

ユーザーガイド

Amazon Q Developer



Amazon Q Developer: ユーザーガイド

Copyright © 2026 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
開始方法	4
サービスのティア	4
空き	4
Pro (プロ)	5
個人アカウントで使用を開始する	6
Builder ID の制限	7
ステップ 1: サインアップ	7
ステップ 2: Amazon Q をインストールする	7
ステップ 3: (オプション) Pro ティアにアップグレードする	8
IAM アイデンティティセンターを使用して開始する	8
ステップ 1: デプロイオプションを選択する	8
ステップ 2: ユーザーをサブスクライブする	14
Pro ティアサブスクリプション	28
サポート対象のリージョン	28
サブスクリプションの請求	31
サブスクリプションステータス	32
開始 URL の表示	34
暗号化方法の管理	35
Q Developer プロファイル	37
サブスクリプションのトラブルシューティング	41
サブスクリプションの集約リストの表示	44
サブスクリプション解除	46
Pro ティアへのアップグレード	50
Kiro へのアップグレード	53
オン AWS	54
Amazon Q Developer Pro サブスクリプションへの認証	54
に関するチャット AWS	55
Amazon Q での Q アーティファクトの使用	56
アクセス許可を追加する	58
会話を開始する	58
コンソールで会話を管理する	58

Amazon Q チャットパネルを操作する	59
チャット設定	60
プロンプトの例	60
リソースに関するチャット	61
Amazon Q にリソースのトラブルシューティングを依頼する	64
コストに関するチャット	70
ネットワークセキュリティに関するチャット	74
E メール送信に関するチャット	75
テレメトリと運用に関するチャット	76
プラグインの使用	77
CloudZero	78
Datadog	85
Wiz	93
Console-to-Code	100
Console-to-Code	100
Console-to-Code を使用できる場所	101
アクセス許可の付与	102
使用	102
コンソールエラーの診断	104
アクセス許可を追加する	105
コンソールでの一般的なエラーの診断	105
サポートとのチャット	105
前提条件	106
適切なサービスを指定する	106
サポートケースの作成	107
フィードバックを残す	109
IDE 内	110
サポートされている IDE	110
Amazon Q のインストール	111
サポートされている IDE バージョン	112
Eclipse IDE 内	112
JetBrains IDE 内	113
Visual Studio Code 内	115
Visual Studio の場合	116
AWS コンソールの IAM プリンシパル	119
コードに関するチャット	119

IDE で Amazon Q の操作	120
タスクの例	121
質問例	122
応答に関する問題の報告	123
コードのレビュー	123
コードの変換	139
コードの説明と更新	191
インラインでのチャット	191
チャットへのコンテキストの追加	194
チャット履歴の圧縮	203
会話の管理をする	205
チャットでのショートカットキーの使用	207
モデルの選択	208
インライン提案の生成	209
提案の一時停止	210
Amazon Q のコード補完の仕組み	215
コーディング環境で AWS の提案	221
ショートカットキーの使用	229
コードリファレンスの使用	235
コードの例	246
サポートされている言語	269
インライン提案	269
変換	270
コードレビュー	271
カスタマイズ	271
Q CLI を使用する	273
MCP の使用	274
MCP の概要	274
CLI で	275
設定コマンド	275
リモート MCP サーバー	276
MCP の設定	277
Q CLI を使用した MCP サーバーの設定	277
IDE で Q を使用して MCP サーバーを設定する	277
MCP サーバーのロード	277
ツールとプロンプト	278

MCP ツールについて	278
使用可能なツールの検出	278
ツールの使用方法	279
プロンプトの使用	279
IDE を使用する	279
MCP IDE 設定ファイル	279
UI へのアクセス	280
MCP サーバーの追加	281
MCP 設定のトラブルシューティング	283
MCP サーバーの有効化	283
MCP サーバーの無効化	283
現在有効になっている MCP サーバーの削除	283
現在無効になっている MCP サーバーの削除	284
ツールのアクセス許可の確認と調整	284
主な利点	284
MCP アーキテクチャ	284
コア MCP の概念	285
ツール	285
プロンプト	286
リソース	286
MCP セキュリティ	286
セキュリティモデル	287
セキュリティに関する考慮事項	287
MCP ガバナンス	287
組織の MCP を無効にする	288
組織の MCP 許可リストの指定	288
サードパーティー統合	302
GitLab Duo with Amazon Q Developer	302
Amazon Q Developer for GitHub (プレビュー)	302
Amazon Q Developer のプロジェクトルール	303
GitLab Duo	303
GitLab Duo の概念	304
開始方法	308
トラブルシューティング	308
GitHub (プレビュー)	310
Amazon Q Developer アプリのインストールとアクセスの承認	311

