上部にある AWS Cloud9 ロゴを選択し、設定を選択します。
2. [環境設定] タブで、[AWS Toolkit] を選択します。
3. [AWS: クライアント側のテレメトリ] の横で、スイッチをオフに切り替えます。

> **Note**
>
> この設定は、Amazon Q だけでなく、 AWS Cloud9 クライアント側のテレメトリ全般を共有するかどうかに影響します。

Lambda

Lambda で Amazon Q を使用する場合、Amazon Q はクライアント側のテレメトリを AWSと共有しません。

SageMaker AI Studio

1. SageMaker AI Studio ウィンドウの上部から設定を選択します。
2. [設定] ドロップダウンから、[詳細設定エディタ] を選択します。
3. Amazon Q のドロップダウンで、[Amazon Q と使用状況データを共有する] の横にあるチェックボックスをオンまたはオフにします。

JupyterLab

1. JupyterLab ウィンドウの上部から、[設定] を選択します。
2. [設定] ドロップダウンから、[詳細設定エディタ] を選択します。
3. Amazon Q のドロップダウンで、[Amazon Q と使用状況データを共有する] の横にあるチェックボックスをオンまたはオフにします。

AWS Glue Studio Notebook

1. AWS Glue Studio Notebook ウィンドウの下部から Amazon Q を選択します。
2. ポップアップメニューから、[AWSとテレメトリを共有する] の横にあるスイッチを切り替えます。

> **Note**
>
> クライアント側のテレメトリの共有の一時停止は、現在の AWS Glue Studio ノートブックの期間中のみ有効です。

Command line

コマンドラインツールの [設定] で、[テレメトリ] を切り替えます。

Transformations on the command line

テレメトリ収集は、変換用のコマンドラインツールでデフォルトで有効になっています。無効にするには、次の手順を実行します。

テレメトリ設定を更新するには

1. を実行してリクエストされた設定の詳細`qct configure`を指定するか、Enter キーを押して既存の設定を使用します。
2. テレメトリ収集を許可するかどうかを確認するプロンプトが表示されたら、 と入力`N`して、がテレメトリデータを収集 AWS しないようにします。
3. テレメトリコレクションを再度有効にする場合は、 `qct configure` を再度実行し、プロンプトが表示され`Y`たら と入力します。

コンテンツの共有からのオプトアウト

コンテンツ AWS の使用については、「」を参照してくださいAmazon Q Developer のサービスの改善。

Visual Studio Code

Amazon Q Developer Pro 階層では、Amazon Q はコンテンツを収集しません。

Amazon Q Developer 無料利用枠で、VS Code でのコンテンツの共有からオプトアウトする手順は次のとおりです。

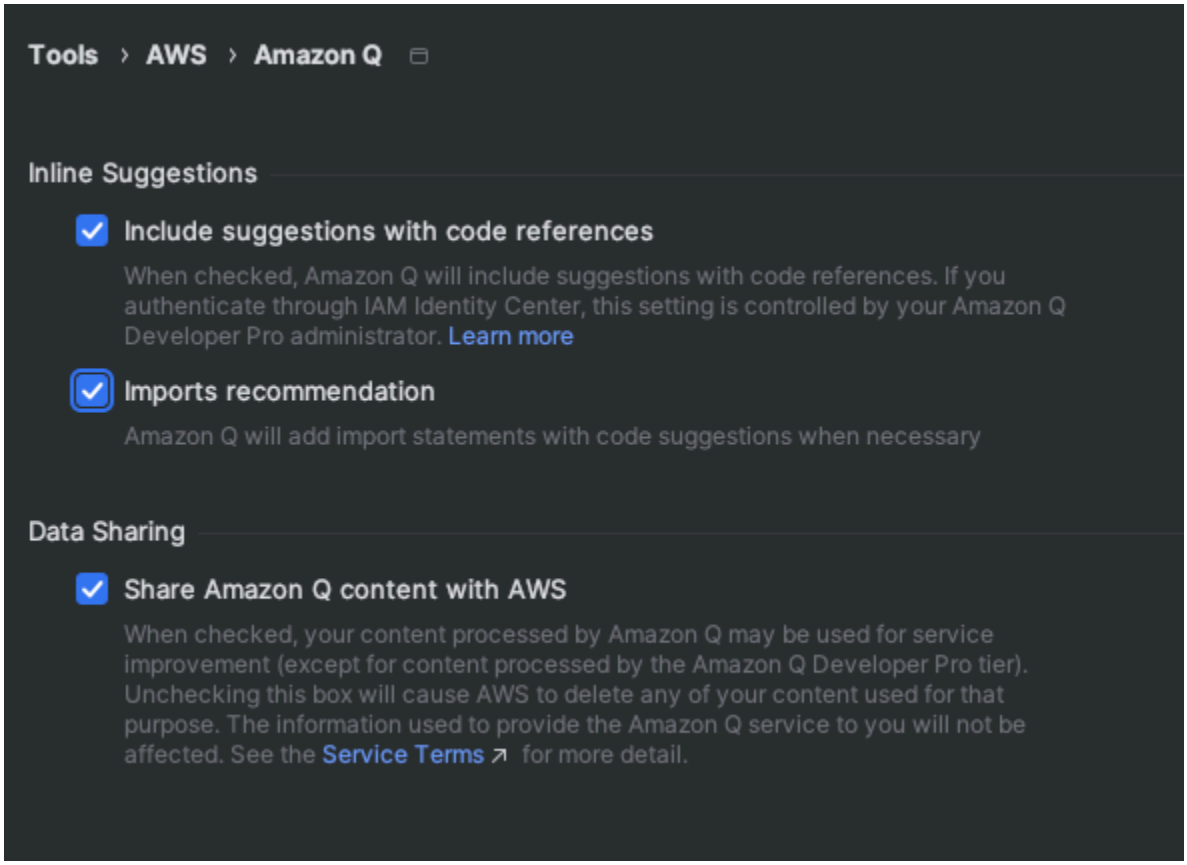
1. VS Code で [設定] を開きます。
2. VS Code ワークスペースを使用している場合は、[ワークスペース] サブタブに切り替えます。VS Code では、ワークスペース設定がユーザー設定より優先されます。
3. [検索] バーに「`Amazon Q: Share Content`」と入力します。
4. チェックボックスをオフにします。

JetBrains

Amazon Q Developer Pro 階層では、Amazon Q はコンテンツを収集しません。

Amazon Q Developer 無料利用枠で、JetBrains での Amazon Q データの共有からオプトアウトする手順は次のとおりです。

1. JetBrains の最新バージョンを利用していることを確認します。
2. JetBrains IDE で、[Preferences] を開きます (Mac では、これは [Settings] の下になります)。
3. 左側のナビゲーションバーで、[Tools] --> [AWS] --> [Amazon Q] を選択します。
4. データ共有で、Amazon Q コンテンツの共有の選択を解除します AWS。



Eclipse

Amazon Q Developer Pro 階層では、Amazon Q はコンテンツを収集しません。

Amazon Q Developer Free 階層で、EclipseIDEs での Amazon Q データの共有をオプトアウトするには、次の手順を使用します。

1. 最新バージョンの IDE Eclipse を使用していることを確認します。
2. Eclipse IDE で、設定 を開きます。
3. 左側のナビゲーションバーで、Amazon Q を選択します。
4. Amazon Q コンテンツを と共有 AWSの横にあるチェックボックスをオフにします。
5. [Apply (適用)] を選択して変更を保存します。

Visual Studio

Amazon Q Developer Pro 階層では、Amazon Q はコンテンツを収集しません。

Amazon Q Developer 無料利用枠で、Visual Studio でのコンテンツの共有からオプトアウトする手順は次のとおりです。

[ツール] -> [オプション] -> [AWS Toolkit] -> [Amazon Q] に移動します。

[Share Amazon Q Content with AWS] を [True] または [False] に切り替えます。

AWS Cloud9

で Amazon Q を使用する場合 AWS Cloud9、Amazon Q はコンテンツを と共有しません AWS。

> **Note**
>
> AWS Cloud9 設定には、Amazon Q コンテンツを共有するためのトグルスイッチが含まれていますが AWS、そのスイッチは機能しません。

Lambda

Amazon Q を Lambda で使用する場合、Amazon Q は AWSとコンテンツを共有しません。

> **Note**
>
> Lambda 設定には、Amazon Q コンテンツを共有するためのトグルスイッチが含まれていますが AWS、そのスイッチは機能しません。

SageMaker AI Studio

SageMaker AI Studio で Amazon Q を使用する場合、Amazon Q はコンテンツを共有しません AWS。

JupyterLab

1. JupyterLab ウィンドウの上部から、[設定] を選択します。
2. [設定] ドロップダウンから、[詳細設定エディタ] を選択します。
3. Amazon Q ドロップダウンで、[Share content with Amazon Q] の横にあるチェックボックスをオンまたはオフにします。

AWS Glue Studio Notebook

AWS Glue Studio Notebook で Amazon Q を使用する場合、Amazon Q はコンテンツを共有しません AWS。

Command line

コマンドラインツールの「設定」で、Amazon Q コンテンツの共有を切り替えます AWS。

Transformations on the command line

変換に Amazon Q コマンドラインツールを使用する場合、Amazon Q はコンテンツを共有しません AWS。

Amazon Q Developer でのクロスリージョン処理

以下のセクションでは、クロスリージョン推論とクロスリージョン呼び出しを使用して Amazon Q Developer サービスを提供する方法について説明します。

クロスリージョン推論

Amazon Q Developer は Amazon Bedrock を利用しており、クロスリージョン推論を使用して異なるにトラフィックを分散 AWS リージョン し、大規模言語モデル (LLM) 推論のパフォーマンスと信頼性を向上させます。クロスリージョン推論を使用すると、次のことが可能になります。

- 高需要期間中のスループットと耐障害性の向上
- パフォーマンスの向上
- Amazon Bedrock でホストされている最も強力な LLMs に依存する、新しく起動された Amazon Q Developer の機能へのアクセス

クロスリージョン推論リクエストは、データが最初に存在する地域の一部 AWS リージョン である内に保持されます。たとえば、米国で作成された Amazon Q Developer プロファイルから行われたリクエストは、米国の AWS リージョン 内に保持されます。一部の Amazon Q Developer の機能と統合は、Q Developer プロファイルが作成されたリージョン以外のリージョンで推論を実行する場合があります。詳細については、「Amazon Q Developer クロスリージョン推論でサポートされているリージョン」を参照してください。

クロスリージョン推論はデータの保存場所を変更しませんが、リクエストと出力結果がデータが存在するリージョン外に移動する可能性があります。すべてのデータは、Amazon の安全なネットワーク経由で送信される間、暗号化されます。クロスリージョン推論の使用に追加コストはかかりません。

クロスリージョン推論は、データの保存先には影響しません。Amazon Q Developer の使用時にデータを保存する場所については、「」を参照してくださいAmazon Q Developer でのデータ保護。

Amazon Q Developer クロスリージョン推論でサポートされているリージョン

次の表は、リクエストの送信元の地域に応じて、リクエストをルーティングできるリージョンを示しています。

サポートされている Amazon Q Developer の地域	推論リージョン
アメリカ	米国東部 (バージニア北部) (us-east-1) 米国西部 (オレゴン) (us-west-2) 米国東部 (オハイオ) (us-east-2)
欧州	ヨーロッパ (フランクフルト) (eu-central-1) 欧州 (アイルランド) (eu-west-1) 欧州 (パリ) (eu-west-3) 欧州 (ストックホルム) (eu-north-1)
アジアパシフィック*	アジアパシフィック (ムンバイ) (ap-south-1) アジアパシフィック (ソウル) (ap-northeast-2) アジアパシフィック (シンガポール) (ap-southeast-1) アジアパシフィック (シドニー) (ap-southeast-2) アジアパシフィック (東京) (ap-northeast-1)

*アジアパシフィックリージョンでのクロスリージョン推論は、アジアパシフィック (ソウル) リージョンで Amazon Q 生成 SQL を使用する場合にのみサポートされます。

Amazon Q Developer を使用できるリージョンの完全なリストについては、「」を参照してくださいAmazon Q Developer がサポートされているリージョン。

クロスリージョン呼び出し

Amazon Q Developer に対して行う特定のリクエストでは、クロスリージョン呼び出しが必要になる場合があります。クロスリージョン呼び出しは、Amazon Q が 1 つの から別の AWS リージョン に対して行う API コールです AWS リージョン。Amazon Q は、リクエストで現在のリージョンとは異なるリージョンから情報を取得する必要がある場合に、クロスリージョン呼び出しを行います。例えば、異なるリージョンにある AWS リソースについて Amazon Q に質問すると、クロスリージョン呼び出しを行ってリソースにアクセスし、関連するデータを取得して質問に回答します。さらに、Amazon Q からの応答がグローバル AWS サービスエンドポイントからの情報を必要とする場合、Amazon Q はデータが保存されているリージョン外で呼び出しを行うことがあります。グローバルサービスの詳細については、「障害分離境界ホワイトペーパー」の「グローバルサービス」を参照してください。 AWS AWS

Amazon Q Developer によって行われたクロスリージョン呼び出しを無効にする場合は、Amazon Q がユーザーに代わって API 呼び出しを実行できないようにするポリシーを作成できます。これにより、Amazon Q が現在のリージョン内で呼び出しを行っている場合でも、Amazon Q がユーザーに代わって API 呼び出しを行う必要がある機能にアクセスできなくなります。Amazon Q がユーザーに代わってクロスリージョン呼び出しを含む API 呼び出しを実行できないようにする IAM ポリシーについては、「」を参照してくださいAmazon Q のアクセス許可でユーザーに代わってアクションを実行することを拒否する。

Amazon Q Developer における ID 管理とアクセス管理

AWS Identity and Access Management （ IAM) は、管理者が AWS resources へのアクセスを安全に制御 AWS のサービス するのに役立つ です。 IAM 管理者は、誰を認証 (サインイン) し、誰に Amazon Q Developer リソースの使用を許可する (アクセス許可を付与 AWS のサービス する) かを制御します。 IAM は、追加料金なしで使用できる です。

トピック

- 対象者
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- Amazon Q Developer が IAM と連携する仕組み
- ポリシーを使用して Amazon Q Developer へのアクセスを管理する
- サードパーティー統合のための Amazon Q Developer へのアクセスを管理する
- Amazon Q Developer のアクセス許可リファレンス

- AWS Amazon Q Developer の マネージドポリシー
- Amazon Q Developer およびユーザーサブスクリプションでサービスリンクロールを使用する

対象者

の使用方法 IAM は、Amazon Q で行う作業によって異なります。

サービスユーザー – 業務を行うために Amazon Q サービスを使用する場合、管理者が必要な認証情報および許可を提供します。業務のために使用する Amazon Q 機能が増えるにつれ、追加の許可が必要な場合があります。アクセスの管理方法を理解すると、管理者に適切なアクセス許可をリクエストするのに役に立ちます。

サービス管理者 — 社内の Amazon Q リソースを担当している場合、Amazon Q へのフルアクセス許可がある可能性が考えられます。サービスのユーザーがどの Amazon Q 機能およびリソースにアクセスするかについて決定するのはユーザーの役割です。その後で、サービスユーザーのアクセス許可を変更するために、 IAM 管理者にリクエストを送信する必要があります。このページの情報を確認して、 の基本概念を理解します IAM。会社が Amazon Q IAM で を使用する方法の詳細については、「Amazon Q の仕組み IAM」を参照してください。

IAM 管理者 – IAM 管理者は、Amazon Q へのアクセスを管理するポリシーの作成方法の詳細について確認する場合があります。IAM 管理者の場合は、サービスへの IAM ユーザーアクセスを管理するポリシーの作成方法の詳細について確認することを検討してください。Amazon Q に固有の情報については、「Amazon Q のAWS リージョン マネージドポリシー」を参照してください。

アイデンティティを使用した認証

認証は、ID 認証情報 AWS を使用して にサインインする方法です。 AWS アカウント ルートユーザー、、または IAM ロールを引き受けることで IAM ユーザー、認証 (にサインイン AWS) される必要があります。

ID ソースを通じて提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、および Googleまたは Facebook認証情報は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインすると、管理者は以前に IAM ロールを使用して ID フェデレーションをセットアップしていました。フェデレーションを使用して にアクセスすると、間接的 AWS にロールを引き受けることになります。

ユーザーの種類に応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 AWS サインイン ユーザーガイド」の「 へのサインイン AWS アカウント方法」を参照してください。

使用する認証方法を問わず、追加のセキュリティ情報の提供を要求される場合もあります。たとえば、 AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを強化することをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「Multi-factor authentication」(多要素認証) および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」を参照してください。

AWS アカウントのルートユーザー

を初めて作成するときは AWS アカウント、アカウント内のすべての AWS のサービス およびリソースへの完全なアクセス権を持つ単一のサインインアイデンティティから始めます。この ID は と呼ばれ AWS アカウントのルートユーザー 、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報を保護し、ルートユーザーのみが実行できるタスク実行に使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「 IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、ID プロバイダーとのフェデレーションを使用して一時的な認証情報 AWS のサービス を使用して にアクセスすることを要求します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリ、または ID ソースを介して提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID がアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成するか、独自の ID ソースのユーザーとグループのセットに接続して同期し、すべての AWS アカウント とアプリケーションで使用できます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center とは) を参照してください。

IAM ユーザー および グループ

IAM ユーザー は、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を持つ IAM ユーザー ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、 IAM ユーザーに長期的な認証情報が必要な特定のユースケースがある場合、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、 のコレクションを指定する ID です IAM ユーザー。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。たとえば、IAMAdmins という名前のグループがあり、そのグループに IAM リソースを管理するアクセス許可を付与できます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー の作成が適している場合 (ロールではなく)」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可 AWS アカウント を持つ 内の ID です。IAM ロールは に似 IAM ユーザー ていますが、特定のユーザーに関連付けられていません。 IAM ロールを切り替える AWS Management Console ことで、 で ロールを一時的に引き受けることができます。 AWS Command Line Interface （ AWS CLI) または AWS API オペレーションを呼び出すか、カスタム URL を使用してロールを引き受けることができます。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM 一時的な認証情報を持つ ロールは、以下の状況で役立ちます。

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「サードパーティーアイデンティティプロバイダー向けロールの作成」を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。ID が認証後にアクセスできるものをコントロールするために、IAM アイデンティティセン

ター は権限セットを IAMのロールに関連付けます。アクセス許可セットの詳細については、AWS IAM Identity Center ユーザーガイドの「アクセス権限セット」を参照してください。

- 一時的な IAM ユーザー アクセス許可 – は、 IAM ロールを引き受けて、特定のタスクに対して異なるアクセス許可を一時的に引き受け IAM ユーザー ることができます。
- クロスアカウントアクセス IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを別のアカウントの信頼済みプリンシパルに許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の AWS のサービスでは、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについて詳しくは、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。
- クロスサービスアクセス – 一部の AWS のサービス は他の の機能を使用します AWS のサービス。サービスは、呼び出し元のプリンシパルの許可、サービスロール、サービスリンクロールを使用してこれを行う場合があります。
  - プリンシパルアクセス許可 – IAM ユーザー または ロールを使用してアクションを実行すると AWS、プリンシパルと見なされます。ポリシーによって、プリンシパルに許可が付与されます。一部のサービスを使用する際に、アクションを実行することで、別サービスの別アクションがトリガーされることがあります。この場合、両方のアクションを実行するためのアクセス許可が必要です。
  - サービスロール – サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受ける IAM ロールです。 IAM 管理者は、 IAM内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービスにアクセス許可を委任するロールの作成」を参照してください。
  - サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。 IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。
- で Amazon EC2 実行されているアプリケーション – IAM ロールを使用して、 Amazon EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、 Amazon EC2 インスタンス内でのアクセスキーの保存に推奨されます。 IAM ロールを Amazon EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルには ロールが含まれており、 Amazon EC2 インスタンスで実行されているプログラムが一時的な認証情報を取得できるようにします。詳細については、IAM

ユーザーガイドの IAM 「 ロールを使用して Amazon EC2 インスタンスで実行されているアプリケーションにアクセス許可を付与する」を参照してください。

IAM ロールを使用するかどうかの詳細については、IAM ユーザーガイドの (ユーザーではなく) IAM ロールを作成するタイミングを参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは AWS 、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「IAM ユーザーガイド」の「JSON ポリシー概要」を参照してください。

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

すべての IAM エンティティ (ユーザーまたはロール) は、アクセス許可なしで始まります。デフォルトでは、ユーザーは何もできず、自分のパスワードを変更することすらできません。何かを実行する許可をユーザーに付与するには、管理者がユーザーに許可ポリシーをアタッチする必要があります。また、管理者は、必要な許可があるグループにユーザーを追加できます。管理者がグループに許可を付与すると、そのグループ内のすべてのユーザーにこれらの許可が付与されます。

IAM ポリシーは、オペレーションの実行に使用するメソッドに関係なく、アクションのアクセス許可を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

ID ベースのポリシーは、、 IAM ユーザーロール、グループなどの ID にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。IAM ポリシー作成方法の詳細については、「IAM ユーザーガイド」の「IAM ポリシーを作成する」を参照してください。

アイデンティティベースのポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーの選択方法については、「IAM ユーザーガイド」の「管理ポリシーとインラインポリシーのいずれかを選択する」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、 Amazon S3 バケットなどのリソースにアタッチする JSON ポリシードキュメントです。サービス管理者は、これらのポリシーを使用して、特定のプリンシパル (アカウントメンバー、ユーザー、またはロール) がそのリソースに対して実行する条件およびアクションを定義することができます。リソースベースのポリシーはインラインポリシーです。マネージド型のリソースベースのポリシーはありません。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、ロール) がリソースにアクセスする許可を持つかについて管理するポリシーのタイプです。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式を使用しません。 Amazon S3、、 Amazon VPC は AWS WAF ACLs。詳細については、「Amazon S3 ユーザーガイド」の「アクセスコントロールリスト (ACL) の概要」を参照してください。

その他のポリシータイプ

AWS は、一般的でない追加のポリシータイプをサポートしています。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 – アクセス許可の境界は、アイデンティティベースのポリシーが IAM エンティティ (IAM ユーザー またはロール) に付与できるアクセス許可の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られるアクセス許可は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPsは、 の組織または組織単位 (OU) の最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、ビジネスが所有する複数の AWS アカウント をグループ化して一元管理するためのサービスです。組織内のす

べての機能を有効にすると、SCP を一部またはすべてのアカウントに適用できます。SCP は、各 AWS アカウント ルートユーザーを含むメンバーアカウントのエンティティのアクセス許可を制限します。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。

- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果として得られるセッションの許可は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分です。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「ポリシー評価ロジック」を参照してください。

Amazon Q Developer が IAM と連携する仕組み

IAM を使用して Amazon Q Developer へのアクセスを管理する前に、Amazon Q Developer で利用できる IAM 機能について理解しておく必要があります。

Amazon Q Developer で使用できる IAM の機能

IAM の機能	Amazon Q のサポート
アイデンティティベースポリシー	はい
リソースベースのポリシー	いいえ
ポリシーアクション	はい
ポリシーリソース	いいえ
ポリシー条件キー	いいえ
ACL	いいえ

IAM の機能	Amazon Q のサポート
ABAC (ポリシー内のタグ)	いいえ
一時的な認証情報	はい
プリンシパル権限	はい
サービスロール	いいえ
サービスリンクロール	あり

Amazon Q およびその他の がほとんどの IAM 機能と AWS のサービス 連携する方法の概要を把握するには、AWS のサービス 「IAM ユーザーガイド」の「IAM と連携する」を参照してください。

Amazon Q のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。ID ベースのポリシーの作成方法については、「IAM ユーザーガイド」の「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「IAM JSON ポリシーの要素のリファレンス」を参照してください。

Amazon Q のアイデンティティベースのポリシー例

Amazon Q Developer のアイデンティティベースのポリシー例を確認するには、「Amazon Q Developer のアイデンティティベースのポリシー例」を参照してください。

Amazon Q 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。詳細については、「IAM ユーザーガイド」の「IAM でのクロスアカウントリソースアクセス」を参照してください。

Amazon Q のポリシーアクション

ポリシーアクションのサポート:あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action` 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは依存アクションと呼ばれます。

このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

Amazon Q アクションのリストを表示するには、「ポリシーを使用して Amazon Q Developer へのアクセスを管理する」を参照してください 。

Amazon Q のポリシーアクションは、アクションの前に次のプレフィックスを使用します。

```
q
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [
      "q:action1",
      "q:action2"
         ]
```

ワイルドカード (*) を使用して複数アクションを指定できます。例えば、Get という単語で始まるすべてのアクションを指定するには次のアクションを含めます。

```
"Action": "q:Get*"
```

Amazon Q Developer のアイデンティティベースのポリシー例を確認するには、「Amazon Q Developer のアイデンティティベースのポリシー例」を参照してください。

Amazon Q のポリシーリソース

ポリシーリソースのサポート: なし

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ステートメントにはResource または NotResource 要素を含める必要があります。ベストプラクティスとして、Amazon リソースネーム (ARN) を使用してリソースを指定します。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

Amazon Q Developer のアイデンティティベースのポリシー例を確認するには、「Amazon Q Developer のアイデンティティベースのポリシー例」を参照してください。

Amazon Q のポリシー条件キー

サービス固有のポリシー条件キーへのサポート: なし

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。イコールや未満などの 条件演算子 を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、 AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「IAM ポリシーの要素: 変数およびタグ」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートしています。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」のAWS 「グローバル条件コンテキストキー」を参照してください。

Amazon Q Developer のアイデンティティベースのポリシー例を確認するには、「Amazon Q Developer のアイデンティティベースのポリシー例」を参照してください。

Amazon Q の ACL

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Amazon Q で ABAC

ABAC (ポリシー内のタグ) のサポート: なし

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「ABAC 認可でアクセス許可を定義する」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「属性ベースのアクセスコントロール (ABAC) を使用する」を参照してください。

Amazon Q で一時的な認証情報の使用

一時的な認証情報のサポート: あり

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドの「IAM AWS のサービス と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。たとえば、会社のシングルサインオン (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「ユーザーから IAM ロールに切り替える (コンソール)」を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用してアクセスすることができます AWS。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM の一時的セキュリティ認証情報」を参照してください。

Amazon Q のクロスサービスプリンシパルの許可

転送アクセスセッション (FAS) のサポート: あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする と組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

Amazon Q のサービスロール

サービスロールのサポート: なし

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービスに許可を委任するロールを作成する」を参照してください。

> **Warning**
>
> サービスロールの許可を変更すると、Amazon Q の機能が破損する可能性があります。Amazon Q が指示するときにのみ、サービスロールを編集してください。

Amazon Q のサービスリンクロール

サービスリンクロールのサポート: あり

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービ

スにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスリンクロールのアクセス許可を表示できますが、編集することはできません。

Amazon Q でのサービスリンクロールの作成または管理の詳細については、「Amazon Q Developer およびユーザーサブスクリプションでサービスリンクロールを使用する」を参照してください。

ポリシーを使用して Amazon Q Developer へのアクセスを管理する

> **Note**
>
> このページの情報は、Amazon Q Developer へのアクセスに関するものです。Amazon Q Business へのアクセス管理の詳細については、「Amazon Q Business デベロッパーガイド」の「Identity-based policy examples for Amazon Q Business」を参照してください。このトピックのポリシーと例は AWS Management Console、、、 AWS Console Mobile Application AWS ウェブサイト AWS Documentation、チャットアプリケーションの Amazon Q に固有のものです。Amazon Q と統合された他のサービスは、異なるポリシーや設定が必要になる場合があります。サードパーティー IDE の Amazon Q のエンドユーザーは、IAM ポリシーを使用する必要はありません。詳細については、Amazon Q の機能または統合を含むサービスのドキュメントを参照してください。

デフォルトでは、ユーザーとロールには Amazon Q を使用するアクセス許可がありません。IAM 管理者は、IAM ID にアクセス許可を付与することで、Amazon Q Developer とその機能へのアクセスを管理できます。

管理者がユーザーにアクセス権を付与する最も簡単な方法は、 AWS マネージドポリシーを使用することです。`AmazonQFullAccess` ポリシーを IAM ID にアタッチして、Amazon Q Developer とその機能へのフルアクセスを許可できます。このポリシーの詳細については、「AWS Amazon Q Developer の マネージドポリシー」を参照してください。

IAM ID が Amazon Q Developer で実行できる特定のアクションを管理するには、ユーザー、グループ、ロールが持つ許可を定義するカスタムポリシーを作成できます。また、サービスコントロールポリシー (SCP) を使用して、組織で使用できる Amazon Q の機能を管理することもできます。

ポリシーで管理できる Amazon Q のアクセス許可の全一覧については、「Amazon Q Developer のアクセス許可リファレンス」を参照してください。

トピック

- ポリシーに関するベストプラクティス

- アクセス許可の割り当て
- サービスコントロールポリシー (SCP) を使用してのアクセス管理
- Amazon Q Developer のアイデンティティベースのポリシー例

ポリシーに関するベストプラクティス

アイデンティティベースのポリシーは、誰がユーザーのアカウントの Amazon Q Developer リソースを作成、アクセス、削除できるどうかを決定します。これらのアクションを実行すると、 AWS アカウントに料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」または「ジョブ機能のAWS マネージドポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「IAM でのポリシーとアクセス許可」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションが などの特定の を通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、「IAM ユーザーガイド」の「IAM JSON ポリシー要素:条件」を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、「IAM ユーザーガイド」の「IAM Access Analyzer でポリシーを検証する」を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーション

が呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「MFA を使用した安全な API アクセス」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

アクセス許可の割り当て

アクセス権限を付与するにはユーザー、グループ、またはロールにアクセス許可を追加します。

- のユーザーとグループ AWS IAM Identity Center :

  アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」の手順に従ってください。

- IAM 内で、ID プロバイダーによって管理されているユーザー:

  ID フェデレーションのロールを作成します。詳細については「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールを作成する」を参照してください。

- IAM ユーザー:
  - ユーザーが担当できるロールを作成します。手順については「IAM ユーザーガイド」の「IAM ユーザーのロールの作成」を参照してください。
  - (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加します。詳細については、「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス権限の追加」を参照してください。

サービスコントロールポリシー (SCP) を使用してのアクセス管理

サービスコントロールポリシー (SCP) は、組織のアクセス許可の管理に使用できる組織ポリシーの一種です。Amazon Q Developer アクションの一部またはすべてに対して許可を指定する SCP を作成することにより、組織で利用できる Amazon Q Developer の機能を管理できます。

SCP を使用して組織内のアクセスを管理する方法の詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシーの作成、更新、削除」および「サービスコントロールポリシーのアタッチとデタッチ」を参照してください。

SCP の例: EU リージョン外の Amazon Q へのアクセスを拒否する

次の SCP は、欧州 (フランクフルト) リージョン (eu-central-1) 以外の Amazon Q Developer の使用を拒否します。

> **Note**
>
> codewhisperer プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAmazonQDeveloperOutsideEU",
      "Effect": "Deny",
      "Action": [
         "codewhisperer:GenerateRecommendations",
         "q:SendMessage",
         "q:GenerateCodeFromCommands",
         "sqlworkbench:GetQSqlRecommendations"
         ],
      "Resource": "*",
      "Condition": {
        "StringNotEquals":
        {"aws:RequestedRegion": [ "eu-central-1"] }
      }
    }
  ]
}
```

SCP の例: Amazon Q へのアクセスを拒否する

次の SCP は、Amazon Q Developer へのアクセスを拒否します。

> **Note**
>
> Amazon Q へのアクセスを拒否しても、コンソール、 AWS ウェブサイト、 AWS ドキュメントページ、または の Amazon Q アイコンまたはチャットパネルは AWS 無効になりません AWS Console Mobile Application。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAmazonQFullAccess",
      "Effect": "Deny",
      "Action": [
        "q:*"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Q Developer のアイデンティティベースのポリシー例

次の IAM ポリシー例は、さまざまな Amazon Q Developer アクションに対するアクセス許可を管理するためのポリシーです。これらを使用してユーザー、ロール、グループに Amazon Q Developer へのアクセスを許可または拒否します。

> **Note**
>
> 次のポリシー例は、Amazon Q Developer の機能に対するアクセス許可を付与するものですが、Amazon Q Developer Pro サブスクリプションを使用して Amazon Q にアクセスするには、追加のアクセス許可が必要になる場合があります。詳細については、「Amazon Q Developer Pro サブスクリプションを利用した Amazon Q へのアクセスをユーザーに許可する」を参照してください。

これらのポリシーは、書かれているとおりに使用することも、使用する個々の Amazon Q の機能に対するアクセス許可を追加することもできます。Amazon Q で IAM のアクセス許可を設定する方法については、「ポリシーを使用して Amazon Q Developer へのアクセスを管理する」を参照してください。

ポリシーで管理できるすべての Amazon Q アクセス許可のリストについては、「Amazon Q Developer のアクセス許可リファレンス」を参照してください。

トピック

- 管理者のアクセス許可
- ユーザーアクセス許可

管理者のアクセス許可

次のポリシーにより、Amazon Q Developer 管理者は Amazon Q サブスクリプション管理コンソールと Amazon Q Developer コンソールで管理タスクを実行できます。

Amazon Q Developer の機能を使用するためのポリシーについては、「ユーザーアクセス許可」を参照してください。

管理者に Amazon Q コンソールの使用を許可する

次のポリシー例では、Amazon Q コンソールでアクションを実行するアクセス許可をユーザーに付与します。Amazon Q コンソールは、Amazon Q と AWS IAM Identity Center との統合を設定する場所です AWS Organizations。他のほとんどの Amazon Q Developer 関連タスクは、Amazon Q Developer コンソールで完了する必要があります。詳細については、「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

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

```
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "organizations:ListAWSServiceAccessForOrganization",
            "organizations:DisableAWSServiceAccess",
            "organizations:EnableAWSServiceAccess",
            "organizations:DescribeOrganization"
         ],
         "Resource":[
            "*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "sso:ListApplications",
            "sso:ListInstances",
            "sso:DescribeRegisteredRegions",
            "sso:GetSharedSsoConfiguration",
            "sso:DescribeInstance",
            "sso:CreateInstance",
            "sso:CreateApplication",
            "sso:PutApplicationAuthenticationMethod",
            "sso:PutApplicationAssignmentConfiguration",
            "sso:PutApplicationGrant",
            "sso:PutApplicationAccessScope",
            "sso:DescribeApplication",
            "sso:DeleteApplication",
            "sso:GetSSOStatus",
            "sso:CreateApplicationAssignment",
            "sso:DeleteApplicationAssignment",
            "sso:UpdateApplication"
         ],
         "Resource":[
            "*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "sso-directory:DescribeUsers",
            "sso-directory:DescribeGroups",
            "sso-directory:SearchGroups",
```

```
            "sso-directory:SearchUsers",
            "sso-directory:DescribeGroup",
            "sso-directory:DescribeUser",
            "sso-directory:DescribeDirectory"
         ],
         "Resource":[
            "*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "signin:ListTrustedIdentityPropagationApplicationsForConsole",
            "signin:CreateTrustedIdentityPropagationApplicationForConsole"
         ],
         "Resource":[
            "*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "codewhisperer:ListProfiles",
            "codewhisperer:CreateProfile",
            "codewhisperer:DeleteProfile"
         ],
         "Resource":[
            "*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "user-subscriptions:ListClaims",
            "user-subscriptions:ListUserSubscriptions",
            "user-subscriptions:CreateClaim",
            "user-subscriptions:DeleteClaim",
            "user-subscriptions:UpdateClaim"
         ],
         "Resource":[
            "*"
         ]
      },
      {
```

```
            "Effect":"Allow",
            "Action":[
               "q:CreateAssignment",
               "q:DeleteAssignment"
            ],
            "Resource":[
               "*"
            ]
         },
         {
            "Effect":"Allow",
            "Action":[
               "iam:CreateServiceLinkedRole"
            ],
            "Resource":[
               "arn:aws:iam::*:role/aws-service-role/user-
subscriptions.amazonaws.com/AWSServiceRoleForUserSubscriptions"
            ]
         }
      ]
}
```

管理者に Amazon Q Developer コンソールの使用を許可する

次のポリシー例では、ユーザーに Amazon Q Developer コンソールへのアクセス許可を付与します。Amazon Q Developer コンソールでは、管理者はサブスクリプション、コードリファレンス、カスタマイズ、チャットプラグインに関連するタスクなど、Amazon Q Developer 関連のほとんどの設定タスクを実行します。このポリシーには、カスタマーマネージド KMS キーを作成および設定するためのアクセス許可も含まれています。

管理者が (Amazon Q Developer コンソールではなく) Amazon Q コンソールを使用して完了する必要がある Amazon Q Developer Pro タスクがいくつかあります。詳細については、「管理者に Amazon Q コンソールの使用を許可する」を参照してください。

> **Note**
>
> カスタマイズまたはプラグインを作成するには、Amazon Q Developer Pro 管理者に追加のアクセス許可が必要です。
>
> - カスタマイズに必要なアクセス許可については、「カスタマイズの前提条件」を参照してください。

- プラグインに必要なアクセス許可については、「」を参照してください管理者にプラグインの設定を許可する。

Amazon Q Developer コンソールを使用するには、2 つのポリシーのいずれかが必要です。必要なポリシーは、Amazon Q Developer を初めて設定するか、レガシー Amazon CodeWhisperer プロファイルを設定するかによって異なります。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

Amazon Q Developer の新しい管理者には、次のポリシーを使用します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sso:ListInstances",
        "sso:CreateInstance",
        "sso:CreateApplication",
        "sso:PutApplicationAuthenticationMethod",
        "sso:PutApplicationGrant",
        "sso:PutApplicationAssignmentConfiguration",
        "sso:ListApplications",
        "sso:GetSharedSsoConfiguration",
        "sso:DescribeInstance",
        "sso:PutApplicationAccessScope",
        "sso:DescribeApplication",
        "sso:DeleteApplication",
        "sso:CreateApplicationAssignment",
        "sso:DeleteApplicationAssignment",
        "sso:UpdateApplication",
```

```
        "sso:DescribeRegisteredRegions",
        "sso:GetSSOStatus"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "sso-directory:GetUserPoolInfo",
        "sso-directory:DescribeUsers",
        "sso-directory:DescribeGroups",
        "sso-directory:SearchGroups",
        "sso-directory:SearchUsers",
        "sso-directory:DescribeDirectory"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "signin:ListTrustedIdentityPropagationApplicationsForConsole",
        "signin:CreateTrustedIdentityPropagationApplicationForConsole"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "user-subscriptions:ListClaims",
```

```
        "user-subscriptions:ListApplicationClaims",
        "user-subscriptions:ListUserSubscriptions",
        "user-subscriptions:CreateClaim",
        "user-subscriptions:DeleteClaim",
        "user-subscriptions:UpdateClaim"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DisableAWSServiceAccess",
        "organizations:EnableAWSServiceAccess"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:ListAliases",
        "kms:CreateGrant",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey*",
        "kms:RetireGrant",
        "kms:DescribeKey"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "codeguru-security:UpdateAccountConfiguration"
      ],
      "Resource": [
```

```
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/q.amazonaws.com/
AWSServiceRoleForAmazonQDeveloper"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "codewhisperer:UpdateProfile",
        "codewhisperer:ListProfiles",
        "codewhisperer:TagResource",
        "codewhisperer:UnTagResource",
        "codewhisperer:ListTagsForResource",
        "codewhisperer:CreateProfile"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "q:ListDashboardMetrics",
        "q:CreateAssignment",
        "q:DeleteAssignment"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:GetMetricData",
        "cloudwatch:ListMetrics"
      ],
```

```
      "Resource": [
        "*"
      ]
    }
  ]
}
```

レガシー Amazon CodeWhisperer プロファイルの場合、次のポリシーにより、IAM プリンシパルが CodeWhisperer アプリケーションを管理できるようになります。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sso-directory:SearchUsers",
        "sso-directory:SearchGroups",
        "sso-directory:GetUserPoolInfo",
        "sso-directory:DescribeDirectory",
        "sso-directory:ListMembersInGroup"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "pricing:GetProducts"
```

```
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "sso:AssociateProfile",
        "sso:DisassociateProfile",
        "sso:GetProfile",
        "sso:ListProfiles",
        "sso:ListApplicationInstances",
        "sso:GetApplicationInstance",
        "sso:CreateManagedApplicationInstance",
        "sso:GetManagedApplicationInstance",
        "sso:ListProfileAssociations",
        "sso:GetSharedSsoConfiguration",
        "sso:ListDirectoryAssociations",
        "sso:DescribeRegisteredRegions",
        "sso:GetSsoConfiguration",
        "sso:GetSSOStatus"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "identitystore:ListUsers",
        "identitystore:ListGroups"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
      ],
      "Resource": [
```

```
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:ListAliases",
        "kms:CreateGrant",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey*",
        "kms:RetireGrant",
        "kms:DescribeKey"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "codeguru-security:UpdateAccountConfiguration"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/q.amazonaws.com/
AWSServiceRoleForAmazonQDeveloper"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "codewhisperer:UpdateProfile",
        "codewhisperer:ListProfiles",
        "codewhisperer:TagResource",
        "codewhisperer:UnTagResource",
```

```
        "codewhisperer:ListTagsForResource",
        "codewhisperer:CreateProfile"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "q:ListDashboardMetrics",
        "cloudwatch:GetMetricData",
        "cloudwatch:ListMetrics"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

管理者にカスタマイズの作成を許可する

次のポリシーは、Amazon Q Developer でカスタマイズを作成および管理するためのアクセス許可を管理者に付与します。

Amazon Q Developer コンソールでカスタマイズを設定するには、Amazon Q Developer 管理者が Amazon Q Developer コンソールにアクセスする必要があります。詳細については、「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。

> **Note**
>
> 次のポリシーでは、IAM サービスは `codeconnections:ListOwners`および アクセス`codeconnections:ListRepositories`許可に関するエラーを報告します。とにかく、これらのアクセス許可を持つポリシーを作成します。アクセス許可は必須であり、エラーが発生してもポリシーは機能します。

> **Note**
>
> codewhisperer プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

次の例では、########自分の AWS アカウント番号に置き換えます。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso-directory:DescribeUsers"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "codewhisperer:CreateCustomization",
                "codewhisperer:DeleteCustomization",
                "codewhisperer:ListCustomizations",
                "codewhisperer:ListCustomizationVersions",
                "codewhisperer:UpdateCustomization",
                "codewhisperer:GetCustomization",
                "codewhisperer:ListCustomizationPermissions",
```

```
                "codewhisperer:AssociateCustomizationPermission",
                "codewhisperer:DisassociateCustomizationPermission"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "codeconnections:ListConnections",
                "codeconnections:ListOwners",
                "codeconnections:ListRepositories",
                "codeconnections:GetConnection"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "codeconnections:UseConnection",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "codeconnections:ProviderAction": [
                        "GitPull",
                        "ListRepositories",
                        "ListOwners"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject*",
                "s3:GetBucket*",
                "s3:ListBucket*"
            ],
            "Resource": [
                "*"
            ]
        }
```

```
    ]
}
```

管理者にプラグインの設定を許可する

次のポリシー例では、Amazon Q Developer コンソールでサードパーティープラグインを表示および設定するアクセス許可を管理者に付与します。

> **Note**
>
> Amazon Q Developer コンソールにアクセスするには、管理者にも で定義されているアクセス許可が必要です[管理者に Amazon Q Developer コンソールの使用を許可する](#)。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "q:CreatePlugin",
        "q:GetPlugin",
        "q:DeletePlugin",
        "q:ListPlugins",
        "q:ListPluginProviders",
        "q:UpdatePlugin",
        "q:CreateAuthGrant",
        "q:CreateOAuthAppConnection",
        "q:SendEvent",
        "q:UpdateAuthGrant",
        "q:UpdateOAuthAppConnection",
        "q:UpdatePlugin",
        "iam:CreateRole",
        "secretsmanager:CreateSecret"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
```

```
      "Action": [
        "iam:PassRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "q.amazonaws.com"
          ]
        }
      }
    }
  ]
}
```

管理者が 1 つのプロバイダーからプラグインを設定できるようにする

次のポリシー例では、プラグインプロバイダーの名前とワイルドカード文字 () を使用してプラグイン ARN で指定された 1 つのプロバイダーからプラグインを設定するアクセス許可を管理者に付与します*。このポリシーを使用するには、リソースフィールドの ARN で以下を置き換えます。

- *AWS-region* – プラグインが作成される AWS リージョン 。
- *AWS-account-ID* – プラグインが設定されているアカウントのアカウント AWS ID。
- *plugin-provider* – CloudZero、、Datadogまたは など、設定を許可するプラグインプロバイダーの名前Wiz。プラグインプロバイダーフィールドでは、大文字と小文字が区別されます。

> **Note**
>
> Amazon Q Developer コンソールにアクセスするには、管理者にも で定義されているアクセス許可が必要です管理者に Amazon Q Developer コンソールの使用を許可する。

JSON

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

```
            "Sid": "Allow permissions to create a plugin from one provider",
            "Effect": "Allow",
            "Action": [
                "q:CreatePlugin",
                "q:GetPlugin",
                "q:DeletePlugin"
            ],
            "Resource": "arn:aws:qdeveloper:us-east-1:AWS-account-
ID:plugin/plugin-provider/*"
        }
    ]
}
```

複数のネットワークまたは複数のサブネットの移行を許可する

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Sid": "MGNNetworkMigrationAnalyzerEC2ResourceSgTag",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateSecurityGroup"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:vpc/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "AWSApplicationMigrationService"
                }
            }
        },
        {
            "Sid": "MGNNetworkMigrationAnalyzerEC2RequestSgTag",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateSecurityGroup"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:security-group/*",
```

```
                "arn:aws:ec2:region:account-id:security-group-rule/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "AWSApplicationMigrationService"
                }
            }
        },

        {
            "Sid": "MGNNetworkMigrationAnalyzerEC2SecurityGroupTags",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:security-group/*",
                "arn:aws:ec2:region:account-id:security-group-rule/*",
                "arn:aws:ec2:region:account-id:network-interface/*",
                "arn:aws:ec2:region:account-id:network-insights-path/*",
                "arn:aws:ec2:region:account-id:network-insights-analysis/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "AWSApplicationMigrationService",
                    "ec2:CreateAction": [
                        "CreateSecurityGroup",
                        "CreateNetworkInterface",
                        "CreateNetworkInsightsPath",
                        "StartNetworkInsightsAnalysis"
                    ]
                }
            }
        },
        {
            "Sid": "MGNNetworkMigrationAnalyzerENIResourceTag",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:subnet/*"
            ],
            "Condition": {
```

```
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "AWSApplicationMigrationService"
                }
            }
        },
        {
            "Sid": "MGNNetworkMigrationAnalyzerENISG",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:security-group/*"
            ]
        },
        {
            "Sid": "MGNNetworkMigrationAnalyzerEC2ResourceTag",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInsightsPath"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "AWSApplicationMigrationService"
                }
            }
        },
        {
            "Sid": "MGNNetworkMigAnalyzerEC2RequestTag",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInsightsPath",
                "ec2:StartNetworkInsightsAnalysis"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "AWSApplicationMigrationService"
```

```
                }
            }
        },
        {
            "Sid": "MGNNetworkMigrationAnalyzeNetwork",
            "Effect": "Allow",
            "Action": [
                "ec2:StartNetworkInsightsAnalysis"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

ユーザーアクセス許可

次のポリシーでは、、、および AWS Documentation サイトを含む AWS アプリケーション AWS Management Console AWS Console Mobile Applicationとウェブサイトで Amazon Q Developer の機能にアクセスすることを許可します。

Amazon Q Developer への管理アクセスを有効にするポリシーについては、「管理者のアクセス許可」を参照してください。

> **Note**
>
> IDE またはコマンドラインで Amazon Q にアクセスするユーザーには、IAM アクセス許可は必要ありません。

Amazon Q Developer Pro サブスクリプションを利用した Amazon Q へのアクセスをユーザーに許可する

次のポリシー例では、Amazon Q Developer Pro サブスクリプションで Amazon Q を使用するアクセス許可を付与します。これらのアクセス許可がないと、ユーザーは Amazon Q の無料利用枠にしかアクセスできません。Amazon Q とチャットしたり、他の Amazon Q 機能を使用するには、このセクションのポリシー例で付与されているアクセス許可など、追加のアクセス許可が必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowGetIdentity",
            "Effect": "Allow",
            "Action": [
                "q:GetIdentityMetaData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSetTrustedIdentity",
            "Effect": "Allow",
            "Action": [
                "sts:SetContext"
            ],
            "Resource": "arn:aws:sts::*:self"
        }
    ]
}
```

Amazon Q にカスタマーマネージドキーへのアクセスを許可する

次のポリシー例では、Amazon Q にキーへのアクセスを許可することで、カスタマーマネージドキーで暗号化された機能にアクセスするためのアクセス許可をユーザーに付与します。このポリシーは、管理者が暗号化用にカスタマーマネージドキーをセットアップしている場合に Amazon Q を使用するために必要です。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "QKMSDecryptGenerateDataKeyPermissions",
            "Effect": "Allow",
            "Action": [
```

```
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:{111122223333}:key/[[key_id]]"
            ],
            "Condition": {
                "StringLike": {
                    "kms:ViaService": [
                        "q.us-east-1.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

ユーザーに Amazon Q とのチャットを許可する

次のポリシー例では、コンソールで Amazon Q とチャットするための許可を付与します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAmazonQConversationAccess",
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation"
      ],
      "Resource": "*"
    }
```

```
  ]
}
```

AWS CloudShellで Amazon Q CLI を使用することをユーザーに許可する

次のポリシー例では、 で Amazon Q CLI を使用するためのアクセス許可を付与します AWS CloudShell。

> **Note**
>
> codewhisperer プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "codewhisperer:GenerateRecommendations",
                "codewhisperer:ListCustomizations"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "q:StartConversation",
                "q:SendMessage"
            ],
            "Resource": "*"
        }
    ]
}
```

ユーザーにコマンドラインで変換の実行を許可する

次のポリシー例では、変換用の Amazon Q コマンドラインツールを使用してコードを変換するアクセス許可を付与します。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
              "qdeveloper:StartAgentSession",
              "qdeveloper:ImportArtifact",
              "qdeveloper:ExportArtifact",
              "qdeveloper:TransformCode"
            ],
            "Resource": "*"
        }
    ]
}
```

ユーザーが Amazon Q でコンソールエラーをトラブルシューティングすることを許可する

次のポリシー例では、Amazon Q でコンソールエラーをトラブルシューティングするためのアクセス許可を付与します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAmazonQTroubleshooting",
      "Effect": "Allow",
      "Action": [
        "q:StartTroubleshootingAnalysis",
        "q:GetTroubleshootingResults",
```

```
        "q:StartTroubleshootingResolutionExplanation",
        "q:UpdateTroubleshootingCommandResult",
        "q:PassRequest",
        "cloudformation:GetResource"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Q を使用して CLI コマンドからコードを生成することをユーザーに許可する

次のポリシー例では、記録された CLI コマンドから Amazon Q を使用してコードを生成するためのアクセス許可を付与します。これにより、Console-to-Code 機能を使用できます。

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
         "Sid": "AllowAmazonQConsoleToCode",
         "Effect": "Allow",
         "Action": "q:GenerateCodeFromCommands",
         "Resource": "*"
       }
   ]
}
```

ユーザーに Amazon Q とのリソースに関するチャットを許可する

次のポリシー例では、リソースについて Amazon Q とチャットするアクセス許可を付与し、Amazon Q がユーザーに代わってリソース情報を取得できるようにします。Amazon Q には、IAM ID がアクセス許可を持つリソースへのアクセス許可のみがあります。

JSON

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

```
  "Statement": [
    {
      "Sid": "AllowAmazonQPassRequest",
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowCloudControlReadAccess",
      "Effect": "Allow",
      "Action": [
         "cloudformation:GetResource",
         "cloudformation:ListResources"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Q がチャットでユーザーに代わってアクションを実行することを許可する

次のポリシー例では、Amazon Q とチャットするアクセス許可を付与し、Amazon Q がユーザーに代わってアクションを実行することを許可します。Amazon Q には、IAM ID が実行するアクセス許可を持つアクションを実行するためのアクセス許可のみが与えられます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAmazonQPassRequest",
      "Effect": "Allow",
```

```
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Q がコストデータにアクセスし、コスト最適化に関する推奨事項を提供できるようにする

次のポリシー例では、コストについて Amazon Q とチャットするアクセス許可を付与し、Amazon Q がコストデータにアクセスし、コスト分析と最適化のレコメンデーションを提供できるようにします。このポリシーには、 AWS Cost Explorer、、 AWS Cost Optimization Hubおよび に関連するアクセス許可が含まれます AWS Compute Optimizer。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAmazonQChatAndPassRequest",
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*"
    },
    {
```

```
      "Sid": "AllowCostAnalysis",
      "Effect": "Allow",
      "Action": [
        "ce:GetCostAndUsage",
        "ce:GetCostForecast",
        "ce:GetTags",
        "ce:GetCostCategories",
        "ce:GetDimensionValues"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowCostOptimization",
      "Effect": "Allow",
      "Action": [
        "cost-optimization-hub:GetRecommendation",
        "cost-optimization-hub:ListRecommendations",
        "cost-optimization-hub:ListRecommendationSummaries",
        "compute-optimizer:GetAutoScalingGroupRecommendations",
        "compute-optimizer:GetEBSVolumeRecommendations",
        "compute-optimizer:GetEC2InstanceRecommendations",
        "compute-optimizer:GetECSServiceRecommendations",
        "compute-optimizer:GetRDSDatabaseRecommendations",
        "compute-optimizer:GetLambdaFunctionRecommendations",
        "compute-optimizer:GetIdleRecommendations",
        "compute-optimizer:GetEffectiveRecommendationPreferences",
        "ce:GetReservationPurchaseRecommendation",
        "ce:GetSavingsPlansPurchaseRecommendation"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Q がユーザーに代わって特定のアクションを実行するためのアクセス許可を拒否する

次のポリシー例では、Amazon Q とチャットするアクセス許可を付与し、Amazon EC2 アクションを除き、IAM ID が実行するアクセス許可を持つアクションをユーザーに代わって実行することを Amazon Q に許可します。このポリシーは、aws:CalledVia グローバル条件キーを使用して、Amazon Q が呼び出す場合にのみ Amazon EC2 アクションが拒否されるように指定します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Deny",
      "Action": [
        "ec2:*"
      ],
      "Resource": "*",
      "Condition": {
            "ForAnyValue:StringEquals": {
               "aws:CalledVia": ["q.amazonaws.com"]
            }
       }
    }
  ]
}
```

Amazon Q のアクセス許可でユーザーに代わって特定のアクションを実行することを許可する

次のポリシー例では、Amazon Q とチャットするアクセス許可を付与し、Amazon EC2 アクションを除き、IAM ID が実行するアクセス許可を持つアクションをユーザーに代わって実行することを Amazon Q に許可します。このポリシーは、Amazon EC2 アクションを実行するアクセス許可を IAM ID に付与しますが、Amazon Q は `ec2:describeInstances` アクションしか実行できません。このポリシーは、aws:CalledVia グローバル条件キーを使用して、Amazon Q には

ec2:describeInstances を呼び出すことのみが許可され、他の Amazon EC2 アクションを呼び出すことは許可されないように指定します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:*"
      ],
      "Resource": "*",
      "Condition": {
            "ForAnyValue:StringNotEquals": {
               "aws:CalledVia": ["q.amazonaws.com"]
            }
       }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:describeInstances"
      ],
      "Resource": "*",
       "Condition": {
            "ForAnyValue:StringEquals": {
               "aws:CalledVia": ["q.amazonaws.com"]
            }
```

```
        }
      }
    ]
}
```

Amazon Q に特定のリージョンでユーザーに代わってアクションを実行するアクセス許可を付与する

次のポリシーの例では、Amazon Q とチャットするアクセス許可を付与し、ユーザーに代わってアクションを実行するときに Amazon Q が us-east-1 リージョンと us-west-2 リージョンに対してのみ呼び出しを行うことを許可します。Amazon Q は他のリージョンに呼び出しを行うことはできません。呼び出し先リージョンを指定する方法の詳細については、「AWS Identity and Access Management ユーザーガイド」の「aws:RequestedRegion」を参照してください。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "q:StartConversation",
        "q:SendMessage",
        "q:GetConversation",
        "q:ListConversations",
        "q:UpdateConversation",
        "q:DeleteConversation",
        "q:PassRequest"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
            "aws:RequestedRegion": [
                "us-east-1",
                "us-west-2"
            ]
        }
      }
    }
  ]
```

```
}
```

Amazon Q のアクセス許可でユーザーに代わってアクションを実行することを拒否する

次のポリシー例では、Amazon Q がユーザーに代わってアクションを実行できないようにします。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAmazonQPassRequest",
      "Effect": "Deny",
      "Action": [
        "q:PassRequest"
      ],
      "Resource": "*"
    }
  ]
}
```

ユーザーが 1 つのプロバイダーのプラグインとチャットできるようにする

次のポリシー例では、プラグインプロバイダーの名前とワイルドカード文字 () を使用してプラグイン ARN で指定された、管理者が設定した特定のプロバイダーのプラグインとチャットするアクセス許可を付与します*。プラグインを削除して再設定した場合、これらのアクセス許可を持つユーザーは、新しく設定されたプラグインへのアクセスを保持します。このポリシーを使用するには、Resourceフィールドの ARN で以下を置き換えます。

- *AWS-region* – プラグインが作成された AWS リージョン 場所。
- *AWS-account-ID* – プラグインが設定されている AWS アカウントのアカウント ID。
- *plugin-provider* – CloudZero、、 Datadogなど、アクセスを許可するプラグインプロバイダーの名前Wiz。プラグインプロバイダーフィールドでは、大文字と小文字が区別されます。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAmazonQConversationAccess",
            "Effect": "Allow",
            "Action": [
                "q:StartConversation",
                "q:SendMessage",
                "q:GetConversation",
                "q:ListConversations",
                "q:UpdateConversation",
                "q:DeleteConversation"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowAmazonQPluginAccess",
            "Effect": "Allow",
            "Action": [
                "q:UsePlugin"
            ],
            "Resource": "arn:aws:qdeveloper:us-east-1:AWS-account-
ID:plugin/plugin-provider/*"
        }
    ]
}
```

ユーザーに特定のプラグインとのチャットを許可する

次のポリシー例では、プラグイン ARN で指定された特定のプラグインとチャットするアクセス許可を付与します。プラグインを削除して再設定すると、このポリシーでプラグイン ARN が更新されない限り、ユーザーは新しいプラグインにアクセスできなくなります。このポリシーを使用するには、Resourceフィールドの ARN で以下を置き換えます。

- *AWS-region* – プラグインが作成された AWS リージョン 場所。
- *AWS-account-ID* – プラグインが設定されているアカウントのアカウント AWS ID。

- *plugin-provider* – CloudZero、、 Datadogなど、アクセスを許可するプラグインプロバイダーの名前Wiz。プラグインプロバイダーフィールドでは、大文字と小文字が区別されます。
- *plugin-ARN* – アクセスを許可するプラグインの ARN。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAmazonQConversationAccess",
            "Effect": "Allow",
            "Action": [
                "q:StartConversation",
                "q:SendMessage",
                "q:GetConversation",
                "q:ListConversations",
                "q:UpdateConversation",
                "q:DeleteConversation"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowAmazonQPluginAccess",
            "Effect": "Allow",
            "Action": [
                "q:UsePlugin"
            ],
            "Resource": "arn:aws:qdeveloper:AWS-region:AWS-account-
ID:plugin/plugin-provider/plugin-ARN"

        }
    ]
}
```