Amazon Q Developer エージェント	312
アプリのインストールの登録	313
GitHub でのブラウザ拡張機能の使用	313
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	314
クイックスタート	314
機能開発とイテレーション	317
コードのレビュー	320
使用制限の引き上げと設定	323
設定	325
トラブルシューティング	327
プロジェクトルールの作成	328
チャットアプリケーションで	330
お使いのチャンネルで Amazon Q チャットを有効にする	330
チャンネルで Amazon Q に質問する	331
セキュリティ	333
データ保護	334
データストレージ	335
データ暗号化	336
サービスの改善	338
IDE およびコマンドラインでのデータ共有のオプトアウト	339
クロスリージョン処理	347
ID とアクセス管理	349
オーディエンス	350
アイデンティティを使用した認証	350
ポリシーを使用したアクセスの管理	354
Amazon Q が IAM で機能する仕組み	356
Amazon Q へのアクセスを管理する	362
統合のための Amazon Q Developer へのアクセスを管理する	402
Amazon Q アクセス許可リファレンス	404
AWS Amazon Q の マネージドポリシー	407
サービスにリンクされたロールの使用	413
コンプライアンス検証	421
耐障害性	422
インフラストラクチャセキュリティ	422
ファイアウォール、プロキシ、データ境界	423
許可リストの一般的な URL	423

Amazon S3 バケット URL および ARN の許可リストへの追加	424
Amazon Q での企業プロキシの設定	426
VPC エンドポイント (AWS PrivateLink)	432
Amazon Q の VPC エンドポイントに関する考慮事項	433
前提条件	433
Amazon Q 用のインターフェイス VPC エンドポイントの作成	433
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	434
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	434
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	435
モニタリングと追跡	436
で AWS CloudTrail	437
CloudTrail での Amazon Q Developer の情報	437
Amazon Q Developer ログファイルエントリの概要	438
CloudWatch の使用	442
特定のユーザーによるアクションの特定	444
カスタマイズ関連ログへのアクセス	461
使用状況メトリクスの表示 (ダッシュボード)	462
ダッシュボードメトリクス	464
ダッシュボードを無効にする	467
ダッシュボードのトラブルシューティング	467
ユーザーごとのアクティビティの表示	468
ユーザーアクティビティレポートのメトリクス	471
ユーザープロンプトのログ記録	477
プロンプトログの例	480
サポート対象のリージョン	491
サポートされているリージョン (デフォルトで有効)	491
サポートされているオプトインリージョン	492
トラブルシューティング	494
ログへのアクセスと分析	494
ログへのアクセスの概要	495
IDE 拡張ログ	495
Amazon Q CLI ログ	496
一般的なログパターンとソリューション	500
ログ分析に関するヘルプの取得	501
Amazon Q Developer のサービス名称変更	502

ドキュメント履歴	503
.....	dxxxvi

Amazon Q Developer とは？

Note

Amazon Bedrock を利用: Amazon Q Developer は Amazon Bedrock をベースに構築されています。Amazon Bedrock に実装された[不正使用の自動検出](#)機能を搭載し、安全、セキュリティ、AI の責任ある使用を強化しています。

Amazon Q Developer は、生成人工知能 (AI) を活用した会話アシスタントであり、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 マネジメントコンソール、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

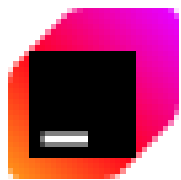
します AWS Console Mobile Application。詳細については、「[AWS アプリやウェブサイトでの Amazon Q Developer の使用](#)」を参照してください。