Amazon Q へのアクセスの拒否

次のポリシー例では、コンソールで Amazon Q を使用するすべてのアクセス許可を拒否します。

> **Note**
>
> Amazon Q へのアクセスを拒否しても、Amazon Q アイコンとチャットパネルは AWS コンソール、 AWS ウェブサイト、 AWS ドキュメントページ、または に表示されます AWS Console Mobile Application。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAmazonQFullAccess",
      "Effect": "Deny",
      "Action": [
        "q:*"
      ],
      "Resource": "*"
    }
  ]
}
```

自分のアクセス許可の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
```

```
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

サードパーティー統合のための Amazon Q Developer へのアクセスを管理する

サードパーティーの統合では、アイデンティティベースでもリソースベースでもない IAM ポリシーではなく、 AWS Key Management Service (KMS) を使用して Amazon Q Developer へのアクセスを管理する必要があります。

管理者がカスタマーマネージドキーを使用してロールポリシーを更新できるようにする

次のキーポリシーの例では、KMS コンソールで設定されたロールにキーポリシーを作成するときに、カスタマーマネージドキー (CMK) を使用するアクセス許可を付与します。CMK を設定するときは、Amazon Q を呼び出すために統合で使用される識別子である IAM ロール ARN を指定する必要があります。GitLab インスタンスなどの統合を既にオンボードしている場合は、CMK で暗号化されるすべてのリソースについてインスタンスを再オンボードする必要があります。

`kms:ViaService` 条件キーは、指定された AWS サービスからのリクエストに KMS キーの使用を制限します。さらに、リクエストが特定のサービスから送信されたときに KMS キーを使用するアク

セス許可を拒否するために使用されます。条件キーを使用すると、コンテンツの暗号化または復号に CMK を使用できるユーザーを制限できます。詳細については、AWS Key Management Service デベロッパーガイドのkms:Via ViaServiceService」を参照してください。 AWS Key Management Service

KMS 暗号化コンテキストでは、対称暗号化 KMS キーを使用して暗号化オペレーションに含めることができるオプションのキーと値のペアのセットがあり、認可と監査性を強化できます。暗号化コンテキストを使用して、暗号化されたデータの整合性と信頼性を検証し、キーポリシーと IAM ポリシーで対称暗号化 KMS キーへのアクセスを制御し、AWS CloudTrail ログで暗号化オペレーションを特定および分類できます。詳細については、AWS Key Management Service デベロッパーガイドの「暗号化コンテキスト」を参照してください。

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Sid0",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/rolename"
            },
            "Action": [
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:Decrypt",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo",
                "kms:GenerateDataKey",
                "kms:Encrypt",
                "kms:DescribeKey"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "q.us-east-1.amazonaws.com",
                    "kms:EncryptionContext:aws-crypto-
ec:aws:qdeveloper:accountId": "accountId"
                }
            }
        }
```

```
    ]
}
```

Amazon Q Developer のアクセス許可リファレンス

Amazon Q Developer は、次の 2 種類の API を使用してサービスを提供します。

- ユーザーおよび管理者のアクセス許可。これは、Amazon Q の使用を制御するためのポリシーで使用できます。
- サービスの提供に使用される他の API。Amazon Q の使用を制御するためのポリシーでは使用できません。

このセクションでは、Amazon Q Developer が使用する API とその動作について説明します。

トピック

- Amazon Q Developer のアクセス許可
- Amazon Q ユーザーサブスクリプション関連のアクセス許可
- その他の Amazon Q Developer API

Amazon Q Developer のアクセス許可

Amazon Q でのアイデンティティによる認証をセットアップし、IAM ID (アイデンティティベースのポリシー) にアタッチできる許可ポリシーを作成する際、以下のアクセス許可をリファレンスとして使用できます。

以下の表は、ポリシーでアクセスを付与または拒否できる Amazon Q Developer のアクセス許可を示しています。

> **Important**
>
> Amazon Q とチャットするには、IAM ID に次のアクションのためのアクセス許可が必要です。
>
> - `StartConversation`
> - `SendMessage`
> - `GetConversation` (コンソールのみ)

- `ListConversations` (コンソールのみ)

このいずれかのアクションがアタッチされたポリシーによって明示的に許可されていない場合、Amazon Q とのチャット時に IAM アクセス許可エラーが返されます。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

q:PassRequest の使用

`q:PassRequest` は、Amazon Q がユーザーに代わって AWS APIs を呼び出すことを許可する Amazon Q アクセス許可です。IAM ID に `q:PassRequest` アクセス許可を追加すると、Amazon Q は IAM ID が呼び出すアクセス許可を持つ API を呼び出すアクセス許可を取得します。例えば、IAM ロールに `s3:ListAllMyBuckets` アクセス許可と `q:PassRequest` アクセス許可がある場合、IAM ロールを引き受けるユーザーが Amazon Q に Amazon S3 バケットを一覧表示するようにリクエストすると、Amazon Q は `ListAllMyBuckets` API を呼び出すことができます。

`q:PassRequest` アクセス許可の範囲を制限する IAM ポリシーを作成できます。例えば、Amazon Q が特定のアクションを実行できないようにしたり、Amazon Q がサービスに対して一部のアクションのみの実行を許可したりすることができます。また、ユーザーに代わってアクションを実行する際に Amazon Q が呼び出すことができるリージョンを指定することもできます。

`q:PassRequest` の使用を制御する IAM ポリシーの例については、次のアイデンティティベースのポリシーの例を参照してください。

- Amazon Q がチャットでユーザーに代わってアクションを実行することを許可する
- Amazon Q がユーザーに代わって特定のアクションを実行するためのアクセス許可を拒否する
- Amazon Q のアクセス許可でユーザーに代わって特定のアクションを実行することを許可する
- Amazon Q に特定のリージョンでユーザーに代わってアクションを実行するアクセス許可を付与する
- Amazon Q のアクセス許可でユーザーに代わってアクションを実行することを拒否する

Amazon Q ユーザーサブスクリプション関連のアクセス許可

Amazon Q Developer 管理者は、組織内のユーザーとグループのサブスクリプションを作成および管理するために、次のアクセス許可が必要です。

次の用語は、サブスクリプションのアクセス許可の内容を理解するのに役立ちます。

ユーザー

内の一意のユーザー ID AWS IAM Identity Center で表される個々のユーザー。

[Group] (グループ)

一意のグループ ID AWS IAM Identity Center で表される 内のユーザーのコレクション。

サブスクリプション

サブスクリプションは 1 人のアイデンティティセンターユーザーに関連付けられ、Amazon Q 機能を使用する権限がそのユーザーに付与されます。サブスクリプションでは、ユーザーが Amazon Q 機能を使用することは許可されません。例えば、Adam が Amazon Q Developer Pro にサブスクライブしている場合、Adam には Amazon Q Developer の機能を使用する権限がありますが、管理者が必要なアクセス許可を付与するまで、これらの機能にアクセスすることはできません。

その他の Amazon Q Developer API

次の表は、IDE で Amazon Q の機能で使用される API を示しています。これらの APIs は、Amazon Q の機能へのアクセスを制御するために使用されませんが、ユーザーが関連する機能にアクセスすると、管理アカウントの AWS CloudTrail ログに表示されます。

> **Note**
>
> `codewhisperer` プレフィックスは、Amazon Q Developer とマージされたサービスのレガシー名です。詳細については、「Amazon Q Developer の名称変更 - 変更の概要」を参照してください。

AWS Amazon Q Developer の マネージドポリシー

AWS 管理ポリシーは、 によって作成および管理されるスタンドアロンポリシーです AWS。 AWS 管理ポリシーは、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できるように、多くの一般的なユースケースにアクセス許可を付与するように設計されています。

管理者がユーザーにアクセス権を付与する最も簡単な方法は、 AWS マネージドポリシーを使用することです。Amazon Q Developer の以下の AWS 管理ポリシーを IAM ID にアタッチできます。

- `AmazonQFullAccess` は、管理者アクセスを含む、Amazon Q Developer とのやり取りを可能にするフルアクセスを、管理者アクセスを含めて提供します。
- `AmazonQDeveloperAccess` は、Amazon Q Developer とのやり取りを可能にするフルアクセスを、管理者アクセスなしで提供します。

> **Note**
>
> IDE またはコマンドラインで Amazon Q にアクセスするユーザーには、IAM アクセス許可は必要ありません。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケースに固有のカスタマー管理ポリシーを定義して、アクセス許可を絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS マネージドポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。 AWS は、新しい が起動されるか、新しい API オペレーション AWS のサービス が既存のサービスで使用できるようになったときに、 AWS マネージドポリシーを更新する可能性が最も高くなります。

詳細については「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

AmazonQFullAccess

`AmazonQFullAccess` マネージドポリシーは、組織のユーザーが Amazon Q Developer にアクセスすることを許可する管理者アクセスを提供します。また、IAM アイデンティティセンターでのログインや Amazon Q Developer Pro サブスクリプションによる Amazon Q へのアクセスなど、Amazon Q Developer とのやり取りを可能にするためのフルアクセスも提供します。

> **Note**
>
> Amazon Q サブスクリプション管理コンソールと Amazon Q Developer Pro コンソールで管理タスクを完了するためのフルアクセスを有効にするには、追加のアクセス許可が必要です。詳細については、「管理者のアクセス許可」を参照してください。

このポリシーのアクセス許可を確認するには、AWS マネージドポリシーリファレンスのAmazonQFullAccess」を参照してください。

AmazonQDeveloperAccess

`AmazonQDeveloperAccess` マネージドポリシーは、Amazon Q Developer とのやり取りを可能にするフルアクセスを、管理者アクセスなしで提供します。これには、Amazon Q Developer Pro サブスクリプションを通じて Amazon Q にアクセスするための IAM アイデンティティセンターへのログインアクセスが含まれます。

Amazon Q の一部の機能を使用するために、追加の許可が必要な場合があります。アクセス許可の詳細については、使用する機能のトピックを参照してください。

このポリシーのアクセス許可を確認するには、AWS マネージドポリシーリファレンスのAmazonQDeveloperAccess」を参照してください。

AWSServiceRoleForAmazonQDeveloper

この AWS 管理ポリシーは、Amazon Q Developer を使用するために一般的に必要なアクセス許可を付与します。このポリシーは、Amazon Q にオンボーディングするときに作成される AWSServiceRoleForAmazonQDeveloper サービスリンクロールに追加されます。

AWSServiceRoleForAmazonQDeveloper を IAM エンティティにアタッチすることはできません。このポリシーは、Amazon Q がユーザーに代わってアクションを実行することを許可するサービスリンクロールにアタッチされます。詳細については、「Amazon Q Developer およびユーザーサブスクリプションでサービスリンクロールを使用する」を参照してください。

このポリシーは、請求/使用状況に対してメトリクスを発行する許可を###に付与します。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `cloudwatch` — プリンシパルが CloudWatch に請求 / 使用状況のメトリクスを発行することを許可します。これは、CloudWatch で Amazon Q の使用状況を追跡するために必要です。

このポリシーのアクセス許可を確認するには、AWS マネージドポリシーリファレンスのAWSServiceRoleForAmazonQDeveloper」を参照してください。

AWSServiceRoleForUserSubscriptions

この AWS 管理ポリシーは、Amazon Q Developer を使用するために一般的に必要なアクセス許可を付与します。このポリシーは、Amazon Q サブスクリプションの作成時に作成される AWSServiceRoleForUserSubscriptions サービスリンクロールに追加されます。

AWSServiceRoleForUserSubscriptions を IAM エンティティにアタッチすることはできません。このポリシーは、Amazon Q がユーザーに代わってアクションを実行することを許可するサービスリンクロールにアタッチされます。詳細については、「Amazon Q Developer およびユーザーサブスクリプションでサービスリンクロールを使用する」を参照してください。

このポリシーは、Amazon Q サブスクリプションがアイデンティティセンターのリソースにアクセスし、サブスクリプションを自動的に更新できるようにします。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `identitystore` – サブスクリプションを自動的に更新できるように、プリンシパルにアイデンティティセンターのディレクトリの変更追跡を許可します。

  `organizations` – サブスクリプションを自動的に更新できるように、プリンシパルに AWS Organizations の変更追跡を許可します。

  `sso` – サブスクリプションを自動的に更新できるように、プリンシパルにアイデンティティセンターのインスタンスの変更追跡を許可します。

このポリシーのアクセス許可を確認するには、AWS マネージドポリシーリファレンスのAWSServiceRoleForUserSubscriptions」を参照してください。

GitLabDuoWithAmazonQPermissionsPolicy

このポリシーは、Amazon Q に接続し、Amazon Q との統合で の機能を利用するアクセス許可を付与GitLab Duoします。このポリシーは、Amazon Q にアクセスするために Amazon Q Developer コンソールから作成された IAM ロールに追加されます。IAM ロールを Amazon リソースネーム (ARN) として GitLab に手動で提供する必要があります。このポリシーでは、以下が許可されます。

- GitLab Duo 使用許可 - イベントとメッセージの送信、認証許可の作成と更新、コードレコメンデーションの生成、プラグインの一覧表示、OAuth アプリ接続の検証などの基本的な操作を許可します。
- GitLab Duo 管理アクセス許可 - OAuth アプリ接続の作成と削除を有効にし、統合設定を制御します。
- GitLab Duo プラグインのアクセス許可 - Amazon Q とGitLab Duoの統合に関連するプラグインを作成、削除、取得するための特定のアクセス許可を付与します。

このポリシーのアクセス許可を確認するには、AWS マネージドポリシーリファレンスのGitLabDuoWithAmazonQPermissionsPolicy」を参照してください。

ポリシーの更新

このサービスがこれらの変更の追跡を開始してからの Amazon Q Developer の AWS マネージドポリシーの更新に関する詳細を表示します。このページへの変更に関する自動アラートを受け取るには、「Amazon Q Developer ユーザーガイドのドキュメント履歴」ページで RSS フィードにサブスクライブしてください。

変更	説明	日付
AmazonQDeveloperAccess – ポリシーを更新	Amazon Q チャットの会話履歴を管理するためのアクセス許可が追加されました。	2025 年 5 月 14 日
AmazonQFullAccess – ポリシーを更新	Amazon Q チャットの会話履歴を管理するためのアクセス許可が追加されました。	2025 年 5 月 14 日
AmazonQFullAccess – ポリシーを更新	サードパーティーの統合プラグインの機能有効化コントロールを更新するためのアクセス許可が追加されました。	2025 年 5 月 2 日

変更	説明	日付
AmazonQFullAccess – ポリシーを更新	サードパーティープラグインの Amazon Q Developer へのアクセスを許可し、インタラクションを有効にするためのアクセス許可が追加されました。	2025 年 4 月 30 日
GitLabDuoWithAmazonQPermissionsPolicy – ポリシーを更新	Amazon Q Developer でサードパーティーの OAuth アプリケーションの更新を許可する追加のアクセス許可が追加されました。	2025 年 4 月 30 日
GitLabDuoWithAmazonQPermissionsPolicy – 新しいポリシー	GitLab が Amazon Q Developer に接続して Amazon Q 統合機能 GitLab Duoで使用できるようにします。	2025 年 4 月 17 日
AWSServiceRoleForUserSubscriptions – ポリシーを更新	Amazon Q がエンドユーザーの E メール検証ステータスを検出できるようにします。	2025 年 2 月 17 日
AmazonQDeveloperAccess – ポリシーを更新	Amazon Q Developer プラグインの使用を有効にするためのアクセス許可が追加されました。	2024 年 11 月 13 日
AmazonQFullAccess – ポリシーを更新	Amazon Q Developer プラグインを設定して使用し、Amazon Q Developer リソースのタグを作成および管理するためのアクセス許可が追加されました。	2024 年 11 月 13 日
AmazonQDeveloperAccess – ポリシーを更新	Amazon Q で CLI コマンドからのコード生成を有効にするためのアクセス許可が追加されました。	2024 年 10 月 28 日

変更	説明	日付
AmazonQFullAccess – ポリシーを更新	Amazon Q で CLI コマンドからのコード生成を有効にするためのアクセス許可が追加されました。	2024 年 10 月 28 日
AmazonQFullAccess – ポリシーを更新	Amazon Q がダウンストリームリソースにアクセスできるようにするためのアクセス許可が追加されました。	2024 年 7 月 9 日
AmazonQDeveloperAccess – 新しいポリシー	管理者アクセスなしで Amazon Q Developer とのやり取りを有効にするためのフルアクセスを提供します。	2024 年 7 月 9 日
AmazonQFullAccess – ポリシーを更新	Amazon Q Developer のサブスクリプションチェックを有効にするためのアクセス許可が追加されました。	2024 年 4 月 30 日
AWSServiceRoleForUserSubscriptions – 新しいポリシー	Amazon Q Subscriptions が、 AWS Organizations ユーザーに代わって AWS IAM Identity Center、 AWS IAM アイデンティティセンターディレクトリ の変更からサブスクリプションを自動的に更新できるようにします。	2024 年 4 月 30 日
AWSServiceRoleForAmazonQDeveloper – 新しいポリシー	Amazon Q がユーザーに代わって Amazon CloudWatch と Amazon CodeGuru を呼び出すことを許可します。	2024 年 4 月 30 日
AmazonQFullAccess – 新しいポリシー	Amazon Q Developer とのやり取りを可能にするフルアクセスを提供します。	2023 年 11 月 28 日
Amazon Q Developer が変更の追跡を開始	Amazon Q Developer は、 AWS 管理ポリシーの変更の追跡を開始しました。	2023 年 11 月 28 日

Amazon Q Developer およびユーザーサブスクリプションでサービスリンクロールを使用する

Amazon Q Developer は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスリンクロールは、Amazon Q Developer に直接リンクされた特殊なタイプの IAM ロールです。サービスにリンクされたロールは Amazon Q Developer によって事前定義されており、サービスがユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれています。

トピック

- Amazon Q Developer でサービスリンクロールを使用する
- ユーザーサブスクリプションでサービスリンクロールを使用する

Amazon Q Developer でサービスリンクロールを使用する

Amazon Q Developer は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスリンクロールは、Amazon Q Developer に直接リンクされた特殊なタイプの IAM ロールです。サービスにリンクされたロールは Amazon Q Developer によって事前定義されており、サービスがユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれています。

サービスリンクロールを使用すると、必要な許可を手動で追加する必要がないため、Amazon Q Developer のセットアップが簡単になります。サービスリンクロールの許可は Amazon Q Developer が定義し、特に定義されない限り、Amazon Q Developer のみがそのロールを引き受けることができます。定義される許可は信頼ポリシーと許可ポリシーに含まれており、その許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスリンクロールを削除するには、最初に関連リソースを削除する必要があります。リソースにアクセスするための許可を誤って削除できないため、これにより、 Amazon Q Developer リソースが保護されます。

サービスにリンクされたロールをサポートする他のサービスの詳細については、AWS「IAM と連携するサービス」を参照し、「サービスにリンクされたロール」列で「はい」を持つサービスを探します。サービスリンクロールに関するドキュメントをサービスで表示するには、リンクで [はい] を選択します。

AWS Amazon Q Developer の マネージドポリシー　についてはこちら。

Amazon Q Developer のサービスリンクロールのアクセス許可

Amazon Q Developer は、AWSServiceRoleForAmazonQDeveloper という名前のサービスリンクロールを使用します。このロールは、アカウントのデータにアクセスして請求を計算するためのアクセス許可を Amazon Q に付与するほか、Amazon CodeGuru でセキュリティレポートを作成してそれにアクセスし、データを CloudWatch に出力するためのアクセス権を提供します。

AWSServiceRoleForAmazonQDeveloper サービスリンクロールは、以下のサービスを信頼してロールを引き受けます。

- `q.amazonaws.com`

AWSServiceRoleForAmazonQDeveloper という名前のロールアクセス許可ポリシーにより、Amazon Q Developer は指定されたリソースに対して次のアクションを実行できます。

- アクション: `cloudwatch:PutMetricData`。対象リソース: `AWS/Q CloudWatch namespace`

ユーザー、グループ、またはロールにサービスリンクロールの作成、編集、または削除を許可するには、アクセス許可を設定する必要があります。詳細については、「IAM ユーザーガイド」の「サービスリンクロールのアクセス許可」を参照してください。

Amazon Q Developer のサービスリンクロールを作成する

サービスリンクロールを手動で作成する必要はありません。で Amazon Q のプロファイルを作成すると AWS Management Console、Amazon Q Developer によってサービスにリンクされたロールが作成されます。

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は同じ方法でアカウントにロールを再作成できます。設定を更新すると、Amazon Q によって、サービスリンクロールが再度作成されます。

IAM コンソールまたは AWS CLI を使用して、`q.amazonaws.com` というサービス名でサービスリンクロールを作成できます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの作成」を参照してください。このサービスリンクロールを削除しても、同じ方法でロールを再作成できます。

Amazon Q Developer のサービスリンクロールを編集する

Amazon Q Developer では、AWSServiceRoleForAmazonQDeveloper サービスリンクロールを編集できません。サービスリンクロールの作成後は、さまざまなエンティティがロールを参照する可能

性があるため、ロール名を変更することはできません。ただし、IAM を使用してロールの説明を編集することはできます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの編集」を参照してください。

Amazon Q Developer のサービスリンクロールを削除する

サービスリンクロールを必要とする機能やサービスが不要になった場合は、ロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスリンクロールのリソースをクリーンアップする必要があります。

> **Note**
>
> リソースを削除しようとしたときに Amazon Q Developer サービスでロールが使用されている場合は、削除が失敗する可能性があります。その場合は、数分待ってからオペレーションを再試行してください。

IAM を使用してサービスリンクロールを手動で削除するには

IAM コンソール、 AWS CLI、または AWS API を使用して、AWSServiceRoleForAmazonQDeveloper サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

Amazon Q Developer サービスリンクロールがサポートされているリージョン

Amazon Q Developer は、サービスを利用できるすべてのリージョンで、サービスリンクロールの使用をサポートしているわけではありません。AWSServiceRoleForAmazonQDeveloper ロールは、以下のリージョンで使用できます。詳細については、「AWS リージョンとエンドポイント」を参照してください。

リージョン名	リージョン識別子	Amazon Q Developer でのサポート
米国東部 (バージニア北部)	us-east-1	はい
米国東部 (オハイオ)	us-east-2	なし
米国西部 (北カリフォルニア)	us-west-1	なし

リージョン名	リージョン識別子	Amazon Q Developer でのサポート
米国西部 (オレゴン)	us-west-2	なし
アフリカ (ケープタウン)	af-south-1	いいえ
アジアパシフィック (香港)	ap-east-1	いいえ
アジアパシフィック (ジャカルタ)	ap-southeast-3	いいえ
アジアパシフィック (ムンバイ)	ap-south-1	なし
アジアパシフィック (大阪)	ap-northeast-3	いいえ
アジアパシフィック (ソウル)	ap-northeast-2	なし
アジアパシフィック (シンガポール)	ap-southeast-1	なし
アジアパシフィック (シドニー)	ap-southeast-2	なし
アジアパシフィック (東京)	ap-northeast-1	なし
カナダ (中部)	ca-central-1	なし
欧州 (フランクフルト)	eu-central-1	なし
欧州 (アイルランド)	eu-west-1	なし
欧州 (ロンドン)	eu-west-2	なし
欧州 (ミラノ)	eu-south-1	いいえ
欧州 (パリ)	eu-west-3	なし
欧州 (ストックホルム)	eu-north-1	なし
中東 (バーレーン)	me-south-1	なし
中東 (アラブ首長国連邦)	me-central-1	なし
南米 (サンパウロ)	sa-east-1	なし

リージョン名	リージョン識別子	Amazon Q Developer でのサポート
AWS GovCloud (米国東部)	us-gov-east-1	なし
AWS GovCloud (米国西部)	us-gov-west-1	いいえ

ユーザーサブスクリプションでサービスリンクロールを使用する

ユーザーサブスクリプションは AWS Identity and Access Management 、(IAM) サービスにリンクされたロールを使用します。サービスリンクロールは、ユーザーサブスクリプションに直接リンクされた特殊なタイプの IAM ロールです。サービスリンクロールは、ユーザーサブスクリプションによって事前定義されており、サービスがユーザーに代わって他の AWS のサービスを呼び出すために必要な、すべてのアクセス許可が付与されています。

サービスリンクロールを使用することで、必要なアクセス許可を手動で追加する必要がなくなるため、ユーザーサブスクリプションの設定が簡単になります。サービスリンクロールのアクセス許可はユーザーサブスクリプションが定義します。特に定義されている場合を除き、ユーザーサブスクリプションのみがそのロールを引き受けることができます。定義される許可は信頼ポリシーと許可ポリシーに含まれており、その許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスリンクロールを削除するには、最初に関連リソースを削除する必要があります。リソースに必要なアクセス許可を誤って削除できないため、これによりユーザーサブスクリプションが保護されます。

サービスにリンクされたロールをサポートする他のサービスの詳細については、AWS「IAM と連携するサービス」を参照し、「サービスにリンクされたロール」列で「はい」があるサービスを探します。サービスリンクロールに関するドキュメントをサービスで表示するには、リンクで [はい] を選択します。

ユーザーサブスクリプションのサービスリンクロールのアクセス許可

ユーザーサブスクリプションは、AWSServiceRoleForUserSubscriptions という名前のサービスにリンクされたロールを使用します。このロールは、サブスクリプションを自動的に更新するために、IAM Identity Center リソースへのユーザーサブスクリプションへのアクセスを提供します。

AWSServiceRoleForUserSubscriptions のサービスリンクロールは、以下のサービスを信頼してロールを引き受けます。

- user-subscriptions.amazonaws.com

AWSServiceRoleForUserSubscriptions という名前のロールアクセス許可ポリシーにより、ユーザーサブスクリプションは指定されたリソースに対して次のアクションを実行できます。

- アクション: identitystore:DescribeGroup。対象リソース: *

  アクション: * 上で identitystore:DescribeUser

  アクション: * 上で identitystore:IsMemberInGroups

  アクション: * 上で identitystore:ListGroupMemberships

  アクション: * 上で organizations:DescribeOrganization

  アクション: * 上で sso:DescribeApplication

  アクション: * 上で sso:DescribeInstance

  アクション: * 上で sso:ListInstances

  アクション: sso-directory:DescribeUser。対象リソース: *

ユーザー、グループ、またはロールにサービスリンクロールの作成、編集、または削除を許可するには、アクセス許可を設定する必要があります。詳細についてはIAM ユーザーガイド の「サービスにリンクされた役割のアクセス許可」を参照してください。

ユーザーサブスクリプションのサービスリンクロールを作成する

サービスリンクロールを手動で作成する必要はありません。でユーザーサブスクリプションを作成すると AWS Management Console、ユーザーサブスクリプションによってサービスにリンクされたロールが作成されます。

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は同じ方法でアカウントにロールを再作成できます。設定を更新すると、ユーザーサブスクリプションによってサービスリンクロールが再度作成されます。

IAM コンソールまたは AWS CLI を使用して、q.amazonaws.com というサービス名でサービスリンクロールを作成できます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの作成」を参照してください。このサービスリンクロールを削除しても、同じ方法でロールを再作成できます。

ユーザーサブスクリプションのサービスリンクロールを編集する

ユーザーサブスクリプションでは、AWSServiceRoleForUserSubscriptions サービスリンクロールを編集できません。サービスリンクロールの作成後は、さまざまなエンティティがロールを参照する可能性があるため、ロール名を変更することはできません。ただし、IAM を使用してロールの説明を編集することはできます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの編集」を参照してください。

ユーザーサブスクリプションのサービスリンクロールを削除する

サービスリンクロールを必要とする機能やサービスが不要になった場合は、ロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスリンクロールのリソースをクリーンアップする必要があります。

> **Note**
>
> リソースを削除しようとしたときにユーザーサブスクリプションのサービスでロールが使用されている場合は、削除が失敗する可能性があります。その場合は、数分待ってからオペレーションを再試行してください。

IAM を使用してサービスリンクロールを手動で削除するには

IAM コンソール、 AWS CLI、または AWS API を使用して、AWSServiceRoleForUserSubscriptions サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

ユーザーサブスクリプションのサービスリンクロールがサポートされているリージョン

Amazon Q Developer のサブスクリプションでは、このサービスを利用できるすべてのリージョンで、サービスリンクロールの使用がサポートされます。詳細については、「AWS リージョンとエンドポイント」を参照してください。

Amazon Q Developer は、サービスを利用できるすべてのリージョンで、サービスリンクロールの使用をサポートしているわけではありません。AWSServiceRoleForUserSubscriptions ロールは、以下のリージョンで使用できます。

リージョン名	リージョン識別子	ユーザーサブスクリプションでのサポート
米国東部 (バージニア北部)	us-east-1	あり
米国西部 (オレゴン)	us-west-2	あり
米国東部 (バージニア北部)	us-east-1	はい
米国東部 (オハイオ)	us-east-2	あり
米国東部 (オハイオ)	us-east-2	はい
米国西部 (北カリフォルニア)	us-west-1	あり
アジアパシフィック (ムンバイ)	ap-south-1	はい
アジアパシフィック (大阪)	ap-northeast-3	はい
アジアパシフィック (ソウル)	ap-northeast-2	はい
アジアパシフィック (シンガポール)	ap-southeast-1	はい
アジアパシフィック (シドニー)	ap-southeast-2	はい
アジアパシフィック (東京)	ap-northeast-1	はい
カナダ (中部)	ca-central-1	はい
欧州 (フランクフルト)	eu-central-1	はい
欧州 (アイルランド)	eu-west-1	はい
欧州 (ロンドン)	eu-west-2	はい
欧州 (パリ)	eu-west-3	はい
欧州 (ストックホルム)	eu-north-1	はい
南米 (サンパウロ)	sa-east-1	あり

Amazon Q Developer のコンプライアンス検証

AWS のサービス が特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、AWS のサービス「コンプライアンスプログラムによる範囲内」を参照して、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、AWS 「 Compliance ProgramsAssurance」を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、「Downloading Reports in AWS Artifact」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービス は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。 は、コンプライアンスに役立つ以下のリソース AWS を提供します。

- セキュリティのコンプライアンスとガバナンス – これらのソリューション実装ガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスの機能をデプロイする手順を示します。
- HIPAA 対応サービスのリファレンス – HIPAA 対応サービスの一覧が提供されています。すべての AWS のサービス が HIPAA の対象となるわけではありません。
- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド – コンプライアンスの観点から責任共有モデルを理解します。このガイドは、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) など) のセキュリティコントロールを保護し、そのガイダンスに AWS のサービス マッピングするためのベストプラクティスをまとめたものです。
- 「 デベロッパーガイド」の「ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が内部プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に把握できます AWS。Security Hub では、セキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに対するコンプライアンスをチェックします。サポートされているサービスとコントロールの一覧については、Security Hub のコントロールリファレンスを参照してください。
- Amazon GuardDuty – 不審なアクティビティや悪意のあるアクティビティがないか環境をモニタリングすることで AWS アカウント、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。GuardDuty を使用すると、特定のコンプライアンスフレームワー

クで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件に対応できます。

- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

Amazon Q Developer の耐障害性

AWS グローバルインフラストラクチャは、 AWS リージョン およびアベイラビリティーゾーンを中心に構築されています。 は、低レイテンシー、高スループット、高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティーゾーン AWS リージョン を提供します。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョン およびアベイラビリティーゾーンの詳細については、AWS 「 グローバルインフラストラクチャ」を参照してください。

Amazon Q Developer のインフラストラクチャセキュリティ

マネージドサービスである Amazon Q は、 AWS グローバルネットワークセキュリティで保護されています。 AWS セキュリティサービスと がインフラストラクチャ AWS を保護する方法については、AWS 「 クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して AWS 環境を設計するには、「Security Pillar AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

AWS 公開された API コールを使用して、ネットワーク経由で Amazon Q Developer にアクセスします。クライアントは以下をサポートする必要があります。

- Transport Layer Security (TLS)。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードはJava 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストにはアクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。またはAWS Security Token Service (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

Amazon Q Developer のファイアウォール、プロキシサーバー、またはデータ境界の設定

ファイアウォール、プロキシサーバー、またはデータ境界を使用している場合は、Amazon Q が期待どおりに動作するように、次の URLs と Amazon リソースネーム (ARNs) へのトラフィックを許可リストに登録してください。

許可リストの一般的な URLs

次の URLs、 を置き換えます。

- IAM Identity Center インスタンスのディレクトリ ID *idc-directory-id-or-alias*alias。IAM アイデンティティセンターの詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM アイデンティティセンターとは) を参照してください。
- *sso-region* と IAM アイデンティティセンターインスタンスが有効になっている AWS リージョン。詳細については、「Amazon Q Developer でサポートされている IAM Identity Center リージョン」を参照してください。
- *profile-region* と Amazon Q Developer プロファイルがインストールされている AWS リージョン。Amazon Q Developer プロファイルの詳細については、Amazon Q Developer プロファイルとは「」および「」を参照してくださいQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。

[URL]	目的
idc-directory-id-or-alias.awsapps.com	認証
oidc.*sso-region* .amazonaws.com	認証
*.sso.*sso-region* .amazonaws.com	認証
*.sso-portal. *sso-region* .amazonaws.com	認証
*.aws.dev	認証
*.awsstatic.com	認証

[URL]	目的
`*.console.aws.a2z.com`	認証
`*.sso.amazonaws.com`	認証
`https://codewhisperer.us-east-1.amazonaws.com`	Amazon Q Developer 機能
`https://q.` *`profile-region`* `.amazonaws.com`	Amazon Q Developer 機能
`https://idetoolkits-hostedfiles.amazonaws.com/*`	IDE の Amazon Q Developer、設定
`https://idetoolkits.amazonwebservices.com/*`	IDE の Amazon Q Developer、エンドポイント
`q-developer-integration.us-east-1.api.aws`	IDE の Amazon Q Developer、エンドポイント
`https://aws-toolkit-language-servers.amazonaws.com/*`	IDE の Amazon Q Developer、言語処理
`https://aws-language-servers.us-east-1.amazonaws.com/*`	IDE の Amazon Q Developer、言語処理
`https://client-telemetry.us-east-1.amazonaws.com`	IDE の Amazon Q Developer、テレメトリ
`cognito-identity.us-east-1.amazonaws.com`	IDE の Amazon Q Developer、テレメトリ

許可リストの Amazon S3 バケット URLs と ARNs

一部の機能では、Amazon Q は AWS サービス所有の Amazon S3 バケットにアーティファクトをアップロードします。データ境界を使用して環境内の Amazon S3 へのアクセスを制御する場合は、対応する Amazon Q の機能を使用するために、これらのバケットへのアクセスを明示的に許可する必要があるかもしれません。

次の表に、Amazon Q がアクセスする必要がある各 Amazon S3 バケットの URL と ARN、および各バケットを使用する機能を示します。Amazon S3 へのアクセスを制御する方法に応じて、バケット URL またはバケット ARN を使用して、これらのバケットを許可リストに登録できます。

Amazon Q Developer プロファイルがインストールされている AWS リージョンでのみ、バケットを許可リストに登録する必要があります。Amazon Q Developer プロファイルの詳細については、「」を参照してくださいAmazon Q Developer プロファイルとは。

Amazon S3 バケット URL と ARN	目的
米国東部 (バージニア北部): • `https://amazonq-code-scan-us-east-1-29121b44f7b.s3.amazonaws.com/` • `arn:aws:s3:::amazonq-code-scan-us-east-1-29121b44f7b` 欧州 (フランクフルト): • `https://amazonq-code-scan-eu-central-1-9374e402cc5.s3.amazonaws.com/` • `arn:aws:s3:::amazonq-code-scan-eu-central-1-9374e402cc5`	Amazon Q コードレビューのアーティファクトをアップロードするために使用される Amazon S3 バケット Amazon Q Developer でコードを確認する
米国東部 (バージニア北部): • `https://amazonq-code-transformation-us-east-1-c6160f047e0.s3.amazonaws.com/` • `arn:aws:s3:::amazonq-code-transformation-us-east-1-c6160f047e0` 欧州 (フランクフルト):	Amazon Q Developer Agent for code transformation のアーティファクトをアップロードするために使用される Amazon S3 バケット

Amazon S3 バケット URL と ARN	目的
• https://amazonq-code-transformation-eu-central-1-a0a89cc2b94.s3.amazonaws.com/ • arn:aws:s3:::amazonq-code-transformation-eu-central-1-a0a89cc2b94	
米国東部 (バージニア北部): • https://amazonq-feature-development-us-east-1-a5b980054c6.s3.amazonaws.com/ • arn:aws:s3:::amazonq-feature-development-us-east-1-a5b980054c6 欧州 (フランクフルト): **Note** URL と ARN は、欧州 (フランクフルト) リージョンでは使用できません。回避策として、ソフトウェア開発のニーズに合わせてエージェントチャット機能を使用するようにユーザーに指示します。	Amazon Q Developer Agent for software development のアーティファクトをアップロードするために使用される Amazon S3 バケット

Amazon S3 バケット URL と ARN	目的
米国東部 (バージニア北部): • `https://amazonq-test-generation-us-east-1-74b667808f2.s3.us-east-1.amazonaws.com/` • `arn:aws:s3:::amazonq-test-generation-us-east-1-74b667808f2` 欧州 (フランクフルト): • `https://amazonq-test-generation-eu-central-1-335c4259858.s3.us-east-1.amazonaws.com/` • `arn:aws:s3:::amazonq-test-generation-eu-central-1-335c4259858`	ユニットテスト生成のために Amazon Q Developer Agent のアーティファクトをアップロードするために使用される Amazon S3 バケット

Amazon Q での企業プロキシの設定

エンドユーザーが企業プロキシの背後で作業している場合は、次の手順を実行して Amazon Q に正常に接続します。

ステップ 1: IDE でプロキシ設定を構成する

IDE でプロキシサーバーの URL を指定します。

> **Note**
>
> HTTP または HTTPS プロキシを設定する必要があります。SOCKS プロキシはサポートされていません。

Eclipse

1. でEclipse、次のように設定を開きます。
   - Windows または Ubuntu の場合：
     - Eclipse メニューバーから、Window を選択し、Preferences を選択します。
   - macOS の場合：
     - メニューバーから を選択しEclipse、macOS のバージョンに応じて設定または設定を選択します。
2. 検索バーで、「」と入力**Amazon Q**して Amazon Q を開きます。
3. プロキシ設定で、HTTPS プロキシ URL を企業プロキシ URL に設定します。

   例: `http://proxy.company.com:8080`、`https://proxy.company.com:8443`
4. Amazon Q 設定を開いたままにして、次のステップに進みます。

JetBrains

JetBrains では、IntelliJ IDEA ドキュメントの HTTP Proxy トピックのガイダンスに従って、プロキシサーバーのホスト名とポートを手動で設定します。

Visual Studio

1. Visual Studio のメインメニューから、ツールを選択し、オプションを選択します。
2. オプションメニューから、AWS Toolkit を展開し、プロキシを選択します。
3. Proxy メニューから、ホストとポートを企業のプロキシホストとポートに設定します。

   例: `http://proxy.company.com:8080`、`https://proxy.company.com:8443`

Visual Studio Code

1. VS Code から、 **CMD + ,** (Mac) または **Ctrl + ,** (Windows/Linux) を押して VS Code Settings を開きます。
2. 設定の検索バーから と入力し**Http: Proxy**、検索結果で検索します。
3. 企業プロキシ URL を入力します。

   例: `http://proxy.company.com:8080`、`https://proxy.company.com:8443`
4. (オプション) 設定検索バーで と入力し**HTTP: No Proxy**、結果で検索します。

5. 項目の追加ボタンを選択し、プロキシをバイパスするドメインをカンマで区切って追加します。

ステップ 2: SSL 証明書処理を設定する

Amazon Q は、システムにインストールされている信頼された証明書を自動的に検出して使用します。証明書エラーが発生した場合は、次の手順を実行して証明書バンドルを手動で指定する必要があります。

> **Note**
>
> 手動設定が必要な状況は次のとおりです。
>
> - プロキシの設定後に証明書関連のエラーが発生しています。
> - 企業プロキシは、システムの信頼ストアにない証明書を使用します。
> - Amazon Q は企業証明書を自動的に検出できません。

Eclipse

- の Amazon Q 設定のプロキシ設定Eclipseで、CA Cert PEM を企業証明書ファイルのパスに設定します。ファイルには、ファイル拡張子 `.pem` が必要です。(`.crt` ファイルは使用できません)。

  パスの例は次のようになります。

  ```
  /path/to/corporate-ca-bundle.pem
  ```

  このファイルを取得する手順については、「企業証明書の取得」を参照してください。

JetBrains

JetBrains では、IntelliJ IDEA ドキュメントの「信頼されたルート証明書」トピックのガイダンスに従って、企業証明書を手動でインストールします。

証明書を取得する手順については、「企業証明書の取得」を参照してください。

Visual Studio

- 次の Windows 環境変数を設定します。

- `NODE_OPTIONS = --use-openssl-ca`
- `NODE_EXTRA_CA_CERTS = cert-path`

cert-path を企業証明書ファイルのパスに置き換えます。ファイルには、ファイル拡張子 .pem が必要です。(.crt ファイルは使用できません)。

パスの例は次のようになります。

```
/path/to/corporate-ca-bundle.pem
```

企業証明書ファイルを取得する手順については、「」を参照してください企業証明書の取得。Windows 環境変数の詳細については、Node.js ドキュメントを参照してください。

Visual Studio Code

1. VS Code から、 **CMD + ,** (Mac) または **Ctrl + ,** (Windows/Linux) を押して VS Code Settings を開きます。
2. 設定検索バーから と入力し**Amazon Q › Proxy: Certificate Authority**、検索結果で検索します。
3. 企業証明書ファイルのパスを入力します。.pem または .crt ファイル拡張子が付けられます。

   パスの例は次のようになります。

   ```
   /path/to/corporate-ca-bundle.pem
   ```

   このファイルを取得する手順については、「」を参照してください企業証明書の取得。

ステップ 3: IDE を再起動する

Amazon Q を変更で更新するには、IDE を再起動する必要があります。

トラブルシューティング

前のセクションの手順を完了しても問題が解決しない場合は、次の手順を使用してトラブルシューティングを行います。

Eclipse

1. プロキシ URL 形式に `http://`または が含まれていることを確認しますhttps://。
2. 証明書ファイルのパスが正しく、アクセス可能であることを確認してください。
3. Eclipse エラーログで Amazon Q ログを確認します。エラーログに移動するには、次のいずれかを実行します。
   - で Amazon Q にサインインしEclipse、右上の Q アイコンの横にある下矢印を選択し、ヘルプ、ログの表示を選択します。
   - Eclipse メニューから、ウィンドウ、表示、エラーログを選択します。

> **Note**
>
> 次のエラーメッセージが表示された場合：
>
> - `unable to verify the first certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `self signed certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `ECONNREFUSED`、インターネット接続とプロキシ情報を確認します。

JetBrains

1. プロキシ URL 形式に `http://`または が含まれていることを確認しますhttps://。
2. 証明書ファイルのパスが正しく、アクセス可能であることを確認してください。
3. JetBrains ログファイルの Amazon Q ログを確認します。ログファイルの検索については、JetBrains IDE サポートページの「IDE ログファイルの検索IDEs 」を参照してください。

> **Note**
>
> 次のエラーメッセージが表示された場合：
>
> - `unable to verify the first certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。

- `self signed certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
- `ECONNREFUSED`、インターネット接続とプロキシ情報を確認します。

Visual Studio

1. プロキシ URL 形式に `http://`または が含まれていることを確認しますhttps://。
2. 証明書ファイルのパスが正しく、アクセス可能であることを確認してください。
3. AWS Toolkit 拡張機能のログを次のように確認します。
   - Visual Studio のメインメニューから [Extensions] を展開します。
   - Toolkit AWS を選択して AWS Toolkit メニューを展開し、View Toolkit Logs を選択します。
   - AWS Toolkit Logs フォルダがオペレーティングシステムで開いたら、ファイルを日付別にソートし、現在の問題に関連する情報を含むログファイルを見つけます。
4. Visual Studio アクティビティログで Amazon Q ログを確認します。詳細については、「アクティビティログを使用した拡張機能のトラブルシューティング」を参照してください。

> **Note**
>
> 次のエラーメッセージが表示された場合 :
>
> - `unable to verify the first certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `self signed certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `ECONNREFUSED`、インターネット接続とプロキシ情報を確認します。

Visual Studio Code

1. プロキシ URL 形式に `http://`または が含まれていることを確認しますhttps://。
2. 証明書ファイルのパスが正しく、アクセス可能であることを確認してください。
3. VS Code の出力パネルで Amazon Q ログを確認します。

> **Note**
>
> 次のエラーメッセージが表示された場合：
>
> - `unable to verify the first certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `self signed certificate`、 の手順に従って証明書を手動でステップ 2: SSL 証明書処理を設定する指定してください。
> - `ECONNREFUSED`、インターネット接続とプロキシ情報を確認します。

企業証明書の取得

企業証明書を取得するには、IT チームに次の情報を依頼してください。

- 企業証明書バンドル。通常は `.pem`または `.crt` ファイルです。
- ホスト名、ポート、認証方法など、プロキシサーバーの詳細。

または、ブラウザから証明書をエクスポートします。

1. 企業ドメインの HTTPS サイトにアクセスします。
2. アドレスバーの近くで、ロックアイコンまたは同様のアイコンを選択します。（アイコンはブラウザベンダーによって異なります）。
3. ルート証明書をファイルにエクスポートします。証明書チェーン全体を含めてください。ルート証明書をエクスポートする手順は、使用しているブラウザによって異なります。詳細な手順については、ブラウザのドキュメントを参照してください。

Amazon Q Developer とインターフェイスエンドポイント (AWS PrivateLink)

> **Note**
>
> Amazon Q Developer は、IDE で使用できる機能のインターフェイスエンドポイントをサポートしています。AWS アプリやウェブサイトでの Amazon Q とのチャットは、VPC エン

ドポイントではサポートされていません。どちらも Amazon Q Developer 変換ウェブエクスペリエンスではありません。

VPC と Amazon Q Developer とのプライベート接続を確立するには、インターフェイス VPC エンドポイントを作成します。インターフェイスエンドポイントは、インターネットゲートウェイ、NAT デバイス、VPN 接続、AWS Direct Connect 接続のいずれも必要とせずに Amazon Q API にプライベートにアクセスできるテクノロジーである AWS PrivateLink を利用しています。VPC のインスタンスは、パブリック IP アドレスがなくても Amazon Q API と通信できます。VPC と Amazon Q の間のトラフィックは、Amazon ネットワーク外に出ることはありません

各インターフェイスエンドポイントは、サブネット内の 1 つ、または複数の Elastic Network Interface によって表されます。

詳細については、「Amazon VPC ユーザーガイド」の「インターフェイス VPC エンドポイント (AWS PrivateLink)」を参照してください。

Amazon Q の VPC エンドポイントに関する考慮事項

Amazon Q 用の VPC エンドポイントを設定する前に、「Amazon VPC ユーザーガイド」の「インターフェイスエンドポイントのプロパティと制限」を確認してください。

Amazon Q は、Amazon Q と連携するように設定されたサービスのコンテキストで、VPC からのあらゆる API アクションの呼び出しをサポートしています。

前提条件

以下の手順を開始する前に、次の項目が揃っていることを確認してください。

- リソースを作成および設定するための適切なアクセス許可を持つ AWS アカウント。
- AWS アカウントに既に作成された VPC。
- AWS サービス、特に Amazon VPC と Amazon Q に精通していること。

Amazon Q 用のインターフェイス VPC エンドポイントの作成

Amazon Q サービス用の VPC エンドポイントを作成するには、Amazon VPC コンソールまたは AWS Command Line Interface (AWS CLI) を使用します。詳細については、 Amazon VPC ユーザーガイド のインターフェイスエンドポイントの作成を参照してください。

次のサービス名を使用して、Amazon Q の次の VPC エンドポイントを作成します。

- com.amazonaws.*region*.q
- com.amazonaws.us-east-1.codewhisperer

region を、Amazon Q Developer プロファイルがインストールされている AWS リージョンに置き換えます。詳細については、「Q Developer コンソールと Q Developer プロファイルでサポートされているリージョン」を参照してください。

> **Note**
>
> Amazon CodeWhisperer エンドポイント (com.amazonaws.us-east-1.codewhisperer) は、米国東部 (バージニア北部) リージョンでのみサポートされています。

エンドポイントのプライベート DNS を有効にすると、リージョンのデフォルト DNS 名 (`q.us-east-1.amazonaws.com` など) を使用して、Amazon Q への API リクエストを実行できます。

詳細については、「Amazon VPC ユーザーガイド」の「インターフェイスエンドポイントを介したサービスへのアクセス」を参照してください。

オンプレミスコンピュータを使用して Amazon Q エンドポイントに接続する

このセクションでは、オンプレミスコンピュータを使用して AWS VPC の AWS PrivateLink エンドポイントを介して Amazon Q に接続するプロセスについて説明します。

1. オンプレミスデバイスと VPC の間に VPN 接続を作成します。
2. Amazon Q 用の インターフェイス VPC エンドポイントを作成します。
3. インバウンド Amazon Route 53 エンドポイントをセットアップします。これにより、オンプレミスデバイスから Amazon Q エンドポイントの DNS 名を使用できます。

コンソール内のコーディング環境を使用して Amazon Q エンドポイントに接続する

このセクションでは、コンソール内のコーディング環境を使用して Amazon Q エンドポイントに接続するプロセスについて説明します。

このコンテキストでは、コンソール内 IDE は、 AWS コンソール内でアクセスし、IAM で に認証する IDE です。例としては、SageMaker AI Studio や AWS Glue Studio などがあります。

1. Amazon Q 用の インターフェイス VPC エンドポイントを作成します。
2. コンソール内のコーディング環境で Amazon Q をセットアップする
   - SageMaker AI Studio
   - AWS Glue スタジオ
3. Amazon Q エンドポイントを使用するようにコーディング環境を設定します。
   - SageMaker AI Studio
   - AWS Glue スタジオ

Amazon EC2 インスタンスのサードパーティー IDE AWS PrivateLink から経由で Amazon Q に接続する Amazon EC2

このセクションでは、Amazon EC2 インスタンスに Visual Studio Code や JetBrains などのサードパーティーの統合開発環境 (IDE) をインストールし、 AWS PrivateLinkを使用して Amazon Q に接続するように設定するプロセスを説明します。

1. Amazon Q 用の インターフェイス VPC エンドポイントを作成します。
2. VPC 内の任意のサブネット内で Amazon EC2 インスタンスを起動します。サードパーティーの IDE と互換性のある Amazon マシンイメージ (AMI) を選択できます。例えば、Amazon Linux 2 AMI を選択できます。
3. この Amazon EC2 インスタンスに接続します。
4. IDE (Visual Studio Code または JetBrains) をインストールして設定します。
5. Amazon Q 拡張機能またはプラグインをインストールします。
6. 経由で接続するように IDE を設定します AWS PrivateLink。
   - Visual Studio Code でのネットワーク接続
   - JetBrains リモート開発

Amazon Q Developer の使用のモニタリングと追跡

モニタリングは、Amazon Q Developer およびその他の AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。 には、Amazon Q Developer アクティビティをモニタリングおよび記録するための以下のモニタリングツールと機能 AWS が用意されています。

- AWS CloudTrail は、 によって、または に代わって行われた API コールおよび関連イベントをキャプチャ AWS アカウント し、指定した Amazon Simple Storage Service (Amazon S3) バケットにログファイルを配信します。呼び出し元のユーザーとアカウント AWS、呼び出し元の送信元 IP アドレス、呼び出しの発生日時を特定できます。詳細については、「AWS CloudTrailを使用した Amazon Q Developer API コールのログ記録」を参照してください。
- Amazon CloudWatch は、 AWS リソースと で実行されるアプリケーションを AWS リアルタイムでモニタリングします。メトリクスを収集および追跡し、カスタマイズされたダッシュボードを作成し、指定されたメトリックが指定したしきい値に達したときに通知またはアクションを実行するアラームを設定できます。例えば、CloudWatch に、Amazon Q がアカウントで呼び出された回数、または毎日のアクティブユーザーの数を追跡させることができます。詳細については、「Amazon CloudWatch による Amazon Q Developer のモニタリング」を参照してください。

Amazon Q Developer には、Amazon Q でのユーザーアクティビティの追跡と記録に役立つ以下の機能も含まれています。

- ダッシュボードには、Amazon Q Developer Pro サブスクライバーのユーザーアクティビティメトリクスの集計が表示されます。詳細については、「ダッシュボードでの Amazon Q Developer ユーザーアクティビティの表示」を参照してください。
- ユーザーアクティビティレポートには、Amazon Q の個々のユーザーが表示されます。詳細については、「」を参照してくださいAmazon Q Developer での特定のユーザーのアクティビティの表示。
- プロンプトログは、ユーザーが統合開発環境 (IDE) で Amazon Q チャットに入力するすべてのプロンプトの記録を提供します。詳細については、「Amazon Q Developer でのユーザーのプロンプトのログ記録」を参照してください。

AWS CloudTrailを使用した Amazon Q Developer API コールのログ記録

Amazon Q Developer Pro は AWS CloudTrail、Amazon Q のユーザー、ロール、または によって実行されたアクションを記録するサービスである と統合されています。CloudTrail は、Amazon Q のすべての API コール AWS のサービス をイベントとしてキャプチャします。キャプチャされた呼び出しには、Amazon Q コンソールからの呼び出しと、Amazon Q API オペレーションへのコード呼び出しが含まれます。証跡を作成すると、Amazon Q のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールの「イベント履歴」に最新のイベントが表示されます。CloudTrail で収集された情報を使用して、Amazon Q に対するリクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

CloudTrail の詳細については、「AWS CloudTrail ユーザーガイド」を参照してください。

CloudTrail での Amazon Q Developer の情報

CloudTrail は、アカウントの作成 AWS アカウント 時に で有効になります。Amazon Q Developer でアクティビティが発生すると、そのアクティビティはイベント履歴の他の AWS のサービス イベントとともに CloudTrail イベントに記録されます。 で最近のイベントを表示、検索、ダウンロードできます AWS アカウント。詳細については、「AWS CloudTrail ユーザーガイド」の「CloudTrail イベント履歴でのイベントの表示」を参照してください。

Amazon Q のイベントなど AWS アカウント、 のイベントの継続的な記録については、証跡を作成します。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョンに適用されます。証跡は、 AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをより詳細に分析し、それを基にアクションを取るために他の AWS のサービス を設定できます。詳細については、「AWS CloudTrail ユーザーガイド:」の以下のトピックを参照してください。

- 証跡作成の概要
- 「CloudTrail がサポートされているサービスと統合」
- 「CloudTrail の Amazon SNS 通知の設定」
- CloudTrail ログファイルの複数のリージョンからの受け取り
- 複数のアカウントから CloudTrail ログファイルを受け取る

Amazon Q Developer のすべてのアクションは CloudTrail により記録され、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストがルートまたは AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか
- リクエストが、ロールとフェデレーティッドユーザーのどちらの一時的なセキュリティ認証情報を使用して送信されたか
- リクエストが別の によって行われたかどうか AWS のサービス

詳細については、「AWS CloudTrail ユーザーガイド」の「CloudTrail userIdentity エレメント」を参照してください。

Amazon Q Developer ログファイルエントリの概要

「トレイル」は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは任意ソースからの単一リクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどの情報を含みます。CloudTrail ログファイルは、公開 API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

Amazon Q Developer もまた、実際にリクエストを行わずに、dryRun パラメータを使用して API コールを行い、アクションに必要なアクセス許可があることを確認します。dryRun パラメータを使用した Amazon Q Developer API への呼び出しはイベントとしてキャプチャされ、requestParameters フィールドに "dryRun" : true を指定して CloudTrail ログに記録されます。

以下の例は、SendMessage アクションを示す CloudTrail ログエントリです。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAXD12ABCDEF3G4HI5J:aws-user",
        "arn": "arn:aws:sts::123456789012:assumed-role/PowerUser/aws-user",
        "accountId": "123456789012",
        "accessKeyId": "ASIAAB12CDEFG34HIJK",
        "sessionContext": {
```

```
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAXD12ABCDEF3G4HI5J",
                "arn": "arn:aws:iam::123456789012:role/PowerUser",
                "accountId": "123456789012",
                "userName": "PowerUser"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-11-28T10:00:00Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-11-28T10:00:00Z",
    "eventSource": "q.amazonaws.com",
    "eventName": "SendMessage",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "123.456.789.012",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101
 Firefox/115.0",
    "requestParameters": {
        "Origin": "https://conversational-experience-
worker.widget.console.aws.amazon.com",
        "conversationId": "a298ec0d-0a49-4d2e-92bd-7d6e629b4619",
        "source": "CONSOLE",
        "conversationToken": "***",
        "utterance": "***"
    },
    "responseElements": {
        "result": {
            "content": {
                "text": {
                    "body": "***",
                    "references": []
                }
            },
            "format": "PLAINTEXT",
            "intents": {},
            "type": "TEXT"
        },
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "metadata": {
```

```
            "conversationExpirationTime": "2024-02-25T19:31:38Z",
            "conversationId": "a298ec0d-0a49-4d2e-92bd-7d6e629b4619",
            "conversationToken": "***",
            "utteranceId": "3b87b46f-04a9-41ef-b8fe-8abf52d2c053"
        },
        "resultCode": "LLM"
    },
    "additionalEventData": {
        "quickAction": "dev"
    },
    "requestID": "19b3c30e-906e-4b7f-b5c3-509f67248655",
    "eventID": "a552c487-7d97-403a-8ec4-d49539c7a03d",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

以下の例は、PassRequest アクションを示す CloudTrail ログエントリです。

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDA6ON6E4XEGIEXAMPLE",
        "arn": "arn:aws:iam::555555555555:user/Mary",
        "accountId": "555555555555",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
            "type": "Role",
            "principalId": "AIDA6ON6E4XEGIEXAMPLE",
            "arn": "arn:aws:iam::555555555555:user/Mary",
            "accountId": "555555555555",
            "userName": "Mary"

        },
        "attributes": {
            "creationDate": "2024-04-10T20:03:01Z",
            "mfaAuthenticated": "false"
        },
        "invokedBy": "q.amazonaws.com"
```

```
    },
    "eventTime": "2024-04-10T20:04:42Z",
    "eventSource": "q.amazonaws.com",
    "eventName": "PassRequest",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "q.amazonaws.com",
    "userAgent": "q.amazonaws.com",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "2d528c76-329e-410b-9516-EXAMPLE565dc",
    "eventID": "ba0801a1-87ec-4d26-be87-EXAMPLE75bbb",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "555555555555",
    "eventCategory": "Management"
}
```

次の例は、Amazon Q がユーザーに代わって s3:ListBuckets アクションを呼び出す CloudTrail ログエントリを示しています。

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDA6ON6E4XEGIEXAMPLE",
        "arn": "arn:aws:iam::555555555555:user/Paulo",
        "accountId": "555555555555",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDA6ON6E4XEGIEXAMPLE",
                "arn": "arn:aws:iam::555555555555:user/Paulo",
                "accountId": "555555555555",
                "userName": "Paulo"
            },
            "attributes": {
                "creationDate": "2024-04-10T14:06:08Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "q.amazonaws.com"
```

```
    },
    "eventTime": "2024-04-10T14:07:55Z",
    "eventSource": "s3.amazonaws.com",
    "eventName": "ListBuckets",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "q.amazonaws.com",
    "userAgent": "q.amazonaws.com",
    "requestParameters": {
        "Host": "s3.amazonaws.com"
    },
    "responseElements": null,
    "additionalEventData": {
        "SignatureVersion": "SigV4",
        "CipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
        "bytesTransferredIn": 0,
        "AuthenticationMethod": "AuthHeader",
        "x-amz-id-2": "ExampleRequestId123456789",
        "bytesTransferredOut": 4054
    },
    "requestID": "ecd94349-b36f-44bf-b6f5-EXAMPLE9c463",
    "eventID": "2939ba50-1d26-4a5a-83bd-EXAMPLE85850",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "555555555555",
    "vpcEndpointId": "vpce-EXAMPLE1234",
    "eventCategory": "Management"
}
```