IDE

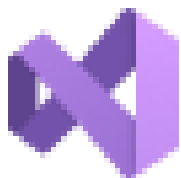
Amazon Q 拡張機能をダウンロードし AWS ビルダー ID、(AWS アカウント不要) を使用して無料でサインインします。



[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](#)



[Eclipse IDE で Amazon Q をダウンロードする \(プレビュー\)](#)

IDE から Amazon Q アイコンをクリックして、チャットや開発ワークフローを開始できます。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。

Amazon Q Developer in chat applications

[AmazonQDeveloperAccess](#) マネージドポリシーを、使用中の IAM ID と、Microsoft Teams または Slack アプリケーションのチャンネルガードレールに追加します。詳細については、「[Amazon Q Developer in chat applications でのチャット](#)」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 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ティア (Free および Pro)、使用可能なさまざまなサインイン方法、AWS IAM アイデンティティセンターの個人アカウント (Builder ID) とワークフォース ID のセットアッププロセスについて説明します。個人の開発者であるか、組織で Amazon Q Developer をセットアップしている管理者であるかにかかわらず、このセクションは、Amazon Q をすぐに使用開始す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 のサービスティア – Free および Pro](#)
- [個人アカウント \(Builder ID\) を使用した開始方法](#)
- [IAM アイデンティティセンターを使用した開始方法](#)
- [Amazon Q Developer Pro サブスクリプション](#)
- [Kiro へのアップグレード](#)

Q Developer のサービスティア – Free および Pro

Amazon Q Developer では、Amazon Q Developer を Free ティアまたは Pro ティアで 사용합니다。以下の情報では、各ティアで何が提供されているかを説明しています。

Free ティア

Amazon Q Developer には、1 か月あたりの制限がある永久 Free ティアがあります。

Free ティアで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。

Free ティアは、[個人アカウント](#) (Builder ID) を持つユーザー、[IAM アイデンティティセンター](#) 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。サインイン方法に応じて、どのインターフェイスで Free ティアの Amazon Q を利用できるかを確認するには、次の表を参照してください。

インターフェイス	サインイン方法 (Free ティア)		
	個人アカウント (Builder ID)	IAM アイデンティティセンター	IAM 認証情報
AWS マネジメントコンソール、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
コマンドライン	 はい	 いいえ	 いいえ

Free ティア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、「[Amazon Q Developer のサービスの改善](#)」を参照してください。

Pro ティア

Pro ティアは Amazon Q Developer Pro と呼ばれ、Free ティア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、高度な機能にもアクセスできます。

Pro ティアの料金の詳細については、「[Amazon Q Developer の料金ページ](#)」を参照してください。

Pro ティアは、[個人アカウント](#) (Builder ID) を持つユーザー、および [IAM アイデンティティセンター ID](#) を持つユーザーが利用できます。サインイン方法に応じて、どのインターフェイスで Pro ティアの Amazon Q を利用できるかを確認するには、次の表を参照してください。

インターフェイス	サインイン方法 (Pro ティア)	
	個人アカウント (Builder ID)	IAM アイデンティティセンター
AWS マネジメントコンソール、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
コマンドライン	 はい	 はい

個人アカウント (Builder ID) を使用した開始方法

個人プロジェクトで 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、Builder ID とも呼ばれる個人アカウントで使用を開始することができます。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 マネジメントコンソール、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して始めることができます。準備ができたら、Builder ID を AWS アカウント に接続して [Pro ティアにアップグレード](#)し、より高い使用制限を活用できます。

Pro ティアで利用できる機能の一覧については、「[Amazon Q Developer の料金ページ](#)」を参照してください。

Free ティアまたは Pro ティアで個人アカウント (Builder ID) の使用を開始するには、次の手順に従います。

- [開始する前に: 個人アカウント \(Builder ID\) の制限を理解する](#)
- [ステップ 1: サインアップ](#)
- [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 ティアにアップグレードする](#)