Amazon CloudWatch による Amazon Q Developer のモニタリング

> **Note**
>
> ここで説明するメトリクスは、IDE での Amazon Q の使用にのみ関連しています。

CloudWatch を使用して Amazon Q Developer をモニタリングできます。CloudWatch は raw データを収集し、それを読み取り可能なほぼリアルタイムのメトリクスに処理します。こうした統計は 15 か月間保持されるため、履歴情報にアクセスして Amazon Q のパフォーマンスをより的確に把握できます。また、特定のしきい値を監視するアラームを設定し、これらのしきい値に達したときに通知

を送信したりアクションを実行したりできます。詳細については、「Amazon CloudWatch ユーザーガイド」を参照してください。

Amazon Q Developer サービスは、AWS/Q 名前空間で以下のメトリクスをレポートします。

ディメンション	メトリクス	ユースケースまたは説明
カウント	Invocations	時間の経過とともにカウントされた呼び出しの数を把握できます。
UserCount	DailyActiveUserTrend	1 日あたりのアクティブユーザー数を把握できます。
SubscriptionUserCount	SubscriptionCount	有料サブスクリプションを持つユーザーの数を把握できます。
UniqueUserCount	MonthlyActiveUniqueUsers	特定の月にアクティブになっているユーザーの数を把握できます。
ProgrammingLanguage、SuggestionState、CompletionType	GeneratedLineCount	Amazon Q Developer によって生成された行数を決定します。
ProgrammingLanguage、SuggestionState、CompletionType	SuggestionReferenceCount	リファレンスを含む提案トリガーの数が発生した回数を把握できます。
ProgrammingLanguage	CodeScanCount	実行されたコードスキャンの数を把握できます。

ディメンション	メトリクス	ユースケースまたは説明
ProgrammingLanguage	TotalCharacterCount	Amazon Q Developer からのすべての提案を含む、ファイル内の文字数。
ProgrammingLanguage	CodeWhispererCharacterCount	Amazon Q Developer によって生成された文字数。

呼び出しを集計するには、Sum 統計を使用します。

DailyActiveUserTrend を集計するには、Sum 統計を使用し、「1 Day」を期間として使用します。

SubscriptionCount を集計するには、Sum 統計を使用します。

MonthlyActiveUniqueUsers を集計するには、Sum 統計を使用し、期間として「30 Days」を使用します。

Amazon CloudWatch Logs で特定のユーザーによるアクションを特定する

Amazon Q Developer の使用についてユーザーレベルのメトリクスを取得できます。特定のアクションを実行したユーザーを確認するには、SendTelemetryEvent というイベントを探し、含まれている SendTelemetryEventRequest タイプの JSON オブジェクトを調べます。そのオブジェクト内で、スキーマは次のように表示されます。

> **Tip**
>
> Amazon Q Developer の個々のユーザーのアクティビティを CSV 形式でレポートに出力することもできます。詳細については、「Amazon Q Developer での特定のユーザーのアクティビティの表示」を参照してください。

```
http://json-schema.org/draft-07/schema#",
    "definitions": {
        "ProgrammingLanguage": {
            "type": "object",
            "properties": {
```

```
                "languageName": {
                    "type": "string",
                    "enum": [
                        "python",
                        "javascript",
                        "java",
                        "csharp",
                        "typescript",
                        "c",
                        "cpp",
                        "go",
                        "kotlin",
                        "php",
                        "ruby",
                        "rust",
                        "scala",
                        "shell",
                        "sql",
                        "json",
                        "yaml",
                        "vue",
                        "tf",
                        "tsx",
                        "jsx",
                        "plaintext"
                    ],
                    "description": "Programming Languages supported by Q"
                }
            }
        },
        "Dimension": {
            "type": "object",
            "properties": {
                "name": {
                    "type": "string",
                    "description": "must match ^[-a-zA-Z0-9._]*$ and be between 1 and
255 characters"
                },
                "value": {
                    "type": "string",
                    "description": "must match ^[-a-zA-Z0-9._]*$ and be between 1 and
1024 characters"
                }
            }
```

```
        }
    },
    "telemetryEvents": {
        "UserTriggerDecisionEvent": {
            "type": "object",
            "properties": {
                "sessionId": {
                    "type": "string",
                    "description": "UUID for the session"
                },
                "requestId": {
                    "type": "string",
                    "description": "UUID for the request"
                },
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "completionType": {
                    "type": "string",
                    "enum": [
                        "BLOCK",
                        "LINE"
                    ]
                },
                "suggestionState": {
                    "type": "string",
                    "enum": [
                        "ACCEPT",
                        "REJECT",
                        "DISCARD",
                        "EMPTY"
                    ]
                },
                "recommendationLatencyMilliseconds": {
                    "type": "number"
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
```

```
                },
                "triggerToResponseLatencyMilliseconds": {
                    "type": "number"
                },
                "suggestionReferenceCount": {
                    "type": "integer"
                },
                "generatedLine": {
                    "type": "integer"
                },
                "numberOfRecommendations": {
                    "type": "integer"
                }
            },
            "required": [
                "sessionId",
                "requestId",
                "programmingLanguage",
                "completionType",
                "suggestionState",
                "recommendationLatencyMilliseconds",
                "timestamp"
            ]
        },
        "CodeCoverageEvent": {
            "type": "object",
            "properties": {
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "acceptedCharacterCount": {
                    "type": "integer"
                },
                "totalCharacterCount": {
                    "type": "integer"
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
```

```
                },
                "unmodifiedAcceptedCharacterCount": {
                    "type": "integer"
                }
            },
            "required": [
                "programmingLanguage",
                "acceptedCharacterCount",
                "totalCharacterCount",
                "timestamp"
            ]
        },
        "UserModificationEvent": {
            "type": "object",
            "properties": {
                "sessionId": {
                    "type": "string",
                    "description": "UUID for the session"
                },
                "requestId": {
                    "type": "string",
                    "description": "UUID for the request"
                },
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "modificationPercentage": {
                    "type": "number",
                    "description": "This is the percentage of AI generated code which
 has been modified by the user"
                },
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
                }
            },
            "required": [
                "sessionId",
                "requestId",
```

```
                "programmingLanguage",
                "modificationPercentage",
                "timestamp"
            ]
        },
        "CodeScanEvent": {
            "type": "object",
            "properties": {
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "codeScanJobId": {
                    "type": "string"
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
                },
                "codeAnalysisScope": {
                    "type": "string",
                    "enum": [
                        "FILE",
                        "PROJECT"
                    ]
                }
            },
            "required": [
                "programmingLanguage",
                "codeScanJobId",
                "timestamp"
            ]
        },
        "CodeScanRemediationsEvent": {
            "type": "object",
            "properties": {
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "CodeScanRemediationsEventType": {
                    "type": "string",
                    "enum": [
                        "CODESCAN_ISSUE_HOVER",
                        "CODESCAN_ISSUE_APPLY_FIX",
                        "CODESCAN_ISSUE_VIEW_DETAILS"
```

```
                    ]
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
                },
                "detectorId": {
                    "type": "string"
                },
                "findingId": {
                    "type": "string"
                },
                "ruleId": {
                    "type": "string"
                },
                "component": {
                    "type": "string"
                },
                "reason": {
                    "type": "string"
                },
                "result": {
                    "type": "string"
                },
                "includesFix": {
                    "type": "boolean"
                }
            }
        },
        "MetricData": {
            "type": "object",
            "properties": {
                "metricName": {
                    "type": "string",
                    "description": "must match pattern ^[-a-zA-Z0-9._]*$ and be between
1 and 1024 characters"
                },
                "metricValue": {
                    "type": "number"
                },
                "timestamp": {
                    "type": "string",
                    "description": "datetime, example: Jul 23, 2024, 12:11:02 AM"
                },
```

```
                "product": {
                    "type": "string",
                    "description": "must match pattern ^[-a-zA-Z0-9._]*$ and be between
 1 and 128 characters"
                },
                "dimensions": {
                    "type": "array",
                    "items": {
                        "$ref": "#/definitions/Dimension"
                    },
                    "description": "maximum size of 30"
                }
            },
            "required": [
                "metricName",
                "metricValue",
                "timestamp",
                "product"
            ]
        },
        "ChatAddMessageEvent": {
            "type": "object",
            "properties": {
                "conversationId": {
                    "type": "string",
                    "description": "ID which represents a multi-turn conversation,
 length between 1 and 128"
                },
                "messageId": {
                    "type": "string",
                    "description": "Unique identifier for the chat message"
                },
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "userIntent": {
                    "type": "string",
                    "enum": [
                        "SUGGEST_ALTERNATE_IMPLEMENTATION",
                        "APPLY_COMMON_BEST_PRACTICES",
                        "IMPROVE_CODE",
                        "SHOW_EXAMPLES",
```

```
                    "CITE_SOURCES",
                    "EXPLAIN_LINE_BY_LINE",
                    "EXPLAIN_CODE_SELECTION",
                    "GENERATE_CLOUDFORMATION_TEMPLATE"
                ]
            },
            "hasCodeSnippet": {
                "type": "boolean"
            },
            "programmingLanguage": {
                "$ref": "#/definitions/ProgrammingLanguage"
            },
            "activeEditorTotalCharacters": {
                "type": "integer"
            },
            "timeToFirstChunkMilliseconds": {
                "type": "number"
            },
            "timeBetweenChunks": {
                "type": "array",
                "items": {
                    "type": "number"
                },
                "description": "maximum size of 100"
            },
            "fullResponselatency": {
                "type": "number"
            },
            "requestLength": {
                "type": "integer"
            },
            "responseLength": {
                "type": "integer"
            },
            "numberOfCodeBlocks": {
                "type": "integer"
            },
            "hasProjectLevelContext": {
                "type": "boolean"
            }
        },
        "required": [
            "conversationId",
            "messageId"
```

```
            ]
        },
        "ChatInteractWithMessageEvent": {
            "type": "object",
            "properties": {
                "conversationId": {
                    "type": "string",
                    "description": "ID which represents a multi-turn conversation,
 length between 1 and 128"
                },
                "messageId": {
                    "type": "string",
                    "description": "Unique identifier for the chat message"
                },
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "interactionType": {
                    "type": "string",
                    "enum": [
                        "INSERT_AT_CURSOR",
                        "COPY_SNIPPET",
                        "COPY",
                        "CLICK_LINK",
                        "CLICK_BODY_LINK",
                        "CLICK_FOLLOW_UP",
                        "HOVER_REFERENCE",
                        "UPVOTE",
                        "DOWNVOTE"
                    ],
                    "description": "Chat Message Interaction Type"
                },
                "interactionTarget": {
                    "type": "string",
                    "description": "Target of message interaction"
                },
                "acceptedCharacterCount": {
                    "type": "integer"
                },
                "acceptedLineCount": {
                    "type": "integer"
                },
```

```
                "acceptedSnippetHasReference": {
                    "type": "boolean"
                },
                "hasProjectLevelContext": {
                    "type": "boolean"
                }
            },
            "required": [
                "conversationId",
                "messageId"
            ]
        },
        "ChatUserModificationEvent": {
            "type": "object",
            "properties": {
                "conversationId": {
                    "type": "string",
                    "description": "ID which represents a multi-turn conversation,
 length between 1 and 128"
                },
                "customizationArn": {
                    "type": "string",
                    "description": "ARN of the customization matching pattern: ^arn:
[-.a-z0-9]{1,63}:codewhisperer:([-.a-z0-9]{0,63}:){2}([a-zA-Z0-9-_:/]){1,1023}$"
                },
                "messageId": {
                    "type": "string",
                    "description": "Unique identifier for the chat message"
                },
                "programmingLanguage": {
                    "$ref": "#/definitions/ProgrammingLanguage"
                },
                "modificationPercentage": {
                    "type": "number",
                    "description": "This is the percentage of AI generated code which
 has been modified by the user"
                },
                "hasProjectLevelContext": {
                    "type": "boolean"
                }
            },
            "required": [
                "conversationId",
                "messageId",
```

```
                "modificationPercentage"
            ]
        },
        "SuggestionState": {
            "type": "string",
            "enum": [
                "ACCEPT",
                "REJECT",
                "DISCARD",
                "EMPTY"
            ]
        },
        "TerminalUserInteractionEvent": {
            "type": "object",
            "properties": {
                "terminalUserInteractionEventType": {
                    "type": "string",
                    "enum": [
                        "CODEWHISPERER_TERMINAL_TRANSLATION_ACTION",
                        "CODEWHISPERER_TERMINAL_COMPLETION_INSERTED"
                    ],
                    "description": "Terminal User Interaction Event Type"
                },
                "terminal": {
                    "type": "string"
                },
                "terminalVersion": {
                    "type": "string"
                },
                "shell": {
                    "type": "string"
                },
                "shellVersion": {
                    "type": "string"
                },
                "duration": {
                    "type": "integer"
                },
                "timeToSuggestion": {
                    "type": "integer"
                },
                "isCompletionAccepted": {
                    "type": "boolean"
                },
```

```
                "cliToolCommand": {
                    "type": "string"
                }
            }
        },
        "FeatureDevEvent": {
            "type": "object",
            "properties": {
                "conversationId": {
                    "type": "string",
                    "description": "ID which represents a multi-turn conversation,
length between 1 and 128"
                }
            },
            "required": [
                "conversationId"
            ]
        }
    },
    "SendTelemetryEventRequest": {
        "type": "object",
        "properties": {
            "clientToken": {
                "type": "string",
                "description": "The client's authentication token"
            },
            "telemetryEvent": {
                "properties": {
                    "oneOf": [
                        {
                            "_comment": "This event is emitted when a user accepts or
rejects an inline code suggestion",
                            "$ref": "#/definitions/userTriggerDecisionEvent"
                        },
                        {
                            "_comment": "This event is emitted every five minutes. It
details how much code is written by inline code suggestion and in total during that
period",
                            "$ref": "#/definitions/codeCoverageEvent"
                        },
                        {
                            "_comment": "This event is emitted when a code snippet from
inline code suggestion has been edited by a user. It details the percentage of that
code snippet modified by the user",
```

```
                            "$ref": "#/definitions/userModificationEvent"
                        },
                        {
                            "_comment": "This field is emitted when a security scan is
requested by a user",
                            "$ref": "#/definitions/codeScanEvent"
                        },
                        {
                            "_comment": "This field is emitted when a security scan
recommended remediation is accepted by a user",
                            "$ref": "#/definitions/codeScanRemediationsEvent"
                        },
                        {
                            "_comment": "This event is deprecated but may still occur
in telemetry. Do not use this.",
                            "$ref": "#/definitions/metricData"
                        },
                        {
                            "_comment": "This event is emitted when Q adds an AI
generated message to the chat window",
                            "$ref": "#/definitions/chatAddMessageEvent"
                        },
                        {
                            "_comment": "This event is emitted when a user interacts
with a chat message",
                            "$ref": "#/definitions/chatInteractWithMessageEvent"
                        },
                        {
                            "_comment": "This event is emitted when a user modifies a
code snippet sourced from chat. It gives a percentage of the code snippet which has
been modified",
                            "$ref": "#/definitions/chatUserModificationEvent"
                        },
                        {
                            "_comment": "This event is emitted when a user interacts
with a terminal suggestion",
                            "$ref": "#/definitions/terminalUserInteractionEvent"
                        },
                        {
                            "_comment": "This event is emitted when a user first
prompts the /dev feature.",
                            "$ref": "#/definitions/featureDevEvent"
                        }
                    ]
```

```
                }
            },
            "optOutPreference": {
                "type": "string",
                "enum": [
                    "OPTIN",
                    "OPTOUT"
                ],
                "description": "OPTOUT and telemetry is only provided to the account of
purchasing enterprise, OPTIN and telemetry may also be used for product improvement"
            },
            "userContext": {
                "type": "object",
                "properties": {
                    "ideCategory": {
                        "type": "string",
                        "enum": [
                            "JETBRAINS",
                            "VSCODE",
                            "CLI",
                            "JUPYTER_MD",
                            "JUPYTER_SM"
                        ]
                    },
                    "operatingSystem": {
                        "type": "string",
                        "description": "The operating system being used"
                    },
                    "product": {
                        "type": "string",
                        "description": "The name of the product being used"
                    },
                    "clientId": {
                        "type": "string",
                        "description": "A UUID representing the individual client being
used"
                    },
                    "ideVersion": {
                        "type": "string",
                        "description": "The version of the Q plugin"
                    }
                },
                "required": [
                    "ideCategory",
```

```
                    "operatingSystem",
                    "product",
                    "clientId",
                    "ideVersion"
                ]
            },
            "profileArn": {
                "type": "string",
                "description": "The arn of the Q Profile used to configure individual
user accounts."
```

SendTelemetryEvent に複数のテレメトリイベントのいずれかが含まれている場合があることを確認します。こうしたイベントはそれぞれ、開発環境間の特定の相互作用について説明しています。

以下では、各イベントをより詳細に説明しています。

UserTriggerDecisionEvent

このイベントは、ユーザーが Amazon Q によって行われた提案とやり取りしたときにトリガーされます。提案が受け入れられたか、拒否されたか、変更されたか、関連するメタデータとともにキャプチャされます。

- `completionType`: 完了がブロックか行か。
- `suggestionState`: ユーザーが提案を受け入れたか、拒否したか、破棄したか。

CodeScanEvent

このイベントは、コードスキャンの実行時に記録されます。スキャンの範囲と結果を追跡し、セキュリティとコード品質チェックに関するインサイトを提供します。

- `codeScanJobId`: コードスキャンジョブの一意の識別子。
- `codeAnalysisScope`: スキャンがファイルレベルまたはプロジェクトレベルのどちらで実行されたか。
- `programmingLanguage`: スキャンされる言語。

CodeScanRemediationsEvent

このイベントは、修正の適用や問題の詳細の表示など、ユーザーが Amazon Q の修復提案とどのようにやり取りしたかをキャプチャします。

- CodeScanRemediationsEventType: 実行された修復アクションのタイプ (詳細の表示や修正の適用など)。
- includesFix: コードの問題に推奨される修正が含まれているかどうかを示すブール値。

ChatAddMessageEvent

このイベントは、進行中のチャットの会話に新しいメッセージが追加されるとトリガーされます。ユーザーのインテントと、関連するコードス二ペットをキャプチャします。

- conversationId: 会話の一意の識別子。
- messageId: チャットメッセージの一意の識別子。
- userIntent: コードの改善やコードの説明など、ユーザーのインテント。
- programmingLanguage: チャットメッセージに関連する言語。

ChatInteractWithMessageEvent

このイベントは、ユーザーがコードス二ペットのコピー、リンクのクリック、リファレンスへのカーソルを合わせるなど、チャットメッセージを操作する際に発生します。

- interactionType: 操作のタイプ (コピー、ホバー、クリックなど)。
- interactionTarget: 操作のターゲット (コードス二ペットやリンクなど)。
- acceptedCharacterCount: 受け入れられたメッセージの文字数。
- acceptedSnippetHasReference: 受け入れられたス二ペットにリファレンスが含まれているかどうかを示すブール値。

TerminalUserInteractionEvent

このイベントは、ターミナルコマンドとのユーザーインタラクションやターミナル環境での完了を記録します。

- terminalUserInteractionEventType: インタラクションのタイプ (ターミナル翻訳やコード完了など)。
- isCompletionAccepted: 完了がユーザーによって受け入れられたかどうかを示すブール値。
- duration: インタラクションにかかった時間。

Amazon CloudWatch Logs でのカスタマイズ関連のメッセージへのアクセス

カスタマイズの作成に関する情報は、Amazon CloudWatch Logs に保存できます。

次のアクセス許可セットを使用して、Amazon Q Developer 管理者にこれらの口グの表示を許可できます。

複数のリソースにログを配信するために必要なアクセス許可の詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「追加のアクセス許可が必要な口ギング [V2]」を参照してください。

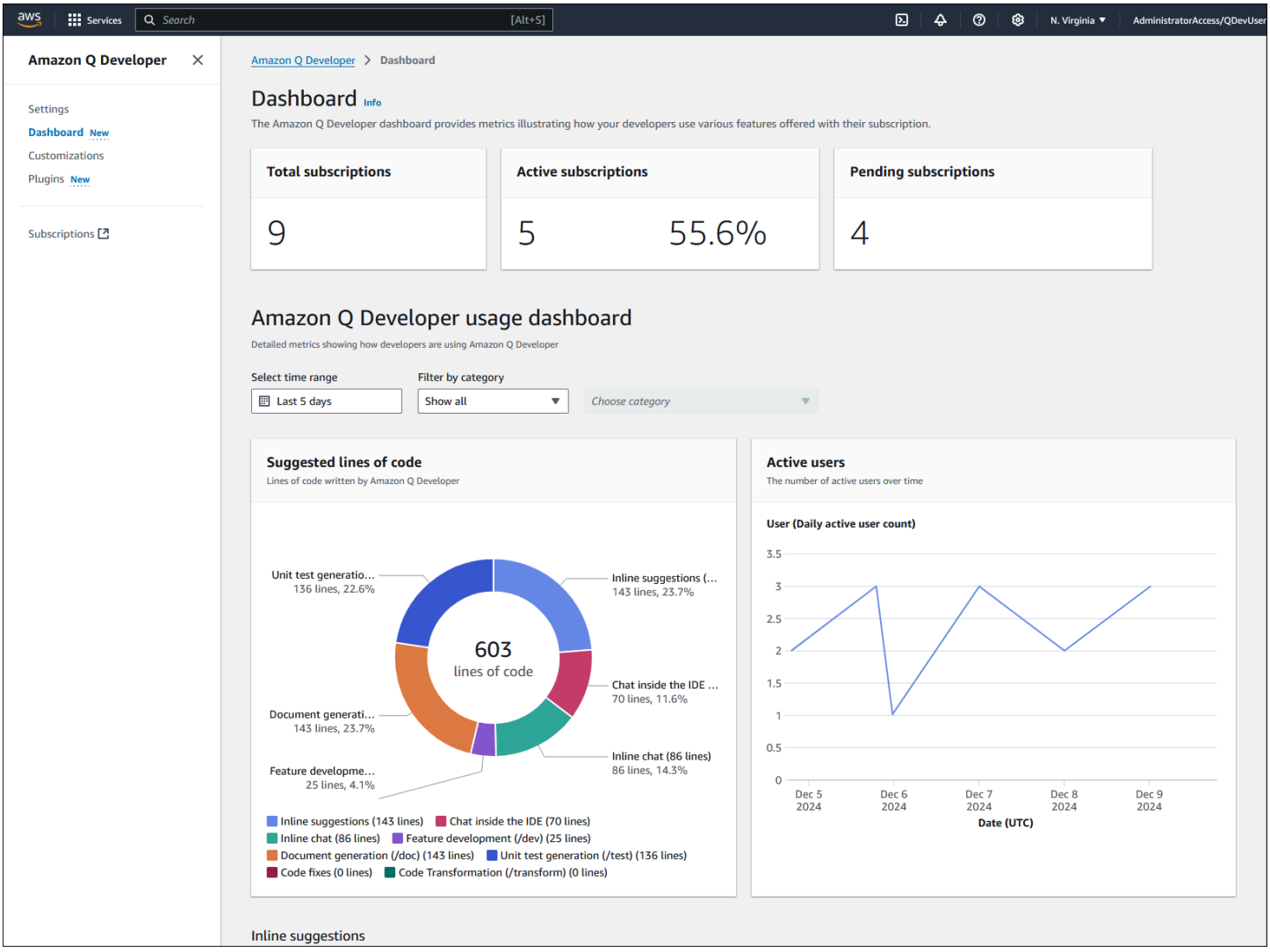
JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
                "logs:PutDeliverySource",
                "logs:GetDeliverySource",
                "logs:DeleteDeliverySource",
                "logs:DescribeDeliverySources",
                "logs:PutDeliveryDestination",
                "logs:GetDeliveryDestination",
                "logs:DeleteDeliveryDestination",
                "logs:DescribeDeliveryDestinations",
                "logs:CreateDelivery",
                "logs:GetDelivery",
                "logs:DeleteDelivery",
                "logs:DescribeDeliveries",
                "firehose:ListDeliveryStreams",
                "firehose:DescribeDeliveryStream",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:logs:us-east-1:111122223333:log-group:*",
```

```
                "arn:aws:firehose:us-east-1:111122223333:deliverystream/*",
                "arn:aws:s3:::*"
            ]
        }
    ]
}
```

ダッシュボードでの Amazon Q Developer ユーザーアクティビティの表示

Amazon Q Developer 管理者のみが使用できる Amazon Q Developer ダッシュボードには、Pro 階層のサブスクライバーがサービスをどのように使用するかについての有用なデータがまとめられています。



Amazon Q は、ほとんどの場合、時間単位で新しいメトリクスを生成して表示します。1 時間ごとに更新されない唯一のセクションはアクティブユーザーウィジェットです。アクティブユーザーウィジェットは、協定世界時 (UTC) クロックに従って毎日更新されます。

ダッシュボードには、サブスクライブしているユーザーから収集されたメトリクスが表示されます。

- 現在サインインしている AWS アカウント

  and

- メンバーアカウント。サブスクリプションの組織全体の可視性が有効になっている管理アカウントにサインインしている場合。

> **Note**
>
> アクティブユーザーウィジェットには、現在サインインしているアカウントの情報のみが表示されます。

ダッシュボードを表示およびフィルタリングするには

1. AWS Management Consoleにサインインします。
2. Amazon Q Developer コンソールに切り替えます。
3. ナビゲーションペインから、[ダッシュボード] を選択します。
4. （オプション) 日付範囲、プログラミング言語、カスタマイズ、または統合開発環境 (IDE) ベンダーで情報をフィルタリングします。

注意:

- ダッシュボードリンクがナビゲーションペインで使用できない場合は、「」を参照してくださいダッシュボードのトラブルシューティング。
- Amazon Q Developer の使用状況をユーザーごとに内訳した日次レポートにユーザーメトリクスを送信する場合は、「」を参照してくださいAmazon Q Developer での特定のユーザーのアクティビティの表示。
- 特定のメトリクスの詳細については、「」を参照Amazon Q Developer ダッシュボード使用状況メトリクスの説明するか、ダッシュボードページの右上にあるヘルプリンク
  ( )
  を選択します。

Amazon Q Developer ダッシュボード使用状況メトリクスの説明

次の表は、Amazon Q Developer ダッシュボードに表示されるメトリクスを示しています。

ダッシュボードのページの詳細については、「ダッシュボードでの Amazon Q Developer ユーザーアクティビティの表示」を参照してください。

メトリクス名	説明
サブスクリプションの合計	サブスクリプションの組織全体の可視性が有効になっている管理 AWS アカウントにサインインしている場合、現在のアカウントのサブスクリプションの合計とメンバーアカウントのサブスクリプションを表示します。 https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/subscribe-visibility.html
アクティブなサブスクリプション	サブスクリプションの組織全体の可視性が有効になっている管理アカウントにサインインしている場合、現在の AWS アカウントのアクティブなサブスクリプションとメンバーアカウントのサブスクリプションの合計を表示します。 https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/subscribe-visibility.html アクティブなサブスクリプションは、統合開発環境 (IDE) で Amazon Q の使用を開始したユーザーに属するサブスクリプションです。これらのサブスクリプションに対して課金されます。アクティブなサブスクリプションの詳細については、「」を参照してくださいAmazon Q Developer サブスクリプションのステータス。
保留中のサブスクリプション	サブスクリプションの組織全体の可視性が有効になっている管理アカウントにサインインしている場合、現在の AWS アカウントの保留中のサブスクリプションとメンバーアカウントのサブスクリプションの合計を表示します。

メトリクス名	説明
	https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/subscribe-visibility.html 保留中のサブスクリプションは、IDE で Amazon Q の使用をまだ開始していないユーザーに属するサブスクリプションです。これらのサブスクリプションには課金されません。保留中のサブスクリプションの詳細については、「」を参照してくださいAmazon Q Developer サブスクリプションのステータス。
使用できるコード行	Amazon Q 機能 (**/dev**、、 **/doc** **/test**など) 別に分類された、ユーザーが受け入れたコード行を示す円グラフを表示します。
Active users	指定した日付範囲に IDE で Amazon Q をアクティブに使用しているサブスクライバーの数を示す折れ線グラフを表示します。
インライン提案	インライン提案機能の提案と承諾された提案の合計数を表示します。承諾された提案の割合は、ユーザーが承諾した提案の数を Amazon Q によって生成された提案の合計で割って計算されます。合計提案数には、承諾された提案とアクティブに拒否された提案が含まれます。ユーザーが IDE で他のオペレーションを入力または開始し続けたために却下された提案は含まれません。
インラインチャット	インラインチャット機能の提案の合計数と受け入れられた提案の数を表示します。承諾された提案の割合は、ユーザーが承諾した提案の数を Amazon Q によって生成された提案の合計で割って計算されます。

メトリクス名	説明
IDE でのチャット – 送信されたメッセージの合計	ユーザーの IDE の Amazon Q チャットウィンドウに Amazon Q からのレスポンスの合計数を表示します。
機能開発 (/dev) – 承認率	機能開発 (/dev) 機能の承諾率を表示します。承諾率は、ユーザーが承諾したコード行数を Amazon Q が提案したコード行の合計で割って計算されます。
ドキュメント生成 (/doc)	ドキュメント生成 (/doc) 機能によって作成および更新されたドキュメントファイル (READMEsやサポートファイルなど) の合計数を表示します。承諾率は、ユーザーが承諾したファイルの更新または作成の数を、Amazon Q が提案したファイルの更新または作成の合計数で割ったものです。
ユニットテスト生成 (/test)	ユニットテスト生成 (/テスト) 機能によって生成されたユニットテストの合計数と、ユーザーが受け入れたユニットテストの数を表示します。承認率は、ユーザーが受け入れたユニットテストの数を Amazon Q によって生成されたユニットテストの合計数で割って計算されます。
コードレビュー (/レビュー)	コードレビュー (/レビュー) 機能によって生成されたコードレビューと検出結果レポートの総数を表示します。合計コードレビュー (手動のみ) と検出結果レポート (手動のみ) は、自動生成されていないコードレビューと検出結果レポートを指します。

メトリクス名	説明
コードの修正	Amazon Q によって生成されたコード修正の総数を表示します。承認率は、ユーザーが受け入れたコード修正の数を Amazon Q によって提案されたコード修正の総数で割って計算されます。
コード変換 (/transform)	変換 (/変換) 機能によって実行されたコード変換の合計数と、処理されたコード行数を表示します。

Amazon Q Developer ダッシュボードの無効化

データプライバシー、ページのロード時間、またはその他の潜在的な問題について懸念がある場合は、Amazon Q Developer ダッシュボードを無効にすることができます。ダッシュボードを無効にすると、ダッシュボードページ (およびそのリンク) は Amazon Q Developer コンソールで使用できなくなります。

ダッシュボードのページの詳細については、「使用状況メトリクスの表示 (ダッシュボード)」を参照してください。

ダッシュボードを無効にするには

1. Amazon Q Developer コンソールを開きます。
   - の組織インスタンスで Amazon Q Developer をセットアップする場合は AWS IAM Identity Center、管理アカウントまたはメンバーアカウントを使用してサインインします。
   - IAM Identity Center のアカウントインスタンスで Amazon Q Developer をセットアップする場合は、そのインスタンスに関連付けられたアカウントを使用してサインインします。
2. 「設定」を選択し、「Amazon Q Developer ユーザーアクティビティ」セクションで「編集」を選択します。
3. Amazon Q Developer 使用状況ダッシュボードを無効にします。

Amazon Q Developer ダッシュボードのトラブルシューティング

Amazon Q Developer ダッシュボードページが利用できない場合は、次の操作を行います。

- アクセス許可を確認します。ダッシュボードを表示するには、次のアクセス許可が必要です。
  - `q:ListDashboardMetrics`
  - `codewhisperer:ListProfiles`
  - `sso:ListInstances`
  - `user-subscriptions:ListUserSubscriptions`
  - 2024 年 11 月 22 日より前に生成されたメトリクスを表示するには、以下`cloudwatch:GetMetricData`と も必要です。 `cloudwatch:ListMetrics`

    権限の詳細については、「管理者に Amazon Q Developer コンソールの使用を許可する」を参照してください。
- 設定を確認します。Amazon Q Developer コンソールで、設定を選択し、Amazon Q Developer 使用状況ダッシュボードの切り替えが有効になっていることを確認します。

ダッシュボードのページの詳細については、「使用状況メトリクスの表示 (ダッシュボード)」を参照してください。

Amazon Q Developer での特定のユーザーのアクティビティの表示

組織内の個々の Amazon Q Developer サブスクライバーのユーザーアクティビティテレメトリを収集し、その情報をレポートに表示するように Amazon Q を設定できます。このレポートでは、特定のユーザーが Amazon Q をどのように使用しているかを把握できます。

Amazon Q は毎日午前 0 時 (00:00) 協定世界時 (UTC) にレポートを生成し、次のパスの CSV ファイルに保存します。

`s3://`*`bucketName`*`/`*`prefix`*`/AWSLogs/`*`accountId`*`/QDeveloperLogs/by_user_analytic/`*`region`*`/`*`year`*`/`*`month`*`/`*`day`*`/00/`*`accountId`*`_by_user_analytic_`*`timestamp`*`.c`

CSV ファイルは次のようにレイアウトされます。

- 各行には、その日に Amazon Q とやり取りしたユーザーが表示されます。
- 各列には、「」で説明されているように、メトリクスが表示されますユーザーアクティビティレポートのメトリクス。メトリクスは、その日に収集されたユーザーテレメトリに基づいて計算されます。

1 日に 1,000 人を超えるユーザーが Amazon Q とやり取りする場合、Amazon Q はデータを、それぞれ 1,000 人のユーザーを含む複数の CSV ファイルに分割し、、 `part_1` `part_2`などのサフィックスがあります。

> **Note**
>
> ユーザーアクティビティレポートを有効にすると、開発者が Amazon Q を有効にして使用状況データを IDE の 設定に送信する方法に関係なく、Amazon Q AWS はテレメトリを収集します。この設定は、組織ではなくAWS 企業がテレメトリを使用できるかどうかを制御します。この設定の詳細については「クライアント側のテレメトリの共有からオプトアウトする」を参照してください。

次の手順を使用して、ユーザーアクティビティレポートを有効にします。

前提条件

ユーザーアクティビティレポートの CSV ファイルを保持する Amazon S3 バケットを作成します。バケットは、次の条件を満たす必要があります。

- Amazon Q Developer プロファイルがインストールされている AWS リージョンにいる。このプロファイルは、IAM Identity Center ワークフォースユーザーを Amazon Q Developer Pro に初めてサブスクライブしたときにインストールされました。このプロファイルとサポートされているリージョンの詳細については、「」および「」を参照してくださいAmazon Q Developer プロファイルとはQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。
- ユーザーがサブスクライブされている AWS アカウントにいる。ユーザーが複数の AWS アカウントにサブスクライブしている場合は、それらの各アカウントにバケットを作成する必要があります。クロスアカウントバケットはサポートされていません。
- （オプションですが推奨) プロンプトのログ記録に使用するバケットとは異なります。
- Amazon Q が CSV ファイルを保存するサブフォルダとも呼ばれるプレフィックスを含めます。CSV ファイルをバケットのルートに保存することはできません。
- 次のようなバケットポリシーが適用されていること。*`bucketName`*、*`region`*、*`accountId`*、*`prefix`* はお客様の情報で置き換えてください。

  JSON

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

```
        {
            "Sid": "QDeveloperLogsWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "q.amazonaws.com"
            },
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucketName/prefix/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "accountId"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:codewhisperer:us-
east-1:accountId:*"
                }
            }
        }
    ]
}
```

(オプション) バケットで SSE-KMS を設定する場合は、以下のポリシーを KMS キーに追加します。

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "q.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "accountId"
        },
        "ArnLike": {
           "aws:SourceArn": "arn:aws:codewhisperer:region:accountId:*"
        }
    }
```

```
}
```

Amazon S3 バケット内のデータの保護については、「Amazon Simple Storage Service ユーザーガイド」の「暗号化によるデータの保護」を参照してください。

ユーザーアクティビティレポートを有効にするには

1. Amazon Q Developer コンソールを開きます。

   Amazon Q Developer コンソールを使用するには、 で定義されているアクセス許可が必要です管理者に Amazon Q Developer コンソールの使用を許可する。
2. [設定] を選択します。
3. Q Developer ユーザーアクティビティレポートで、編集を選択します。
4. トグル ユーザーごとに詳細なメトリクスを収集します。
5. S3 の場所で、CSV レポートを保持するために使用する Amazon S3 URI を入力します。例: `s3://amzn-s3-demo-bucket/user-activity-reports/`

ユーザーアクティビティレポートのメトリクス

次の表は、Amazon Q Developer によって生成されたユーザーアクティビティレポートに含まれるメトリクスを示しています。

これらのレポートの詳細については、「」を参照してくださいAmazon Q Developer での特定のユーザーのアクティビティの表示。

メトリクス名	説明
Chat_AICodeLines	Amazon Q によって提案され、ユーザーが受け入れるコード行。このメトリクスには、Amazon Q チャット (インラインチャットではない) を介して生成され、IDE に挿入されたコードが含まれます。
Chat_MessagesInteracted	ユーザーが Amazon Q と肯定的にやり取りしたチャットメッセージの数。肯定的なやり取りの例: リンクのクリック、提案の挿入、Amazon Q からの応答の取り消し。このメトリクス

メトリクス名	説明
	には、Amazon Q チャット (インラインチャットではない) によって生成されたメッセージが含まれます。
Chat_MessagesSent	Amazon Q との間で送受信されるメッセージの数。このメトリクスには、Amazon Q チャット (インラインチャットではない) のユーザープロンプトと Amazon Q レスポンスが含まれます。 ???
CodeFix_AcceptanceEventCount	Amazon Q によって提案され、ユーザーが受け入れたコード修正の数。このメトリクスは、/review コマンドで生成されたコード修正に適用されます。
CodeFix_AcceptedLines	Amazon Q によって提案され、ユーザーが受け入れるコード行。このメトリクスは、/review コマンドで生成されたコード行に適用されます。
CodeFix_GeneratedLines	Amazon Q によって提案されるコード行。このメトリクスは、/review コマンドで生成されたコード行に適用されます。
CodeFix_GenerationEventCount	Amazon Q によって提案されたコード修正の数。このメトリクスは、/review コマンドで生成されたコード修正に適用されます。
CodeReview_FailedEventCount	検出されたが、Amazon Q がコード修正を提案できなかったコードの問題の数。このメトリクスは、/review コマンドを使用して生成されたコードの問題に適用されます。
CodeReview_FindingsCount	Amazon Q で検出されたコードの問題の数。このメトリクスは、/review コマンドを使用して検出されたコードの問題に適用されます。

メトリクス名	説明
CodeReview_SucceededEventCount	見つかったコードの問題のうち、Amazon Q が推奨されるコード修正を生成できたものの数。このメトリクスは、/review コマンドを使用して見つかったコードの問題に適用されます。
Dev_AcceptanceEventCount	Amazon Q によって提案され、ユーザーが受け入れたコード機能の数。このメトリクスは、/dev コマンドで生成されたコード機能に適用されます。
Dev_AcceptedLines	Amazon Q によって提案され、ユーザーが受け入れるコード行。このメトリクスは、/dev コマンドで生成されたコード行に適用されます。
Dev_GeneratedLines	Amazon Q によって提案されるコード行。このメトリクスは、/dev コマンドで生成されたコード行に適用されます。
Dev_GenerationEventCount	Amazon Q によって提案されるコード機能の数。このメトリクスは、/dev コマンドで生成されたコード機能に適用されます。
DocGeneration_AcceptedFileUpdates	Amazon Q によって提案され、ユーザーが受け入れたファイル更新の数。このメトリクスは、/doc コマンドで生成されたファイル更新に適用されます。
DocGeneration_AcceptedFilesCreations	Amazon Q によって提案され、ユーザーが受け入れたファイル作成の数。このメトリクスは、/doc コマンドで生成されたファイル作成に適用されます。
DocGeneration_AcceptedLineAdditions	Amazon Q によって提案され、ユーザーが承諾したドキュメントの追加行。このメトリクスは、/doc コマンドで生成されたドキュメントに適用されます。

メトリクス名	説明
DocGeneration_AcceptedLineUpdates	Amazon Q によって提案され、ユーザーが承諾したドキュメントの更新行。このメトリクスは、/doc コマンドを使用して生成されたドキュメントに適用されます。
DocGeneration_EventCount	ユーザーが /doc コマンドを使用して Amazon Q を使用した回数。
DocGeneration_RejectedFileCreations	Amazon Q によって提案され、ユーザーによって拒否されたファイル作成の数。このメトリクスは、/doc コマンドで生成されたファイル作成に適用されます。
DocGeneration_RejectedFileUpdates	Amazon Q によって提案され、ユーザーによって拒否されたファイル更新の数。このメトリクスは、/doc コマンドで生成されたファイル更新に適用されます。
DocGeneration_RejectedLineAdditions	Amazon Q によって提案され、ユーザーによって拒否されたドキュメントの追加行。このメトリクスは、/doc コマンドで生成されたドキュメントに適用されます。
DocGeneration_RejectedLineUpdates	Amazon Q によって提案され、ユーザーによって拒否されたドキュメントの更新行。このメトリクスは、/doc コマンドを使用して生成されたドキュメントに適用されます。
InlineChat_AcceptedLineAdditions	Amazon Q によって提案され、ユーザーが受け入れるコード追加の行。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて生成されたコードの追加が含まれます。

メトリクス名	説明
InlineChat_AcceptedLineDeletions	Amazon Q によって提案され、ユーザーが受け入れたコード削除の行。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて提案されたコードの削除が含まれます。
InlineChat_AcceptanceEventCount	ユーザーによって受け入れられたインラインチャット (Amazon Q チャットではない) 提案の数。
InlineChat_DismissalEventCount	ユーザーによって中止されたインラインチャット (Amazon Q チャットではない) 提案の数。「中止」とは、コード提案は表示されたが、ユーザーが IDE で入力したり他の操作を実行したりし続け、コード提案を明示的に受け入れたり拒否したりしなかったことを意味します。
InlineChat_DismissedLineAdditions	Amazon Q によって提案され、ユーザーによって中止されたコード追加の行。「中止」とは、コード提案は表示されたが、ユーザーが IDE で入力したり他の操作を実行したりし続け、コード提案を明示的に受け入れたり拒否したりしなかったことを意味します。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて生成されたコードの追加が含まれます。

メトリクス名	説明
InlineChat_DismissedLineDeletions	Amazon Q によって提案され、ユーザーによって中止されたコード削除の行。「中止」とは、コード提案は表示されたが、ユーザーが IDE で入力したり他の操作を実行したりし続け、コード提案を明示的に受け入れたり拒否したりしなかったことを意味します。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて提案されたコードの削除が含まれます。
InlineChat_EventCount	ユーザーが関与したインラインチャット (Amazon Q チャットではない) セッションの数。
InlineChat_RejectedLineAdditions	Amazon Q によって提案され、ユーザーによって拒否されたコード追加の行。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて生成されたコードの追加が含まれます。
InlineChat_RejectedLineDeletions	Amazon Q によって提案され、ユーザーによって拒否されたコード削除の行。このメトリクスには、インラインチャット (Amazon Q チャットではない) を通じて提案されたコードの削除が含まれます。
InlineChat_RejectionEventCount	ユーザーによって拒否されたインラインチャット (Amazon Q チャットではない) 提案の数。
Inline_AICodeLines	Amazon Q によって提案され、ユーザーが受け入れるコード行。このメトリクスには、インライン提案として受け入れられたコードが含まれます。
Inline_AcceptanceCount	ユーザーが受け入れたインライン提案の数。

メトリクス名	説明
Inline_SuggestionsCount	ユーザーに表示されるインライン提案の数。
TestGeneration_AcceptedLines	Amazon Q によって提案され、ユーザーが受け入れるコード行。このメトリクスは、/test コマンドで生成されたコード行に適用されます。
TestGeneration_AcceptedTests	Amazon Q によって提案され、ユーザーが受け入れたユニットテストの数。このメトリクスは、/test コマンドで生成されたユニットテストに適用されます。
TestGeneration_EventCount	ユーザーが /test コマンドを使用して Amazon Q を使用した回数。
TestGeneration_GeneratedLines	Amazon Q によって提案されるコード行。このメトリクスは、/test コマンドで生成されたコード行に適用されます。
TestGeneration_GeneratedTests	Amazon Q によって提案されたユニットテストの数。このメトリクスは、/test コマンドで生成されたユニットテストに適用されます。
Transformation_EventCount	ユーザーが /transform コマンドを使用して Amazon Q を使用した回数。ただし、ユーザーがコマンドラインでコードを変換した回数は除きます。
Transformation_LinesGenerated	Amazon Q によって提案されるコード行。このメトリクスは、/transform コマンドで生成されたコードに適用されます。ただし、コマンドラインで変換されたコードは除きます。
Transformation_LinesIngested	変換のために Amazon Q に提供されるコード行。このメトリクスは、/transform コマンドを通じて提供されるコードに適用されます。ただし、コマンドラインでの変換、または SQL 変換のために提供されるコードは除きます。

Amazon Q Developer でのユーザーのプロンプトのログ記録

管理者は、統合開発環境 (IDE) でユーザーが Amazon Q とやり取りするすべてのインライン提案とチャット会話のログ記録を有効にできます。こうしたログは、監査、デバッグ、分析、コンプライアンスの確保に役立ちます。

デベロッパーがインライン提案を使用する場合、Amazon Q は承認された提案とアクティブに拒否された提案を記録します。デベロッパーが Amazon Q とチャットすると、Amazon Q はデベロッパーのプロンプトと Amazon Q のレスポンスの両方を記録します。デベロッパーが **/dev** コマンドを使用して Amazon Q Agent for software development とチャットする場合、プロンプトのみがログに記録されます

Amazon Q は、作成した Amazon S3 バケットの次のパスにログを保存します。

bucketName/*prefix*/AWSLogs/*accountId*/QDeveloperLogs/*log-type*/*region*/*year*/*month*/*day*/*utc-hour*/*zipFile*.gz/*logFile*.json

前のパスでは、 *log-type*は次のいずれかです。

- `GenerateAssistantResponse` — チャットログを保持
- `GenerateCompletions` — インライン完了ログを保持
- `StartTaskAssistCodeGeneration` — **/dev**ログを保持

ログファイルの内容の例と説明については、「」を参照してくださいAmazon Q Developer のプロンプトログの例。

プロンプトログ記録機能には、ログの保持に使用される Amazon S3 バケットのストレージコストと、バケットの暗号化に使用されるオプションの KMS キーの少額料金以外の料金はかかりません。

プロンプトのログ記録を有効にするには、次の手順を使用します。

前提条件

プロンプトログを保持する Amazon S3 バケットを作成します。バケットは、次の条件を満たす必要があります。

- Amazon Q Developer プロファイルがインストールされている AWS リージョンにいる。このプロファイルは、ユーザーが Amazon Q Developer Pro に初めてサブスクライブしたときにインストールされました。このプロファイルとサポートされているリージョンの詳細については、「」およ

び「」を参照してくださいAmazon Q Developer プロファイルとはQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。

- ユーザーがサブスクライブされている AWS アカウントにいる。ユーザーが複数の AWS アカウントにサブスクライブしている場合は、各アカウントにバケットを作成する必要があります。クロスアカウントバケットはサポートされていません。
- 次のようなバケットポリシーが適用されていること。*bucketName*、*region*、*accountId*、*prefix* はお客様の情報で置き換えてください。

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "QDeveloperLogsWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "q.amazonaws.com"
            },
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucketName/prefix/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "accountId"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:codewhisperer:us-
east-1:accountId:*"
                }
            }
        }
    ]
}
```

バケットで SSE-KMS を設定する場合は、KMS キーに次のポリシーを追加します。

```
{
```

```
    "Effect": "Allow",
    "Principal": {
        "Service": "q.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "accountId"
        },
        "ArnLike": {
           "aws:SourceArn": "arn:aws:codewhisperer:region:accountId:*"
        }
    }
}
```

Amazon S3 バケット内のデータの保護については、「Amazon Simple Storage Service ユーザーガイド」の「暗号化によるデータの保護」を参照してください。

プロンプトのログ記録を有効にするには

1. Amazon Q Developer コンソールを開きます。

   Amazon Q Developer コンソールを使用するには、 で定義されているアクセス許可が必要です管理者に Amazon Q Developer コンソールの使用を許可する。

   > **Note**
   >
   > スタンドアロンアカウント管理者または管理アカウント管理者としてサインインする必要があります。メンバーアカウント管理者は、プロンプトのログ記録を有効にすることはできません。

2. [設定] を選択します。
3. [環境設定] で、[編集] を選択します。
4. [設定の編集] ウィンドウで、[Q Developer プロンプトのログ記録] のトグルを切り替えます。
5. Amazon S3 ロケーションで、ログの受信に使用する Amazon S3 URI を入力します。例: `s3://amzn-s3-demo-bucket/qdev-prompt-logs/`

Amazon Q Developer のプロンプトログの例

このセクションでは、Amazon Q Developer によって生成されたプロンプトログの例を示します。

各例は、ログファイルのフィールドを説明する表です。

プロンプトログの詳細については、「」を参照してくださいAmazon Q Developer でのユーザーのプロンプトのログ記録。

トピック

- インライン提案ログの例
- チャットログの例
- /dev ログの例

インライン提案ログの例

次の例は、ユーザーがインライン提案を受け入れたときに生成されるログファイルを示しています。


```
{
    "records": [
        {
            "generateCompletionsEventRequest": {
                "leftContext": "import * cdk from 'aws-cdk-lib';\r\nimport * s3
 from 'aws-cdk-lib/aws-s3';\r\nimport { Stack, StackProps } from 'constructs';\r
\nexport class MyStack extends Stack {\r\n  constructor(scope: cdk.App, id: string,
 props?: StackProps) {\r\n    super(scope, id, props);\r\n\r\n    new s3.Bucket(this,
 'XXXXXXXX', {\r\n      versioned: true\r\n    });\r\n  }\r\n  ",
                "rightContext": "",
                "fileName": "cdk-modified.ts",
                "customizationArn": null,
                "userId": "d-92675051d5.b8f1f340-9081-70ad-5fc5-0f37151937a6",
                "timeStamp": "2025-01-06T15:09:16.412719Z"
            },
            "generateCompletionsEventResponse": {
                "completions": ["synth() {\n    return
 cdk.App.prototype.synth.apply(this, arguments);\n  }"],
                "requestId": "797c70ee-abc9-4cc7-a148-b9df17f6ce48"
            }
        }
    ]
```

```
}
```

次の表に、インライン候補のログファイルのフィールドを示します。

フィールド名	説明
records	インライン補完とも呼ばれる一連のインライン提案を含む最上位フィールド。
generateCompletionsEventRequest	インラインコード提案のリクエストについて説明します。リクエストは、ユーザーに代わって Amazon Q によって行われます。
leftContext	Amazon Q がインライン提案を生成するコンテキストに使用しているカーソルの前のコードを示します。
rightContext	Amazon Q がインライン提案を生成するコンテキストに使用しているカーソルの後のコードを示します。
fileName	インライン提案がリクエストされたファイルの名前を示します。
customizationArn	インライン提案に適用されるカスタマイズの Amazon リソースネーム (ARN) を示します。この例では、カスタマイズはありません。カスタマイズの詳細については、「」を参照してください提案のカスタマイズ。
userId	インライン提案をリクエストしたユーザーのユーザー ID を示します。
timeStamp	インライン提案がリクエストされた日時を示します。タイムスタンプは協定世界時 (UTC) です。
generateCompletionsEventResponse	Amazon Q によって生成されたインライン提案について説明します。

フィールド名	説明
completions	Amazon Q が提供し、ユーザーが承諾したインライン提案を示します。
requestId	イベントを一意に識別するインラインリクエストの ID。

チャットログの例

次の例は、ユーザーが IDE で Amazon Q を使用してチャットするときに生成されるログファイルを示しています。


```
{
    "records": [
        {
            "generateAssistantResponseEventRequest": {
                "prompt": "Can you explain how to set up object-level access control in
 an S3 bucket using AWS CDK?",
                "chatTriggerType": "MANUAL",
                "customizationArn": null,
                "userId": "d-92675051d5.b8f1f340-9081-70ad-5fc5-0f37151937a6",
                "timeStamp": "2025-01-06T15:05:56.313070Z"
            },
            "generateAssistantResponseEventResponse": {
                "assistantResponse": "I'll explain how to set up object-level access
 control for an S3 bucket using AWS CDK. Here's how you can implement different
 access control configurations: [[1]](https://docs.aws.amazon.com/AmazonS3/latest/
API/s3_example_s3_Scenario_ManageACLs_section.html)\n\n1. Using Bucket Owner Enforced
 (Recommended Modern Approach):\n```typescript\nimport * as cdk from 'aws-cdk-lib';
\nimport * as s3 from 'aws-cdk-lib/aws-s3';\n\nnew s3.Bucket(this, 'XXXXXXXX', {\n
  objectOwnership: s3.ObjectOwnership.BUCKET_OWNER_ENFORCED,\n  blockPublicAccess:
 s3.BlockPublicAccess.BLOCK_ALL\n});\n```\n\n2. Using Public Access Block
 Configuration:\n```typescript\nimport * as cdk from 'aws-cdk-lib';\nimport * as s3
 from 'aws-cdk-lib/aws-s3';\n\nnew s3.Bucket(this, 'XXXXXXXX', {\n  blockPublicAccess:
 new s3.BlockPublicAccess({\n    blockPublicAcls: true,\n    blockPublicPolicy:
 true,\n    ignorePublicAcls: true,\n    restrictPublicBuckets: true\n  })\n});\n```
\n\n3. Setting Access Control for Specific Use Cases:\n```typescript\nimport * as
 cdk from 'aws-cdk-lib';\nimport * as s3 from 'aws-cdk-lib/aws-s3';\nimport * as
 iam from 'aws-cdk-lib/aws-iam';\n\nconst bucket = new s3.Bucket(this, 'XXXXXXXX',
 {\n  // Enable versioning for object-level recovery\n  versioned: true,\n  \n  //
```

```
 Enforce encryption\n  encryption: s3.BucketEncryption.S3_MANAGED,\n  \n  // Block
 all public access\n  blockPublicAccess: s3.BlockPublicAccess.BLOCK_ALL,\n  \n  //
 Enforce object ownership\n  objectOwnership: s3.ObjectOwnership.BUCKET_OWNER_ENFORCED,
\n  \n  // Enable server access logs\n  serverAccessLogsPrefix: 'logs/'\n});\n
\n// Grant read access to specific IAM role\nconst role = new iam.Role(this,
 'MyRole', {\n  assumedBy: new iam.ServicePrincipal('lambda.amazonaws.com'),
\n});\nbucket.grantRead(role);\n\n// Grant write access to specific objects
 with prefix\nbucket.addToResourcePolicy(new iam.PolicyStatement({\n  actions:
 ['s3:PutObject'],\n  resources: [bucket.arnForObjects('uploads/*')],\n  principals:
 [new iam.AccountRootPrincipal()]\n}));\n```\n\n4. Setting up Lifecycle Rules
 for Objects:\n```typescript\nimport * as cdk from 'aws-cdk-lib';\nimport
 * as s3 from 'aws-cdk-lib/aws-s3';\n\nnew s3.Bucket(this, 'XXXXXXXX', {\n
 lifecycle: [\n    {\n      transitions: [\n        {\n          storageClass:
 s3.StorageClass.INFREQUENT_ACCESS,\n          transitionAfter: cdk.Duration.days(30)\n
       },\n        {\n          storageClass: s3.StorageClass.GLACIER,\n
   transitionAfter: cdk.Duration.days(90)\n        }\n      ],\n      expiration:
 cdk.Duration.days(365)\n    }\n  ]\n});\n```\n\nKey points to remember:\n- Always use
 the principle of least privilege\n- Block public access by default unless explicitly
 required\n- Use BUCKET_OWNER_ENFORCED for object ownership when possible\n- Consider
 implementing lifecycle rules for cost optimization\n- Use versioning for sensitive
 data\n- Implement server-side encryption\n- Use resource policies and IAM roles
 for access control instead of ACLs\n- Monitor access patterns using server access
 logging\n\nThese configurations help create a secure and well-managed S3 bucket with
 appropriate access controls at both bucket and object levels.",
                "followupPrompts": "How can you implement multi-factor authentication
 for S3 bucket access using AWS CDK?What are the best practices for setting up object-
level versioning in AWS S3 using CDK?How can you use AWS CDK to create a custom error
 message for unauthorized access attempts on an S3 bucket?",
                "messageMetadata": {
                    "conversationId": "491bd3cd-d70d-40c0-a452-5ef3f0878422",
                    "utteranceId": null
                },
                "codeReferenceEvents": [],
                "supplementaryWebLinksEvent": [
                    {
                        "uri": "https://docs.aws.amazon.com/AmazonS3/latest/API/
s3_example_s3_Scenario_ManageACLs_section.html",
                        "title": "[1] Manage access control lists (ACLs) for Amazon S3
 buckets using an AWS SDK - Amazon Simple Storage Service",
                        "snippet": "The following code example shows how to manage
 access control lists (ACLs) for Amazon S3 buckets.\n\n.NET\n\n**AWS SDK for .NET**
\n\n```\n    using System;\n    using System.Collections.Generic;\n    using
 System.Threading.Tasks;\n    using Amazon.S3;\n    using Amazon.S3.Model;\n\n
   /// <summary>\n    /// This example shows how to manage Amazon Simple Storage
```