開始する前に: 個人アカウント (Builder ID) の制限を理解する

Amazon Q で使用する個人アカウント (Builder ID) を作成する前に、その制限を理解します。

- Free ティアで Builder ID を使用する場合、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、「[料金ページ](#)」を参照してください。使用制限を引き上げる必要がある場合は、以下の手順に従って Builder ID を Pro ティアにサブスクライブするか、「[IAM アイデンティティセンターを使用した開始方法](#)」のガイダンスに従って IAM アイデンティティセンターを使用します。
- ビルダー ID が Pro 階層にある場合、使用制限は高くなりますが、すべての Pro 階層専用機能が提供されるわけではありません。利用できない Pro ティアの機能の一覧については、「[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 ティアの機能が必要な場合は、IAM アイデンティティセンターを使用します。詳細については、「[IAM アイデンティティセンターを使用した開始方法](#)」を参照してください。
- Free ティアおよび Pro ティアで Builder ID を使用する場合、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。[アプリ AWS マネジメントコンソール](#)、[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 マネジメントコンソールやウェブサイトで 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 アイデンティティセンターを使用した開始方法](#)」を参照してください。

ステップ 1: サインアップ

無料の個人アカウント (Builder ID) にサインアップします。E メールアドレスまたは既存の Google アカウントを使用してサインアップできます。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 (Builder 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](#)

- [Kiro CLI をインストールします。](#)

ステップ 3: (オプション) Pro ティアにアップグレードする

Pro ティアにアップグレードして、制限を引き上げます。「[個人アカウントのアップグレード \(Builder ID\)](#)」を参照してください。

IAM アイデンティティセンターを使用した開始方法

IAM アイデンティティセンター は、管理者がエンドユーザーの ID を管理するために使用するサービスです。Amazon Q Developer においては、管理者は IAM アイデンティティセンターを使用して、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、このガイドでは 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びます。

以下の場合には、IAM アイデンティティセンターで使用を開始する必要があります。

- Pro ティアの Amazon Q Developer で複数のユーザーを設定する管理者の場合。IAM アイデンティティセンターを使用することで、ユーザーは Amazon Q Developer 機能の完全なスイートを利用でき、管理者は Amazon Q Developer サブスクリプションをエンタープライズレベルで管理できます。例えば、ユーザーのサブスクリプションをキャンセルしたり、ユーザーを一括で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。
- [制限](#)により個人アカウント (Builder ID) を使用することができない個人ユーザーの場合。

IAM アイデンティティセンターで使用を開始するには、次の手順を実行します。

IAM アイデンティティセンターで開始する

- [ステップ 1: デプロイオプションを選択する](#)
- [ステップ 2: Amazon Q Developer Pro にワークフォースを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 1: どこで IAM アイデンティティセンターを有効にするか – IAM アイデンティティセンターの詳細については、「AWS IAM アイデンティティセンター ユーザーガイド」の「[What is IAM Identity Center?](#)」を参照してください。
- 決定 2: どこに Amazon Q Developer プロファイルを作成するか – 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。
- 決定 3: ワークフォースユーザーをどこにサブスクライブするか – サブスクリプションの詳細については、「[Amazon Q Developer Pro サブスクリプション](#)」を参照してください。

デプロイオプションは、これら 3 つの決定の組み合わせで構成されます。

次の表にデプロイオプションの説明を示します。[ステップ 2: Amazon Q Developer Pro にワークフォースをサブスクライブする](#)に進む前に、オプションを選択してください。

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	エンドユーザーが自分自身 (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での実装を行うことなく、ユーザー自身で Pro ティアの機能を試すことができます。 個人アカウント (Builder ID) よりも豊富な機能。詳細につ	機能が少ない。IAM アイデンティティセンター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、スタンダード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を有効にする • Amazon Q Developer プロファイルを作成する • 自分自身 (およびチームメンバー) をサブスクライブする 詳細な手順については、「スタンダードアカウントで 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。	いては、「Builder ID の制限」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、次の操作を行います。 - 管理アカウントで、IAM アイデンティティセンターを有効にします。 - メンバーアカウントで、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する 詳細な手順については、「メンバーアカウントで 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。	より多くの機能。IAM アイデンティティセンター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²。 管理の分散 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ます。これは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 20 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 20 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を有効にする • 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する 詳細な手順については、「メンバーアカウントで 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。	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したり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 ティア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。	より少ない機能。IAM アイデンティティセンター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を有効にする • 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 を共有する 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。	より多くの機能。IAM アイデンティティセンター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²。	ベストプラクティスに準拠していない。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、委任管理者がサポートされていない Amazon Q Developer の制限により、管理アカウントの管理者が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。これは、メンバーアカウントにタスクを委任するベストプラクティスに沿ったものではありません。