```
 Service\n    /// (Amazon S3) access control lists (ACLs) to control Amazon S3
 bucket\n    /// access.\n    /// </summary>\n    public class ManageACLs\n    {\n
        public static async Task Main()\n        {\n            string bucketName
 = \"amzn-s3-demo-bucket1\";\n            string newBucketName = \"amzn-s3-demo-
bucket2\";\n            string keyName = \"sample-object.txt\";\n            string
 emailAddress = \"someone@example.com\";\n\n            // If the AWS Region where
 your bucket is located is different from\n            // the Region defined for
 the default user, pass the Amazon S3 bucket's\n            // name to the client
 constructor. It should look like this:\n            // RegionEndpoint bucketRegion =
 RegionEndpoint.USEast1;\n            IAmazonS3 client = new AmazonS3Client();\n\n
          await TestBucketObjectACLsAsync(client, bucketName, newBucketName, keyName,
 emailAddress);\n        }\n\n        /// <summary>\n        /// Creates a new Amazon
 S3 bucket with a canned ACL, then retrieves the ACL\n        /// information and then
 adds a new ACL to one of the objects in the\n        /// Amazon S3 bucket.\n
  /// </summary>\n        /// <param name=\"client\">The initialized Amazon S3 client
 object used to call\n        /// methods to create a bucket, get an ACL, and add a
 different ACL to\n        /// one of the objects.</param>\n        /// <param name=
\"bucketName\">A string representing the original Amazon S3\n        /// bucket name.</
param>\n        /// <param name=\"newBucketName\">A string representing the name of
 the\n        /// new bucket that will be created.</param>\n        /// <param name=
\"keyName\">A string representing the key name of an Amazon S3\n        /// object
 for which we will change the ACL.</param>\n        /// <param name=\"emailAddress\">A
 string representing the email address\n        /// belonging to the person to whom
 access to the Amazon S3 bucket will be\n        /// granted.</param>\n        public
 static async Task TestBucketObjectACLsAsync(\n            IAmazonS3 client,\n
    string bucketName,\n            string newBucketName,\n            string keyName,
\n            string emailAddress)\n        {\n            try\n            {\n
           // Create a new Amazon S3 bucket and specify canned ACL.\n
 var success = await CreateBucketWithCannedACLAsync(client, newBucketName);\n\n
          // Get the ACL on a bucket.\n                await GetBucketACLAsync(client,
 bucketName);\n\n                // Add (replace) the ACL on an object in a
 bucket.\n                await AddACLToExistingObjectAsync(client, bucketName,
 keyName, emailAddress);\n            }\n            catch (AmazonS3Exception
 amazonS3Exception)\n            {\n                Console.WriteLine($\"Exception:
 {amazonS3Exception.Message}\");\n            }\n        }\n\n        /// <summary>\n
        /// Creates a new Amazon S3 bucket with a canned ACL attached.\n        ///
 </summary>\n        /// <param name=\"client\">The initialized client object used to
 call\n        /// PutBucketAsync.</param>\n        /// <param name=\"newBucketName\">A
 string representing the name of the\n        /// new Amazon S3 bucket.</param>\n
    /// <returns>Returns a boolean value indicating success or failure.</returns>\n
      public static async Task<bool> CreateBucketWithCannedACLAsync(IAmazonS3 client,
 string newBucketName)\n        {\n            var request = new PutBucketRequest()\n
           {\n                BucketName = newBucketName,\n                BucketRegion
 = S3Region.EUWest1,\n\n                // Add a canned ACL.\n                CannedACL
```

```
 = S3CannedACL.LogDeliveryWrite,\n            };\n\n            var response =
 await client.PutBucketAsync(request);\n            return response.HttpStatusCode
 == System.Net.HttpStatusCode.OK;\n        }\n\n\n        /// <summary>\n
   /// Retrieves the ACL associated with the Amazon S3 bucket name in the\n
    /// bucketName parameter.\n        /// </summary>\n        /// <param name=
\"client\">The initialized client object used to call\n        /// PutBucketAsync.</
param>\n        /// <param name=\"bucketName\">The Amazon S3 bucket for which we
 want to get the\n        /// ACL list.</param>\n        /// <returns>Returns an
 S3AccessControlList returned from the call to\n        /// GetACLAsync.</returns>
\n        public static async Task<S3AccessControlList> GetBucketACLAsync(IAmazonS3
 client, string bucketName)\n        {\n            GetACLResponse response = await
 client.GetACLAsync(new GetACLRequest\n            {\n                BucketName =
 bucketName,\n            });\n\n            return response.AccessControlList;\n
    }\n\n\n\n        /// <summary>\n        /// Adds a new ACL to an existing object
 in the Amazon S3 bucket.\n        /// </summary>\n        /// <param name=\"client
\">The initialized client object used to call\n        /// PutBucketAsync.</param>\n
        /// <param name=\"bucketName\">A string representing the name of the Amazon
 S3\n        /// bucket containing the object to which we want to apply a new ACL.</
param>\n        /// <param name=\"keyName\">A string representing the name of the
 object\n        /// to which we want to apply the new ACL.</param>\n        /// <param
 name=\"emailAddress\">The email address of the person to whom\n        /// we will be
 applying to whom access will be granted.</param>\n        public static async Task
 AddACLToExistingObjectAsync(IAmazonS3 client, string bucketName, string keyName,
 string emailAddress)\n        {\n            // Retrieve the ACL for an object.\n
         GetACLResponse aclResponse = await client.GetACLAsync(new GetACLRequest\n
        {\n                BucketName = bucketName,\n                Key = keyName,\n
           });\n\n            S3AccessControlList acl = aclResponse.AccessControlList;
\n\n            // Retrieve the owner.\n            Owner owner = acl.Owner;\n\n
          // Clear existing grants.\n            acl.Grants.Clear();\n\n
  // Add a grant to reset the owner's full permission\n            // (the previous
 clear statement removed all permissions).\n            var fullControlGrant = new
 S3Grant\n            {\n                Grantee = new S3Grantee { CanonicalUser =
 acl.Owner.Id },\n            };\n            acl.AddGrant(fullControlGrant.Grantee,
 S3Permission.FULL_CONTROL);\n\n            // Specify email to identify grantee
 for granting permissions.\n            var grantUsingEmail = new S3Grant\n
    {\n                Grantee = new S3Grantee { EmailAddress = emailAddress },\n
                Permission = S3Permission.WRITE_ACP,\n            };\n\n
  // Specify log delivery group as grantee.\n            var grantLogDeliveryGroup
 = new S3Grant\n            {\n                Grantee = new S3Grantee { URI =
 \"http://acs.amazonaws.com/groups/s3/LogDelivery\" },\n                Permission =
 S3Permission.WRITE,\n            };\n\n            // Create a new ACL.\n
   var newAcl = new S3AccessControlList\n            {\n                Grants = new
 List<S3Grant> { grantUsingEmail, grantLogDeliveryGroup },\n                Owner =
 owner,\n            };\n\n            // Set the new ACL. We're throwing away the
```

```
 response here.\n            _ = await client.PutACLAsync(new PutACLRequest\n
   {\n                BucketName = bucketName,\n                Key = keyName,\n
         AccessControlList = newAcl,\n            });\n        }\n\n    }\n\n\n```\n\n*
 For API details, see the following topics in _AWS SDK for .NET API Reference_.\n  \n
 * GetBucketAcl\n\n* GetObjectAcl\n\n* PutBucketAcl\n\n* PutObjectAcl\n\nFor a complete
 list of AWS SDK developer guides and code examples, see Developing with Amazon S3
 using the AWS SDKs. This topic also includes information about getting started and
 details about previous SDK versions.\n"
                    }
                ],
                "requestId": "dad38fc0-815c-45f7-970a-db916cb7f131"
            }
        }
    ]
}
```

フィールド名	説明
records	プロンプトとレスポンスのセットを含む最上位フィールド。
generateAssistantResponseEventRequest	IDE のチャットウィンドウでユーザーが入力したプロンプトについて説明します。
prompt	ユーザーがチャットウィンドウに入力したプロンプトを示します。
chatTriggerType	MANUAL は、ユーザーがチャットウィンドウにプロンプトを入力したか、チャットウィンドウで提案された質問のいずれかをクリックしたことを示します。 INLINE_CHAT は、ユーザーがメインコーディングウィンドウの小さな入力画面にプロンプトを入力したことを示します。インラインチャットの詳細については、「」を参照してくださいAmazon Q Developer とのインラインチャット。
customizationArn	チャットに適用されるカスタマイズの Amazon リソースネーム (ARN) を示します。この例では、カスタマイズはありません。カスタマイズ

フィールド名	説明
	の詳細については、「」を参照してください提案のカスタマイズ。
userId	プロンプトを入力したユーザーのユーザー ID を示します。
timeStamp	ユーザーがプロンプトを入力した日時を示します。タイムスタンプはユニバーサルタイム (UTC) です。
generateAssistantResponseEventResponse	Amazon Q によって生成されたレスポンスについて説明します。
assistantResponse	Amazon Q がユーザーのプロンプトに提供したレスポンスを示します。
followupPrompts	レスポンスの最後にユーザーに表示されるフォローアッププロンプトの例を示します。
messageMetadata	レスポンスに関連付けられたメタデータについて説明します。
conversationId	レスポンスの会話 ID を示します。会話 ID は、チャットセッションでメッセージをグループ化します。
utteranceId	レスポンスの発話 ID を示します。発話 ID は、ダイアログまたはデータセット内のプロンプトと別のプロンプトを区別するラベルです。
codeReferenceEvents	レスポンスに含まれるコードリファレンスへのリンクについて説明します。
supplementaryWebLinksEvent	レスポンスの最後にユーザーに表示されるリンクを示します。
requestId	イベントを一意に識別するレスポンスの ID。

/dev ログの例

次の例は、ユーザーが IDE の Amazon Q チャットで**/dev**コマンドを入力したときに生成されるログファイルを示しています。

```
{
    "records": [
        {
            "startTaskAssistCodeGenerationEventRequest": {
                "prompt": "write a python application that prints 'hello world!' text
 to the screen and format it in red bold text",
                "chatTriggerType": "MANUAL",
                "conversationId": "da1c95b6-84e1-46a2-9ef9-fe92f5ee169e",
                "customizationArn": null,
                "userId": "d-92675051d5.b8f1f340-9081-70ad-5fc5-0f37151937a6",
                "timeStamp": "2025-01-13T15:40:27.808027101Z"
            },
            "startTaskAssistCodeGenerationEventResponse": {
                "requestId": "e504f126-7197-4e3c-a046-1a10d5a3f3e0"
            }
        }
    ]
}
```

フィールド名	説明
records	プロンプトとレスポンスのセットを含む最上位フィールド。
startTaskAssistCodeGenerationEventRequest	IDE のチャットウィンドウにユーザーが入力した**/dev**プロンプトについて説明します。
prompt	ユーザーがチャットウィンドウに入力した**/dev**プロンプトを示します。
chatTriggerType	MANUAL は、ユーザーがチャットウィンドウにプロンプトを入力したか、チャットウィンドウで提案された質問のいずれかをクリックしたことを示します。 INLINE_CHAT は、ユーザーがメインコーディングウィンドウの小さな入力

フィールド名	説明
	画面にプロンプトを入力したことを示します。インラインチャットの詳細については、「」を参照してくださいAmazon Q Developer とのインラインチャット。
conversationId	レスポンスの会話 ID を示します。会話 ID は、チャットセッションでメッセージをグループ化します。
customizationArn	チャットに適用されるカスタマイズの Amazon リソースネーム (ARN) を示します。この例では、カスタマイズはありません。カスタマイズの詳細については、「」を参照してください提案のカスタマイズ。
userId	プロンプトを入力したユーザーのユーザー ID を示します。
timeStamp	ユーザーがプロンプトを入力した日時を示します。タイムスタンプはユニバーサルタイム (UTC) です。
startTaskAssistCodeGenerationEventResponse	Amazon Q によって生成されたレスポンスについて説明します。現在、**/dev**コマンドへのレスポンスの記録はサポートされていないため、フィールドにはレスポンスは含まれません。
assistantResponse	Amazon Q がユーザーのプロンプトに提供したレスポンスを示します。
requestId	イベントを一意に識別するレスポンスの ID。

Amazon Q Developer がサポートされているリージョン

> **Note**
>
> このページに記載されていないオプトインリージョンから情報を取得することを Amazon Q Developer に要求するリクエストを行うと、Amazon Q はそのリージョンを呼び出すことができます。Amazon Q が呼び出すことができるリージョンへのアクセスを管理するには、「Amazon Q に特定のリージョンでユーザーに代わってアクションを実行するアクセス許可を付与する」を参照してください。

このトピックでは、Amazon Q Developer を使用できる AWS リージョン について説明します。の詳細については AWS リージョン、「 AWS アカウント管理 リファレンスガイド」のAWS リージョン「アカウントで使用できる を指定する」を参照してください。

データは、Amazon Q Developer を使用するリージョンとは異なるリージョンで処理される場合があります。Amazon Q Developer でのクロスリージョン処理の詳細については、「」を参照してくださいクロスリージョン処理。処理中にデータが保存される場所については、「データ保護」を参照してください。

サポートされているリージョン (デフォルトで有効)

Amazon Q Developer は AWS Management Console、以下の AWS Console Mobile Application、、AWS ウェブサイト、 AWS Documentation ウェブサイト、統合チャットアプリケーションで利用できます AWS リージョン。これらのリージョンはデフォルトで有効になっています。つまり、使用前に有効にする必要はありません。詳細については、「デフォルトで有効になっているリージョン」を参照してください。

> **Note**
>
> 内の Amazon Q Developer サービスは AWS Management Console 、「」で説明されているリージョンでのみ使用できますQ Developer コンソールと Q Developer プロファイルでサポートされているリージョン。Amazon Q Developer 設定を管理者として管理するには、Amazon Q Developer サービスに移動し、リージョンセレクタを使用してサポートされているリージョンに切り替える必要があります。

次のリージョンでは、チャットや Amazon Q コンソールの他の機能を使用できます。Amazon Q の機能には、これらのすべてのリージョンで利用できないものもあります。使用する機能のトピックで、提供状況を確認できます。

- 米国東部(オハイオ)
- 米国東部 (バージニア北部)
- 米国西部 (北カリフォルニア)
- 米国西部 (オレゴン)
- アジアパシフィック (ムンバイ)
- アジアパシフィック (大阪)
- アジアパシフィック (ソウル)
- アジアパシフィック (シンガポール)
- アジアパシフィック (シドニー)
- アジアパシフィック (東京)
- カナダ (中部)
- 欧州 (フランクフルト)
- 欧州 (アイルランド)
- 欧州 (ロンドン)
- 欧州 (パリ)
- 欧州 (ストックホルム)
- 南米 (サンパウロ)

サポートされているオプトインリージョン

Amazon Q Developer でオプトインリージョンを使用するには、リージョンを手動で有効にする必要があります。詳細については、「オプトインリージョン」を参照してください。

オプトインリージョンは Amazon Q Developer 無料利用枠でのみサポートされています。以下のオプトインリージョンがサポートされています。

- アフリカ (ケープタウン)
- アジアパシフィック (香港)

- アジアパシフィック (ハイデラバード)
- アジアパシフィック (ジャカルタ)
- アジアパシフィック (マレーシア)
- アジアパシフィック (メルボルン)
- アジアパシフィック (タイ)
- カナダ西部 (カルガリー)
- 欧州 (ミラノ)
- 欧州 (スペイン)
- 欧州 (チューリッヒ)
- メキシコ (中部)
- 中東 (バーレーン)
- 中東 (アラブ首長国連邦)
- イスラエル (テルアビブ)

Amazon Q Developer のトラブルシューティング

このトラブルシューティングガイドは、Amazon Q Developer を使用する際の一般的な問題を解決するのに役立ちます。各セクションでは、問題の説明、考えられる原因、step-by-step解決策、検証手順を明確に示して、迅速に軌道に戻せます。

特定の問題に遭遇する前に、以下の一般的なトラブルシューティング手順を試してください。

- インターネット接続が安定していることを確認する
- 有効な認証情報を使用して Amazon Q にサインインしていることを確認します。
- Amazon Q が IDE またはコマンドラインで最新であることを確認します。
- 問題が解決しない場合は、IDE を再起動するか、「」と入力/quitして Amazon Q コマンドラインを終了し、再起動します。

これらの手順で問題が解決しない場合は、以下の特定のトラブルシューティングセクションを参照してください。

トピック

- Amazon Q Developer ログへのアクセスと使用

Amazon Q Developer ログへのアクセスと使用

Amazon Q Developer は、問題の診断と解決に役立つ詳細なログを生成します。このガイドでは、さまざまな Amazon Q インターフェイスのログにアクセスし、トラブルシューティングに必要な情報を取得するためのログ記録レベルを設定する方法について説明します。

クイックナビゲーション：

- ログアクセスの概要
- IDE 拡張ログ
- Amazon Q CLI ログ
- 一般的なログパターンとソリューション
- ログ分析に関するヘルプの取得

ログアクセスの概要

Amazon Q Developer ログにアクセスするには、サービスの使用方法に応じて、主に 2 つの方法があります。

- IDE 拡張機能 - VS Code および JetBrains IDEsには、Amazon Q 固有のログにアクセスするための「ログの表示」ボタンがあります
- コマンドラインインターフェイス (Amazon Q CLI) - ログは設定可能な詳細レベルを持つ一時ディレクトリにローカルに保存されます。

> **Important**
>
> ログファイルには、ファイルパス、コードスニペット、コマンド出力、アカウント IDs、リソース名など、Amazon Q との会話ややり取りからの機密情報が含まれている場合があります。ログファイルを他のユーザーと共有する場合は注意が必要です。

IDE 拡張機能ログ

IDE インターフェイスを介したログへのアクセス

1. IDE で Amazon Q チャットパネルを開く (VS Code または JetBrains)
2. チャットパネルの右上隅にあるログの表示ボタンをクリックします。
3. 表示される機密警告を確認する
4. ログファイルの場所は、レビューのためにシステムのファイルマネージャーで開きます。

IDE 拡張ログの分析

IDE 拡張ログを確認するときは、以下を探します。

- エラーメッセージ - 「ERROR」または「FATAL」を含む行は重大な問題を示します
- 認証の問題 - 認証または認証情報関連のエラーを探します
- ネットワーク接続 - 接続タイムアウトまたはネットワーク関連のエラー
- 機能固有のエラー - コードの提案、チャット、またはその他の特定の機能に関連する問題

Amazon Q CLI ログ

Amazon Q CLI は、詳細設定に関係なく、すべてのオペレーションの包括的なログを自動的に生成します。ログは常にファイルに書き込まれますが、詳細フラグはターミナル出力に表示される内容のみを制御します。

Amazon Q CLI のログの場所とファイル

Amazon Q CLI ログは、次の場所に自動的に保存されます。

オペレーティングシステム	ログの場所
macOS	`$TMPDIR/qlog/` (通常は `/var/folders/.../qlog/`)
Linux/WSL	`$XDG_RUNTIME_DIR/qlog/` または `$TMPDIR/qlog/` または `/tmp/qlog/`
Windows	`%TEMP%\qlog\`

Amazon Q CLI は、複数の特殊なログファイルを自動的に作成します。

`chat.log` - 以下を含む主要な Amazon Q CLI ラッパーログ。

- Amazon Q CLI の初期化および起動オペレーション
- AWS SDK 呼び出し (Cognito Identity、認証フロー)
- ネットワークオペレーション (HTTP/TLS 接続、証明書処理)
- 低レベルのシステムオペレーション (テレメトリ、ソケット接続)
- AWS サービスエンドポイントの解決と接続プーリング
- インフラストラクチャコンポーネントの詳細なデバッグ情報

`qchat.log` - 以下を含むアプリケーション固有のログをチャットします。

- チャットアプリケーションのエラーと状態処理の問題
- MCP (モデルコンテキストプロトコル) サーバー管理と接続エラー
- アプリケーションレベルの移行の問題
- ユーザーインタラクションの中断とチャット処理の失敗

- 高レベルのアプリケーションロジックエラー

`mcp.log` - Model Context Protocol サーバーログ (MCP サーバーを使用する場合に入力)

`translate.log` - 自然言語からシェルへの翻訳ログ (翻訳機能を使用する場合に入力)

ログファイルの主な違い

範囲と詳細の違い：

- `chat.log`: Q CLI インフラストラクチャ全体を対象とする包括的なシステムレベルのログ記録
- `qchat.log`: チャット機能に特化したアプリケーションレベルのログ記録

コンテンツフォーカスの違い：

- `chat.log`: AWS SDK の内部、ネットワークプロトコル、認証フロー、システムオペレーション
- `qchat.log`: チャットロジック、MCP サーバーのライフサイクル、ユーザーエクスペリエンスの問題、アプリケーションエラー

> **Note**
>
> ログファイルはローカルマシンにのみ保存され、 に送信されません AWS。すべてのログファイルは、詳細なフラグがなくても、最初に CLI を使用すると自動的に作成されます。

Amazon Q CLI のトラブルシューティングワークフロー

ログから診断情報を収集するには、この方法に従います。

1. システムのログディレクトリを特定します。

   Linux/WSL の場合：

   ```
   echo $XDG_RUNTIME_DIR/qlog/
   ```

   macOS の場合：

   ```
   echo $TMPDIR/qlog/
   ```

Windows の場合:

```
echo %TEMP%\qlog\
```

2. 最大の詳細度で Amazon Q CLI コマンドを実行して、ターミナルの詳細な出力を確認します。

```
q -vvv chat
```

3. 発生している問題を再現する
4. Amazon Q CLI を終了し、関連するログファイルを調べます。ほとんどの問題については、両方のメインログファイルを確認してください。

macOS/Linux の場合：

```
less -r $XDG_RUNTIME_DIR/qlog/qchat.log
less -r $XDG_RUNTIME_DIR/qlog/chat.log
```

macOS の代替：

```
less -r $TMPDIR/qlog/qchat.log
less -r $TMPDIR/qlog/chat.log
```

Windows の場合:

```
type %TEMP%\qlog\qchat.log
type %TEMP%\qlog\chat.log
```

5. トラブルシューティング中のリアルタイムのログモニタリングには、以下を使用します。

すべてのログファイルを同時にモニタリングします。

```
tail -f $XDG_RUNTIME_DIR/qlog/*.log
```

特定のファイルをモニタリングします。

```
tail -f $XDG_RUNTIME_DIR/qlog/qchat.log
```

```
tail -f $XDG_RUNTIME_DIR/qlog/chat.log
```

Amazon Q CLI ログの分析

Amazon Q CLI ログは、標準のログ記録レベルを使用して情報を重要度別に分類します。

ERROR

通常のオペレーションを妨げる重大な問題 - トラブルシューティング時にここから始める

WARN

機能を停止しないが、注意すべき潜在的な問題

情報

アプリケーションの動作に関する一般的な運用メッセージ

DEBUG

より詳細な調査に役立つ詳細な技術情報

Amazon Q CLI ログを調べるときは、さまざまなログファイルの以下の主要な領域に焦点を当てます。

`qchat.log` 分析 - 次のようなアプリケーションレベルの問題。

- ERROR chat_cli::cli::chat - チャット処理と状態管理エラー
- ERROR chat_cli::cli::agent - 移行とエージェント関連の問題
- ERROR chat_cli::telemetry - テレメトリの検証と送信の失敗

`chat.log` 分析 - 以下を含むランタイムオペレーションの詳細。

- DEBUG q_cli::cli - Amazon Q CLI コマンドの実行と初期化
- DEBUG aws_sdk_* - AWS SDK オペレーションとサービスコール
- DEBUG rustls::* - TLS/SSL 接続の確立と証明書の処理
- DEBUG hyper_* - HTTP 接続管理とネットワークオペレーション
- エラー fig_telemetry - システムテレメトリとソケット接続の問題

一般的な分析のヒント：

- タイムスタンプ - 問題が発生したときにログエントリを関連付ける
- エラーパターン - 繰り返し発生するエラーやエラーカスケードを探す

- リクエスト IDs - 特定の API コールとその結果を追跡する
- 接続状態 - ネットワーク接続と認証ステータスのモニタリング

> **Tip**
>
> 検索機能を備えた grep、awk、またはテキストエディタなどのツールを使用して、特定のエラーメッセージまたはパターンのログをフィルタリングします。例: grep -i error $XDG_RUNTIME_DIR/qlog/*.log

一般的なログパターンとソリューション

ログとその一般的な解決策で発生する可能性のある一般的な問題は次のとおりです。

MCP サーバー接続エラー

ログパターン (qchat.log): "Background listening thread for client [server-name]: RecvError(Closed)" or "All senders dropped for transport layer"

解決策： MCP サーバープロセスの実行が停止しました。これは通常、Amazon Q CLI を終了するとき、またはサーバーが正常にシャットダウンするときに予想される動作です。

チャット処理の中断

ログパターン (qchat.log): 「現在の状態 err=Interrupted { tool_uses: None } の処理中にエラーが発生しました」

解決策： これは、チャット操作がユーザー (Ctrl+C など) によってキャンセルされ、予想される動作である場合に発生します。

テレメトリ検証エラー

ログパターン (qchat.log): 「cw テレメトリイベント err=ValidationError [ValidationException]: Improperly formed request」

解決策： これらは通常、コア機能に影響を与えない重要なテレメトリ送信の問題です。

移行の警告

ログパターン (qchat.log): 「移行が次の理由で行われませんでした: 移行の中止」

解決策： これは通常、設定の移行に関連する重要ではない警告であり、通常は無視できます。

認証の失敗

ログパターン (chat.log): AWS SDK オペレーションの認証関連のエラー

解決策： q loginを実行して AWS 認証情報を再認証またはチェックする

ネットワーク接続の問題

ログパターン (chat.log 内):「接続タイムアウト」、「ネットワークに到達できません」、または失敗した HTTP 接続

解決策： インターネット接続とファイアウォールの設定を確認する

AWS SDK オペレーションの失敗

ログパターン (chat.log): Cognito ID オペレーションの失敗または認証情報の取得エラー

解決策： AWS 認証情報とネットワーク接続を確認します。再認証が必要になる場合があります

ログ分析のヘルプ

ログの分析や問題の解決にサポートが必要な場合：

- サポートに連絡するときは、関連するログの抜粋を含めます (機密情報を削除) 。
- 問題が発生するタイミングと再現手順に関するコンテキストを提供する

Amazon Q Developer の名称変更 - 変更の概要

2024 年 4 月 30 日、Amazon CodeWhisperer は Amazon Q Developer の一部になりました。このセクションでは、CodeWhisperer でご利用いただいていた機能に関するドキュメントがこのガイドのどこにあるかを紹介しています。

CodeWhisperer の使用から Amazon Q Developer の使用に移行するにあたり、次の変更点が最も重要になると考えられます。

- プロフェッショナル階層 (Amazon Q Developer Pro) の管理設定は、CodeWhisperer Professional の場合とは異なります。
- Amazon Q Developer とチャットするには AWS Management Console、、 AWS ドキュメントおよびマーケティングウェブサイトを使用します。

CodeWhisperer でご利用いただいていた以下の機能は、Amazon Q Developer の一部として使用できます。ただし、いくつか変更点があります。

- サードパーティー IDE でのコーディング提案
- 別の AWS サービスのコンテキストでのコーディング提案
- コマンドラインでの提案
- カスタマイズ
- コードレビュー
- ダッシュボード

Amazon Q Developer ユーザーガイドのドキュメント履歴

以下の表は、Amazon Q Developer ユーザーガイドのドキュメント履歴を説明しています。RSS フィードにサブスクライブして、このドキュメントの更新に関する通知を受け取ることができます。

変更	説明	日付
デプロイオプショントピックを更新しました	デプロイオプショントピックを更新し AWS リージョン、組織ごとに最大 10 個のアカウントでユーザーをサブスクライブできることを示しました。	2025 年 8 月 15 日
トラブルシューティングセクションとトラブルシューティングログセクションを追加	Amazon Q Developer の機能に関する問題の診断と解決に役立つ包括的なトラブルシューティングドキュメントと詳細なログアクセスと分析ガイドを追加しました。	2025 年 8 月 15 日
コマンドライン変換ジョブ履歴セクションを追加	ユーザーが`qct history`コマンドラインから変換ジョブ履歴を表示できるようにする コマンドのドキュメントを追加しました。	2025 年 8 月 7 日
サブスクリプショントピックを更新しました	Amazon Q Developer サブスクリプションの集約リストの表示トピックを更新し、新しい手順と修正を行いました。	2025 年 8 月 6 日
IDE 変換ジョブ履歴トピックを追加	変換ジョブ履歴機能のドキュメントを追加しました。これにより、ユーザーは の最近の Java 変換ジョブからアーティ	2025 年 8 月 6 日

	ファクトを表示、管理、取得できますVisual Studio Code。	
GitLab セルフマネージドインスタンスの更新	GitLab Duo Amazon Q 設定では、GitLab 17.11.0 以降のセルフマネージドインスタンスが必要です。	2025 年 8 月 5 日
トラブルシューティングのトピックを追加しました	サブスクライブしているユーザーのトラブルシューティングトピックを追加しました。	2025 年 8 月 1 日
VSC でエージェントチャットを使用したコードレビューの手順を追加	でコードレビューを実行し、エージェントチャットを使用してコードの問題に対処する手順を追加しましたVisual Studio Code。	2025 年 7 月 31 日
IDE チャットのモデル選択を追加	利用可能なモデルを含むモデルの選択トピックと、IDE でチャット用のモデルを選択する手順を追加しました。	2025 年 7 月 31 日
キーボードショートカットトピックを追加	さまざまな IDEsとプラットフォームで使用できるキーボードショートカットの包括的なリストを提供する Amazon Q Developer トピックのキーボードショートカットを追加しました。	2025 年 7 月 31 日

カスタムエージェント機能を追加	さまざまなユースケースで使用できるツール、アクセス許可、コンテキストを設定することで、特定のワークフローに合わせて Amazon Q Developer CLI をカスタマイズできるカスタムエージェント機能を追加しました。	2025 年 7 月 31 日
チャット履歴圧縮トピックを追加しました	IDE で Amazon Q Developer とチャットするときにコンテキストウィンドウの制限を管理する方法を説明するチャット履歴圧縮を Amazon Q Developer トピックに追加しました。	2025 年 7 月 30 日
プロファイルの削除トピックを追加しました	Amazon Q Developer プロファイルの削除トピックを追加しました。	2025 年 7 月 24 日
テレメトリ収集のオプトアウトステップを追加	変換に Amazon Q コマンドラインツールを使用しているときにテレメトリ収集をオプトアウトする手順を追加しました。	2025 年 7 月 21 日
ファーストパーティの依存関係コンテンツのGitHubワークフローを追加	「コード変換のワークフローのカスタマイズ」セクションを更新し、ファーストパーティの依存関係ページを処理する新しい GitHub ワークフローを追加しました。	2025 年 7 月 16 日

ユーザーアクティビティメトリクストピックを更新しました	ユーザーアクティビティレポートのメトリクストピックを更新し、インラインチャット機能に関連する追加のメトリクスを追加しました。	2025 年 7 月 15 日
プロキシ設定セクションを更新しました	Amazon Q での企業プロキシの設定を更新しEclipse、、JetBrains、Visual Studio の手順を追加しました。	2025 年 7 月 15 日
IDE チャットに追加されたイメージコンテキストのサポート	Amazon Q では、IDE チャットパネルでのコンテキストとしてのイメージの使用がサポートされるようになりました。この機能により、ユーザーはプロンプト時にイメージを含めることができ、UI モックアップやシーケンス図からコードを生成するなどのシナリオが可能になります。	2025 年 7 月 9 日
Builder ID のアップグレードセクションを更新しました	個人アカウントのアップグレード (ビルダー ID) セクションに、複数のビルダー IDs を 1 つの にリンクできないことを示すメモを追加しました。AWS アカウント。	2025 年 7 月 7 日

プロファイル共有セクションを更新しました	プロファイル共有の有効化セクションでは、プロファイル共有が有効になっているときに、管理アカウントのプロファイル設定がメンバーアカウントのプロファイル設定を上書きすることを誤って示していました。このステートメントは削除され、プロファイル共有の目的が明確になりました。	2025 年 7 月 2 日
GitHubコンテンツの Amazon Q Developer にスラッシュコマンドの詳細を追加しました	Amazon Q Developer for GitHub をスラッシュコマンドで呼び出すと、機能開発、コードレビュー、コード変換を実行できます。また、 を呼び出すと、GitHubコンテンツの Amazon Q Developer へのリンクが提供されます。	2025 年 6 月 30 日
IDEs	IDE コード変換コンテンツのセキュリティの維持、コード変換の一時停止またはキャンセル、`qct transform` コマンドの新しいパラメータ、LLM サマリー出力に関する詳細を追加しました。QCT CLI は もサポートするようになりましたhttps://q.eu-central-1.amazonaws.com/ 。	2025 年 6 月 26 日
サードパーティープラットフォームでプロジェクトルールを作成するための新しいコンテンツ	標準とベストプラクティスについては、GitLab と GitHub でプロジェクトルールを作成できます。	2025 年 6 月 26 日

サードパーティー統合の KMS ポリシーを更新	Amazon Q Developer "kms:GenerateDataKey" へのマネージドアクセスのための KMS ポリシーに"kms:Decrypt" "kms:DescribeKey" 、、"kms:Encrypt" 、 を追加しました。 https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/third-party-context-project-rules.html	2025 年 6 月 25 日
プロキシ設定手順を追加	Amazon Q for セクションに企業プロキシの設定を追加しましたVisual Studio Code。	2025 年 6 月 25 日
ユーザーアクティビティレポートの更新	ユーザーアクティビティレポートは UTC 午前 0 時に生成され、それぞれに 1,000 人のユーザーを含む複数のファイルに分割されるようになりました。	2025 年 6 月 24 日
Amazon S3 バケットセクションを更新しました	Amazon S3 バケット URL URLs と ARNs を許可リストセクションに更新しました。 Visual Studio Code	2025 年 6 月 24 日
Amazon Q Developer の運用調査のブランド変更	Amazon Q Developer の運用調査のブランド変更を CloudWatch 調査に反映するために、さまざまなセクションを更新しました。	2025 年 6 月 24 日

GitHub機能開発とコード変換の詳細を更新	Amazon Q Developer for GitHub を呼び出して、問題内のコメントでスラッシュコマンドを使用して機能開発を実行し、コードを変換できるようになりました。	2025 年 6 月 19 日
VS Code IDE チャットに追加されたルールボタン	UI を使用してプロジェクトルールを作成および管理するためのルールボタンをチャットインターフェイスに追加しました。	2025 年 6 月 18 日
VS Code IDE チャットにピン留めされたコンテキストのサポートが追加されました	VS Code にコンテキストピン留め機能を追加し、チャットインタラクション全体で選択したコンテキスト項目を維持し、手動ピン留めとシステム追加コンテキスト管理をサポートしました。	2025 年 6 月 18 日
IDEs でのコード変換の詳細を更新	ソースコードバージョンとターゲットコードバージョンに関するコード変換の詳細、およびプロジェクトライブラリと依存関係をアップグレードするための追加の変換要件を追加しました。複数の差分に関する詳細を削除しました。	2025 年 6 月 17 日
ネットワークセキュリティについて Amazon Q とチャットする	コンソールおよびチャットアプリケーションで、Amazon Q とのネットワークセキュリティについてチャットできます。	2025 年 6 月 17 日

新しいデータストレージトピックを追加	Amazon Q Developer のデータストレージトピックに、コンテンツの保存場所に関する情報を追加しました。	2025 年 6 月 13 日
IDE の MCP	MCP サーバーを IDE の Amazon Q Developer で使用できるようになりました。	2025 年 6 月 12 日
個人アカウント (ビルダー ID) のアップグレード	個人アカウント (ビルダー IDs を持つユーザーは、Pro 階層にアップグレードできるようになりました。	2025 年 6 月 11 日
サブスクリプションステータスの更新	Amazon Q Developer サブスクリプションステータスで使用できないサブスクリプションステータスの説明を更新しました。 https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/q-admin-setup-subscribe-status.html	2025 年 6 月 9 日
CLI での変換の依存関係の指定のサポート	依存関係のアップグレードファイルを指定し、変換用のコマンドラインツールを使用してJavaアップグレードの変換プランを変更できます。	2025 年 6 月 9 日
コマンドラインリファレンスの拡張	新しいコマンドリファレンスセクションでは、Q CLI の呼び出しに使用できる引数について詳しく説明します。	2025 年 6 月 9 日
コマンドラインでのチャットのモデル選択	Amazon Q がコマンドラインでチャットセッションに使用するモデルを選択できるようになりました。	2025 年 6 月 5 日

JetBrains、Visual Studio、EclipseIDEs に拡張されたチャットコンテキストのサポート	Amazon Q は、JetBrains、Visual Studio、IDE Eclipse チャットパネルのコンテキストをサポートするようになりました。	2025 年 6 月 5 日
サポートされているすべての IDEs でのエージェントチャット	エージェントチャットは、サポートされているすべての IDEs で使用できます。	2025 年 6 月 5 日
GitHub更新されたコードレビューの詳細	Amazon Q Developer for GitHub は、GitHubプルリクエスト内でコードレビューを実行できるようになりました。新しいコメントの`/q review`スラッシュコマンドを使用して開始できます。	2025 年 6 月 2 日
チャットのコスト最適化機能	コスト分析に加えて、Amazon Q とチャットして AWS コスト最適化に関するインサイトを得ることができます。	2025 年 6 月 2 日
ファイアウォールページを更新しました	Amazon Q Developer Pro のファイアウォール、プロキシサーバー、またはデータ境界の設定ページの URLs を更新し、追加しました。	2025 年 5 月 22 日
MCP 設定コマンド	Amazon Q の MCP サーバーを設定するために、Q 以外のコマンドラインから直接付与できるコマンド。	2025 年 5 月 21 日

カスタマイズポリシートピックを更新しました	管理者にカスタマイズの作成を許可するトピックでは、カスタマイズポリシーの作成時に発生する可能性のあるアクセス許可エラーについて説明するようになりました。	2025 年 5 月 16 日
トラブルシューティングタスクの更新	Amazon Q Developer Pro サブスクリプションのトラブルシューティングトピックに詳細情報が含まれるようになりました。	2025 年 5 月 15 日
GitLab Duoおよび の新しいコードレビュー動作を更新しました GitHub	自動コードレビューが更新され、新規または再開された GitLab マージリクエストとGitHubプルリクエストでトリガーされます。コードレビューは、後続のコミットによってトリガーされません。	2025 年 5 月 15 日
更新された VPC エンドポイント	Amazon Q Developer およびインターフェイスエンドポイント (AWS PrivateLink) ページの VPC エンドポイントを更新しました。	2025 年 5 月 15 日
MCP サーバーのロードを改善	MCP サーバーがバックグラウンドでロードされるようになりました。これにより、すべてのサーバーが初期化されるのを待たずに、すぐに Amazon Q とのやり取りを開始できます。	2025 年 5 月 15 日

チャットにイメージサポートを追加	Amazon Q は、 イメージモードで `fs_read` ツールを使用して、チャットセッションでイメージを直接分析および説明できるようになりました。	2025 年 5 月 15 日
会話の永続性を追加	Amazon Q は、会話が行われるディレクトリに基づいて会話を自動的に記憶し、会話の状態を手動で管理するための `/export`および `/import` コマンドを提供します。	2025 年 5 月 15 日
マネージドポリシーの更新: AmazonQFullAccess および AmazonQDeveloperAccess	会話履歴を管理するためのアクセス許可が AmazonQFullAccess ポリシーと AmazonQDeveloperAccess ポリシーに追加されました。	2025 年 5 月 14 日
コンソールチャットでの複数の会話のサポート	AWS コンソールで Amazon Q との会話を保存および切り替えることができます。	2025 年 5 月 14 日
サブスクリプションの詳細を追加	アカウント間で Amazon Q Developer Pro にユーザーをサブスクライブするトピックには、IAM Identity Center と Amazon Q Developer プロファイルをインストールする場所に関する情報が含まれるようになりました。	2025 年 5 月 13 日
変換ウェブサイトを削除しました	Amazon Q Developer の変換ウェブサイトが廃止され、関連ドキュメントも削除されました。	2025 年 5 月 12 日

更新された用語	このガイド全体で、アイデンティティ対応コンソールセッションという用語をアイデンティティ拡張コンソールセッションに更新しました。	2025 年 5 月 10 日
SCP の例を追加しました	このサービスコントロールポリシー (SCP) は、EU リージョン外の Amazon Q へのアクセスを拒否します。	2025 年 5 月 8 日
の Amazon Q Developer GitHub	Amazon Q Developer for GitHubに関する情報。セットアップ、主要な機能、設定に焦点を当てた概念と手順が含まれます。	2025 年 5 月 5 日
コンテキストフックを追加	コンテキストフックのサポートが追加されました。	2025 年 5 月 3 日
管理ポリシーが更新されました	アクセス許可が AmazonQFullAccess に追加されました。	2025 年 5 月 2 日
IDE のエージェントチャット	エージェントチャット機能は IDE で使用できます。	2025 年 5 月 1 日
カスタマイズの更新	カスタマイズで追加の言語がサポートされるようになりました。	2025 年 4 月 30 日
管理ポリシーとポリシー例の更新	アクセス許可が AmazonQFullAccess、GitLabDuoWithAmazonQPermissionsPolicy、および管理者によるプラグインの設定を許可に追加されました。	2025 年 4 月 30 日

MCP サポートを追加	CLI で MCP のサポートを追加しました。	2025 年 4 月 29 日
アップグレードの詳細を追加	無料利用枠から Pro 利用枠にアップグレードする方法に関する情報を追加しました。	2025 年 4 月 28 日
会話履歴のサポート	これで、IDE で Amazon Q とチャットするときに会話履歴が保存されました。	2025 年 4 月 21 日
コンテキストとしてのコードのサポート	IDE で Amazon Q とチャットするときに、クラス、関数、グローバル変数をコンテキストとして指定できるようになりました。	2025 年 4 月 21 日
Amazon Q オンボーディングとポリシーGitLab Duoによるの更新	GitLab Duo での Amazon Q が更新され、オンボーディングおよびアクセス許可ポリシー (GitLabDuoWithAmazonQPermissionsPolicy) が変更されました。	2025 年 4 月 16 日
ダッシュボードのアクセス許可を更新しました	Amazon Q Developer ダッシュボードの表示に必要なアクセス許可のリストを更新しました。	2025 年 4 月 15 日
コマンドラインのセキュリティドキュメントを改善	セキュリティに関する考慮事項、ベストプラクティス、ツールのアクセス許可の安全な使用に関する包括的なガイダンスを含む、セキュリティドキュメントの再編成と強化。	2025 年 4 月 13 日

コマンドラインのセキュリティと設定の強化	設定オプションを含む新しいコマンドライン設定セクションを追加しました。機密環境のセキュリティのベストプラクティスを含むツールアクセス許可のドキュメントを強化しました。	2025 年 4 月 12 日
サブスクリプションエクスペリエンスの更新	Amazon Q Developer Pro にユーザーをサブスクライブし、Amazon Q Developer プロファイルをインストールするワークフローが Amazon Q コンソールから Amazon Q Developer コンソールに移動されました。	2025 年 4 月 10 日
インラインチャットは で利用できます Eclipse	で Amazon Q とインラインでチャットできますEclipse。	2025 年 4 月 10 日
Amazon Q Developer プロファイルが欧州 (フランクフルト) で利用可能に	Amazon Q Developer にサブスクライブすると、欧州 (フランクフルト) リージョンでプロファイルを作成できます。	2025 年 4 月 10 日
/tools 機能が CLI に追加されました	/tools コマンドを使用して、Amazon Q がシステムでアクションを実行するために使用するツールのアクセス許可を管理できます。	2025 年 4 月 10 日
英語以外の自然言語のサポート	IDE およびコマンドラインで Amazon Q とチャットできます。 https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/command-line.html	2025 年 4 月 9 日

Amazon Q GitLab Duoを使用した の更新	GitLab Duo Amazon Q での は、インラインポリシーの変更に関して更新されており、オプションで CMK ポリシーを作成できます。この機能は/fix削除されました。	2025 年 4 月 8 日
変換に関する E メール通知	変換に関連する更新に関する E メール通知を受け取る場合があります。	2025 年 4 月 8 日
新しいコンテキスト、プロンプト、プロジェクトルールのトピック	チャットへのコンテキストの追加、プロンプトの保存、プロジェクトルールの作成のトピックが追加されました。	2025 年 4 月 4 日
サブスクリプショントピックの更新	サブスクリプションの理解、集約されたサブスクリプションの表示、プロファイル共有の有効化のトピックが修正されました。	2025 年 3 月 25 日
ポリシーの更新例	「Amazon Q コンソールの使用を管理者に許可する」および「Amazon Q Developer コンソールの使用を管理者に許可する」のポリシー例が、 アクセスsso:CreateInstance 許可で更新されました。	2025 年 3 月 24 日
カスタマイズでの C++ と C# のサポート	カスタマイズで C++ と C# がサポートされるようになりました。	2025 年 3 月 20 日

リソースに関するチャットの更新	複数の AWS リソースとサービスについて Amazon Q とチャットして、 AWS インフラストラクチャと設定に関する回答を取得できます。	2025 年 3 月 13 日
ドキュメント生成の追加言語サポート	ドキュメント生成のエージェントが C++ と C# をサポートするようになりました。	2025 年 3 月 12 日
新しいサブスクリプション関連の制限	Amazon Q Developer Pro にユーザーをサブスクライブするトピックを更新し、 が管理する組織 AWS アカウント 内で Amazon Q Developer を最大 50 個まで有効にできることを示しました AWS Organizations。	2025 年 3 月 6 日
CLI チャットへのコンテキスト統合	Amazon Q CLI にコンテキスト統合が追加され、Amazon Q がユースケースをより深く理解し、より関連性の高いコンテキスト対応のレスポンスを提供できるようになりました。	2025 年 3 月 6 日
ポリシーの修正	「管理者が Amazon Q コンソールを使用できるようにする」で説明されているポリシーで JSON 構文エラーが修正されました。	2025 年 2 月 28 日

変換用のコマンドラインツールの新しいバージョン	変換用のコマンドラインツールの最新バージョンには、 を介した IAM による認証のサポートが含まれています AWS CLI。	2025 年 2 月 28 日
Pro 階層へのアップグレード	Amazon Q Developer 無料利用枠トピックの Pro 利用枠にアップグレードする方法に関する情報を追加しました。	2025 年 2 月 25 日
カスタマイズポリシーの更新	アクセス許可がカスタマイズポリシーに追加されました。	2025 年 2 月 25 日
新しいダッシュボードトピック	次のトピックが追加されました。Amazon Q Developer ダッシュボードの使用状況メトリクスの説明。	2025 年 2 月 21 日
新しいクロスリージョン処理トピック	クロスリージョン処理トピックでは、Amazon Q Developer がリクエストを処理し、 間で呼び出し AWS リージョン を行ってサービスを提供する方法について説明します。	2025 年 2 月 21 日
マネージドポリシーの更新	アクセス許可が AWSServiceRoleForUserSubscriptions に追加されました。	2025 年 2 月 21 日
/doc の機能強化	Amazon Q は、**/doc**コマンドに応じてインフラストラクチャ図を生成できるようになりました。	2025 年 2 月 20 日

新しいサブスクリプショントピック	Amazon Q Developer サブスクリプションステータスと集約された Amazon Q Developer サブスクリプションの表示の 2 つのサブスクリプション関連トピックが追加されました。	2025 年 2 月 19 日
チャットアプリケーションの Amazon Q Developer の章	チャットアプリケーションの Amazon Q Developer がチャットアプリケーションの Amazon Q Developer になりました。新しい章では、サポートされている機能について説明します。	2025 年 2 月 19 日
Java 21 変換のサポート	IDE およびコマンドラインで Java アプリケーションを Java 21 にアップグレードできます。	2025 年 2 月 14 日
新しいファイアウォールトピック	Amazon Q Developer のファイアウォールまたはプロキシサーバーの設定トピックが追加されました。	2025 年 2 月 14 日
変換用のコマンドラインツールの新しいバージョン	変換用のコマンドラインツールの最新バージョンには、Java アプリケーションでの埋め込み SQL の変換のサポートが含まれています。	2025 年 2 月 12 日
ユーザーアクティビティレポートの修正	ユーザーアクティビティレポート CSV ファイルへのパスが修正されました。	2025 年 2 月 10 日

変換されたコードの保持期間の更新	Amazon Q は、変換されたコードを 24 時間から 30 日間保持するようになりました。	2025 年 2 月 7 日
新しいサブスクリプションワークフロー	Amazon Q Developer にユーザーをサブスクライブする手順が改善されました。	2025 年 2 月 6 日
変換用のコマンドラインツールの新しいバージョン	変換用のコマンドラインの最新バージョンには、アップグレードされた Java コードを複数のコミットで受信する機能が含まれています。	2025 年 2 月 3 日
/dev の機能強化	Amazon Q は、**/dev**コマンドに応答して生成するコードをテストできるようになりました。	2025 年 1 月 31 日
カスタマイズセクションの更新	カスタマイズの作成トピックでは、カスタマイズに任意の数のリポジトリを含めることができることがわかりました。	2025 年 1 月 24 日
プロンプトログ記録の例	プロンプトログ記録の有効化セクションに、サンプルログが含まれるようになりました。	2025年1月23日
CloudZero プラグイン	CloudZero プラグインは Amazon Q チャットで利用できます。	2025年 1 月 15 日
ユーザーアクティビティレポートの更新	新しいメトリクスがユーザーアクティビティレポートに追加されました。	2024 年 12 月 16 日

ダッシュボードの更新	古いダッシュボードに関する情報が Amazon Q Developer Pro ダッシュボードセクションから削除されました。フィルターとメトリクスに関する情報が追加されました。	2024 年 12 月 16 日
Amazon Q を使用したトラブルシューティング	リソースのトラブルシューティングを Amazon Q に依頼するセクションが追加されました。	2024 年 12 月 13 日
ID 拡張セッションの更新	ID 拡張コンソールセッションを有効にする手順については、「組織インスタンスを使用した Amazon Q Developer Pro 階層へのユーザーのサブスクライブ」セクションで説明されています。	2024 年 12 月 6 日
新しいテスト生成エージェント	Amazon Q テスト生成機能を使用してユニットテストを生成できます。	2024 年 12 月 3 日
大規模な変換	Amazon Q は、.NET、メインフレーム、VMware ワークロードを一括で変換できます。	2024 年 12 月 3 日
GitLab デュオと Amazon Q	の概念、開始手順、トラブルシューティングなど、GitLab DuoAmazon Q に関する情報。	2024 年 12 月 3 日
IDE でのドキュメント生成	Amazon Q は、サポートされている IDE でコードの READMEs を生成できます。IDEs	2024 年 12 月 3 日

IDE でのコードレビュー	Amazon Q コードレビュー、以前のセキュリティスキャンでは、サポートされている IDEs でコードの問題を検出して対処できます。	2024 年 12 月 3 日
IDE での .NET 変換	Amazon Q は、プレビューで利用可能な Visual Studio の Linux 互換クロスプラットフォームアプリケーションに .NET アプリケーションを移植できます。	2024 年 12 月 3 日
コマンドラインでの変換	Java アプリケーションは、プレビューで利用可能なコマンドラインで変換できます。	2024 年 11 月 27 日
IDE での変換の複数の差分	複数の差分で Amazon Q から変換変更を受け取ることを選択できます。	2024 年 11 月 27 日
の Amazon Q Eclipse	Amazon Q プラグインは、 のプレビューで利用できます Eclipse。	2024 年 11 月 27 日
コスト分析	以前はプレビューで利用可能だったコスト分析機能が一般公開されました。	2024 年 11 月 26 日
埋め込み SQL コードの変換	IDE の Amazon Q 変換を使用して、Java アプリケーションの埋め込み SQL コードを変換できます。	2024 年 11 月 22 日
ダッシュボードの更新	Amazon Q Developer Pro ダッシュボードが新しいメトリクスで更新されました。	2024 年 11 月 22 日

CodeConnections リポジトリ	CodeConnections 接続を使用してカスタマイズを作成するときに、使用するリポジトリを選択できるようになりました。	2024 年 11 月 22 日
Amazon Q コマンドラインが Linux をサポートするようになりました	Amazon Q コマンドラインは Linux 環境をサポートしています。Ubuntu 22 および 24 をサポートしており、それ以外の場合は GNOME v42+ またはディスプレイサーバーが Xorg で入力メソッドフレームワークが IBus である環境で動作する場合があります。	2024 年 11 月 21 日
ユーザーをサブスクライブする	Amazon Q Developer Pro 階層へのアクセスの設定でユーザーをサブスクライブする手順が更新され、新しいユーザーインターフェイス (UI) 要素が反映されました。	2024 年 11 月 20 日
カスタマイズの変更	チャットでのカスタマイズ機能が一般利用可能になりました。また、、、.md、、.mdx.rstおよびのファイルタイプを使用してカスタマイズを作成できるようになりました.txt。	2024 年 11 月 20 日

サポートされている IAM アイデンティティセンターリージョン	Amazon Q Developer Pro サブスクリプションの IAM アイデンティティセンターインスタンスをセットアップできるリージョンに関する情報を含むセクションが追加されました。	2024 年 11 月 18 日
言語の追加	Dart、Lua、R、Swift、SystemVerilog、Powershell のサポートが追加され、JSON と YAML のサポートが拡張されました。	2024 年 11 月 18 日
カスタマーマネージドキーのサポート	カスタマーマネージドキーとそれらで暗号化できる機能の使用に関する情報が、データ暗号化トピックに追加されました。	2024 年 11 月 18 日
クロスリージョン推論	Amazon Q Developer でのクロスリージョン推論に関するトピックが追加されました。	2024 年 11 月 18 日
Amazon Q Developer Pro クォータ	Pro 階層クォータセクションが追加されました。	2024 年 11 月 18 日
マネージドポリシーの更新: AmazonQFullAccess	AmazonQFullAccess ポリシーにアクセス許可が追加されました。	2024 年 11 月 13 日
マネージドポリシーの更新: AmazonQDeveloperAccess	AmazonQDeveloperAccess ポリシーに追加のアクセス許可が追加されました。	2024 年 11 月 13 日

Amazon Q プラグイン	プラグインを使用すると、ユーザーはサードパーティーのツールが提供するメトリクスについて Amazon Q とチャットできます。	2024 年 11 月 13 日
ユーザーアクティビティレポート	ユーザーアクティビティレポートを有効にできるようになりました。	2024 年 11 月 8 日
カスタマイズセクションの更新	データの準備セクションでは、ファイルとディレクトリの命名に関する制限について説明します。	2024 年 11 月 5 日
Amazon Q Developer Pro セクションを明確にしました	Amazon Q Developer Pro にユーザーをサブスクライブする手順を明確にしました。	2024 年 11 月 1 日
インラインチャット	新しいインラインチャット機能を使用してコードを変換できます。	2024 年 10 月 29 日
マネージドポリシーの更新: AmazonQFullAccess および AmazonQDeveloperAccess	AmazonQFullAccess ポリシーと AmazonQDeveloperAccess ポリシーにアクセス許可が追加されました。	2024 年 10 月 28 日
カスタマイズに関するセクションの修正	「カスタマイズの作成」セクションに、コードベースがバケットのルートではなく Amazon S3 のフォルダに存在する必要があるという説明が追加されました。	2024 年 10 月 28 日
プロンプトのログ記録に関するセクションの説明の明確化	「プロンプトのログ記録の有効化」セクションの文言が明確になりました。	2024 年 10 月 24 日

Amazon S3 バケットポリシーの修正	「プロンプトのログ記録の有効化」に表示される Amazon S3 バケットポリシーに含まれていた JSON 構文エラーが修正されました。	2024 年 10 月 22 日
機能に関する章の拡張	Amazon Q Developer のさまざまな機能を説明する章が大幅に拡張されました。	2024 年 10 月 3 日
Console-to-Code	以前は Amazon EC2 の機能としてプレビューで利用可能だった Console-to-Code が、Amazon Q Developer の機能として一般提供されるようになりました。Amazon EC2、Amazon VPC、Amazon RDS と統合されています。	2024 年 10 月 3 日
新しいポリシー: で Amazon Q CLI を使用する AWS CloudShell	ID ベースのポリシーでは、ユーザーは で Amazon Q CLI を使用できます AWS CloudShell。	2024 年 10 月 2 日
プロンプトのログ記録	ユーザーの IDE プロンプトを Amazon S3 バケットにログ記録できます。	2024 年 9 月 16 日
セットアップのコンテンツの更新	「使用開始」の章が大幅に簡素化され、再構成されました。	2024 年 8 月 15 日
IDE VPC アクセスに必要な CodeWhisperer エンドポイント	Amazon VPC からのアクセスには、q エンドポイントと codewhisperer エンドポイントの両方を含める必要があります。	2024 年 7 月 18 日

新しいエンドポイント	エンドポイントで、codewhisperer の代わりに q 文字列を使用できるようになりました。	2024 年 7 月 12 日
カスタマイズの一般提供	カスタマイズ機能が一般提供されるようになりました。	2024 年 7 月 10 日
カスタマイズに関するチャット (プレビュー)	プレビューで、カスタマイズ機能を使用してコードベースに関連する質問を行うことができます。	2024 年 7 月 10 日
マネージドポリシーの更新: AmazonQFullAccess	AmazonQFullAccess ポリシーにアクセス許可が追加されました。	2024 年 7 月 9 日
新しいマネージドポリシー: AmazonQDeveloperAccess	AmazonQDeveloperAccess マネージドポリシーは、Amazon Q Developer とのやり取りを可能にするフルアクセスを、管理者アクセスなしで提供します。	2024 年 7 月 9 日
Amazon Q Developer 管理ポリシーの更新	Amazon Q Developer 管理者に権限を与えるポリシーが更新され、 が追加されましたsso:ListProfiles 。	2024 年 6 月 19 日
信頼されたアクセスに関するセクション	新しいセクションでは、Amazon Q Developer 管理者がメンバーアカウントと設定を共有する方法について、よりわかりやすく説明しています。	2024 年 6 月 19 日

セットアップ手順の更新	「使用開始」の章が改善され、アカウントインスタンスのサポートが追加されました。	2024 年 6 月 6 日
コード例の更新	コード例に C と C++ が追加され、C# の例が改善されました。	2024 年 6 月 6 日
マネージドポリシーの更新: AmazonQFullAccess	AmazonQFullAccess ポリシーにアクセス許可が追加されました。	2024 年 4 月 30 日
新しいサービスリンクロール: AWSServiceRoleForUserSubscriptions	AWSServiceRoleForUserSubscriptions サービスリンクロールにより、ユーザーサブスクリプションが IAM アイデンティティセンターリソースにアクセスできるようになり、サブスクリプションが自動的に更新されます。	2024 年 4 月 30 日
新しいサービスリンクロール: AWSServiceRoleForAmazonQDeveloper	AWSServiceRoleForAmazonQDeveloper サービスリンクロールは、データにアクセスしてデータを生成し、レポートを作成するためのアクセス許可を付与します。	2024 年 4 月 30 日
新しいマネージドポリシー: AWSServiceRoleForUserSubscriptionPolicy	AWSServiceRoleForUserSubscriptionPolicy を使用すると、プリンシパルは IAM アイデンティティセンターのディレクトリと AWS Organizations 変更を追跡できます。	2024 年 4 月 30 日

新しいマネージドポリシー: AWSServiceRoleForAmazonQDeveloperPolicy	AWSServiceRoleForAmazonQDeveloperPolicy を使用すると、Amazon Q Developer はユーザーに代わって CloudWatch と CodeGuru を呼び出すことができます。	2024 年 4 月 30 日
GA リリース	一般ユーザーが Amazon Q Developer を利用できるようになりました。	2024 年 4 月 30 日
Amazon CodeWhisperer の統合	Amazon CodeWhisperer が Amazon Q Developer の一部になりました。	2024 年 4 月 30 日
新しいガイド名	このサービスおよび付属のユーザーガイドが、Amazon Q Developer という名前に変更されました。	2024 年 3 月 29 日
新しいアクセス許可	コンソールで Amazon Q とチャットするには、ListConversations アクションが必要です。	2024 年 3 月 5 日
データ保護に関する新しいトピック	Amazon Q では、サービス改善の目的でコンテンツを使用するようになりました。	2024 年 1 月 25 日
新しいトピック	チャットアプリケーションで Amazon Q Developer で設定された SlackおよびMicrosoft Teamsチャネルに Amazon Q を追加する方法の手順を追加しました。	2024 年 1 月 18 日

プレビューリリース	Amazon Q Developer ユーザーガイドの初回リリースです。	2023 年 11 月 28 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。