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 および [AWS アプリ AWS マネジメントコンソール](#)や[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 アイデンティティセンター 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 組織インスタンスは、アカウントインスタンスと比較して、すべての IAM アイデンティティセンター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一覧については、「AWS IAM アイデンティティセンター 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: Amazon Q Developer Pro にワークフォースを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](#)」の説明に従ってデプロイオプションを選択したら、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、[AWS Organizations](#) によって管理される組織の一部ではありません。

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、最後に「[どのようなリソースが作成されましたか?](#)」を読み、サブスクライブ時に自分の代わりにインストールおよび設定されたリソースを確認します。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するため、および 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を持っている。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 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設定されている。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 アイデンティティセンターのリージョン](#)。IAM アイデンティティセンターインスタンスをインストールし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インストールされます。IAM Identity Center インスタンスは、最初のユーザーをサブスクライブした AWS リージョン にインストールされます。IAM アイデンティティセンターの詳細については、「AWS IAM アイデンティティセンター 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 アイデンティティセンター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、自分自身をサブスクライブする

1. スタンドアロン AWS マネジメントコンソール 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 [前提条件](#) 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。
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、「[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。
4. [開始] ボタンを選択します。

 Note

[開始] ボタンの代わりに [設定] ボタンが表示される場合は、「開始」ワークフローをすでに実行しているため、[ステップ 2: チームメンバーをサブスクライブする](#) にスキップできることを意味します。

[ユーザーを作成] 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、AWS アカウントへのサインアップに使用したものと同じでも異なってもかまいません。

[続行] をクリックしてください。

[Amazon Q Developer プロファイルを作成] 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] に自分の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、「[Amazon Q Developer でのクロスリージョン処理](#)」を参照してください。ダッシュボードメトリクスを無効にするには、「[Amazon Q Developer ダッシュボードの無効化](#)」を参照してください。

[アプリケーションを作成] 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が [保留中] 状態で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] 状態に変わります。

サブスクライブされたので、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、今すぐ、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「AWS IAM アイデンティティセンターへの参加の招待」および「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 を設定します。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。

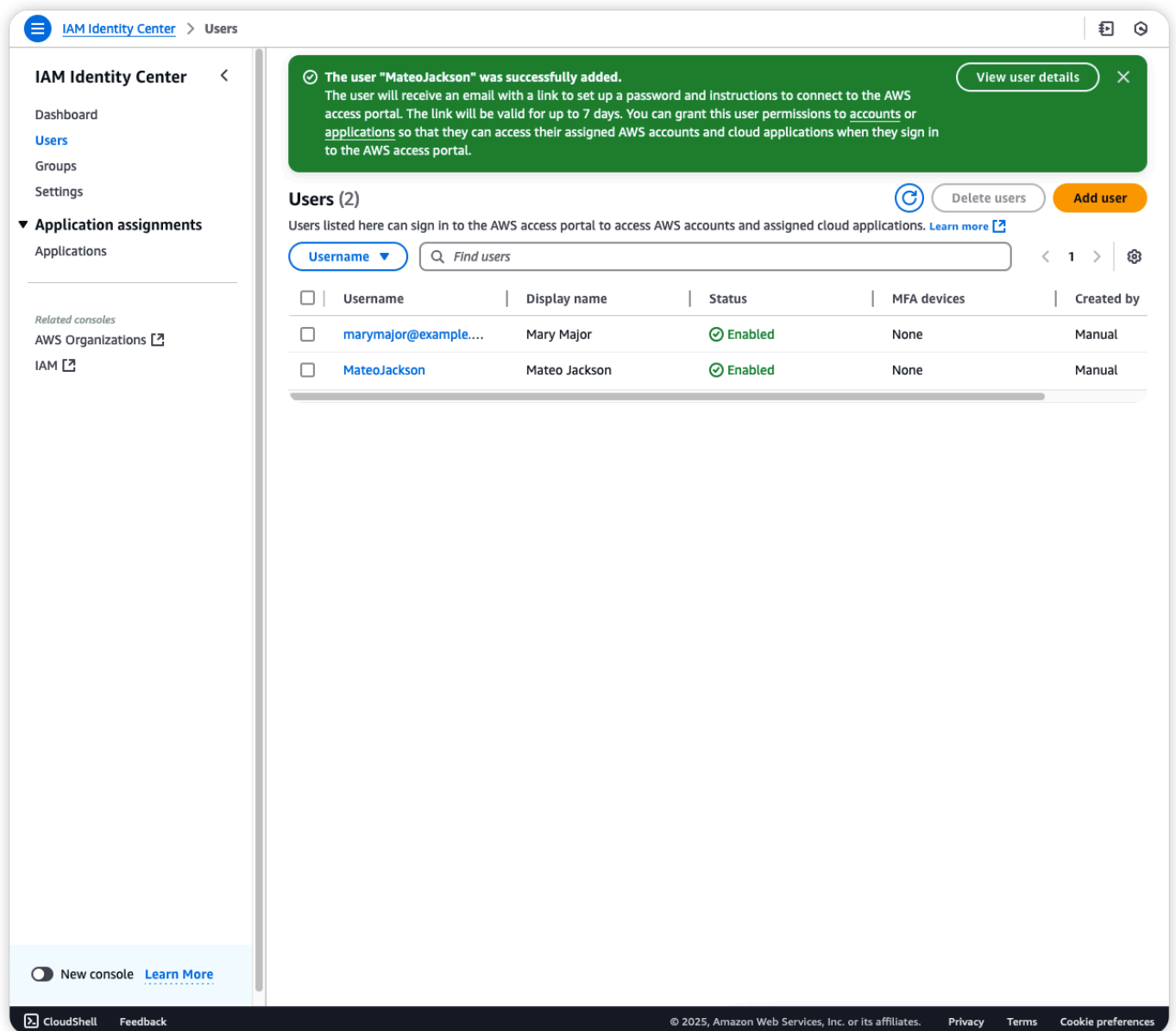
チームメンバーを追加するには

1. (IAM コンソールではなく) IAM アイデンティティセンターコンソールに切り替えます。

Note

IAM アイデンティティセンターは、自分自身をサブスクライブしたときに、自分の代わりに設定されています。設定された IAM アイデンティティセンターの詳細については、[「どのようなリソースが作成されましたか?」](#)を参照してください。

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 アイデンティティセンターユーザーガイド」の「[IAM アイデンティティセンター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. 次の手順に進み、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] を選択して、[サブスクリプションを作成] 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。

- 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。名前が自動的に入力されません。

 Note

ダイアログボックスは、ユーザー名またはグループ名のみと一致します。E メールアドレスとは一致しません。

- [割り当てる] を選択します。
- 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 マネジメントコンソール 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。この 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？

自分自身 (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 は自分に代わって次の AWS リソースを作成しました。

- 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。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の詳細については、「[AWS IAM アイデンティティセンター ユーザーガイド](#)」の「[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](#)」を参照してください。

 Note

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には[制限事項](#)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 のウェブサイトでは 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 アイデンティティセンター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 アイデンティティセンターの最初のユーザー 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、次の手順を使用して、自分のアカウントの 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、自分の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「[AWS Organizations の用語と概念](#)」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するため、および 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を持っている。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 アイデンティティセンターの組織インスタンスが設定されていること。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 アイデンティティセンターのリージョン](#)。IAM アイデンティティセンターの詳細については、「AWS IAM アイデンティティセンター 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 マネジメントコンソール 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、「[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。
4. [開始する] を選択します。

[Amazon Q Developer プロファイルを作成] 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] に自分の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、「[Amazon Q Developer でのクロスリージョン処理](#)」を参照してください。
 - [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「[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。
 - ダッシュボードメトリクスの無効化については、「[Amazon Q Developer ダッシュボードの無効化](#)」を参照してください。

[アプリケーションを作成] 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には、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で使用可能なものが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のみと一致します。E メールアドレスとは一致しません。

4. [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。24 時間以内に、Amazon Q Developer Pro ライセンスの使用を開始する方法については記載された、「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 マネジメントコンソール](#) 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 アイデンティティセンター ユーザーガイド」の「[ID 対応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 アプリ AWS マネジメントコンソール、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したときに、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 20 AWS アカウント 個まで作成できます。AWS リージョン

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

メンバーアカウントは AWS アカウント、管理アカウント以外のものであり、によって管理される組織の一部です [AWS Organizations](#)。

メンバーアカウントの所有者は、次の手順を使用して、自分のアカウントの 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と管理アカウントのどちらでユーザーをサブスクライブするかわからない場合ヘルプで [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、メンバーアカウント、および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「[AWS Organizations の用語と概念](#)」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するため、および 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を持っている。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 IAM アイデンティティセンターの組織インスタンスが設定されている、またはメンバーアカウントに 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設定されている。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 アイデンティティセンターのリージョン](#)。IAM アイデンティティセンターインスタンスをインストールし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インストールされます。IAM Identity Center インスタンスは、最初のユーザーをサブスクライブした n AWS リージョンにインストールされます。IAM アイデンティティセンターの詳細については、「AWS IAM アイデンティティセンター 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 マネジメントコンソール 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、「[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。
4. [開始] ボタンを選択します。

 Note

[開始] ボタンの代わりに [設定] ボタンが表示される場合は、「開始」ワークフローをすでに実行しているため、[ステップ 2: 他のユーザーをサブスクライブする](#) にスキップできることを意味します。

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 IAM アイデンティティセンターがまだ存在しない場合にのみ、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ます。
- 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が存在しますが、そのインスタンスにユーザーがない場合、Amazon Q は既存の IAM アイデンティティセンターにユーザーを作成します。

[Amazon Q Developer プロファイルを作成] 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] に自分の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、「[Amazon Q Developer でのクロスリージョン処理](#)」を参照してください。ダッシュボードメトリクスを無効にするには、「[Amazon Q Developer ダッシュボードの無効化](#)」を参照してください。

[アプリケーションを作成] 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、[保留中] 状態で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] 状態に変わります。

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 マネジメントコンソール 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。この 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、他のユーザーがまだ追加されていない場合は、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 [サブスクライブ] を選択して Amazon Q Developer Pro にサブスクライブします。

IAM アイデンティティセンターにユーザーを追加する手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の「[IAM アイデンティティセンター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 マネジメントコンソールやウェブサイト](#) 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 アイデンティティセンター ユーザーガイド」の「[ID 対応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、ユーザーは引き続き AWS マネジメントコンソール、AWS アプリ、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 マネジメントコンソール、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 アイデンティティセンター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 アイデンティティセンターのアカウントインスタンスの詳細については、AWS IAM アイデンティティセンター「ユーザーガイド」の「[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](#)」を参照してください。

 Note

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には[制限事項](#)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 アイデンティティセンター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 アイデンティティセンターの最初のユーザー。(チームメンバーも追加した可能性があります。)
- 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 20 AWS アカウント 個まで作成できます。AWS リージョン

Amazon Q Developer Pro サブスクリプション

Pro ティア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 Free ティアを超える使用制限を必要とする専門的な開発者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。料金ティア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 (Builder ID) 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の解除や Pro ティアへのアップグレードなどの主要なタスクに関する手順ごとの説明も提供します。

トピック

- [Amazon Q Developer Pro のリージョンサポート](#)
- [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 の表示](#)
- [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 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 のリージョンサポート

Pro ティアのリージョン情報は、個人アカウント (Builder ID) を持つエンドユーザーか、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の管理者かによって異なります。

個人アカウント (Builder ID) ユーザー

個人アカウント (Builder ID) の所有者については、Pro ティア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 アイデンティティセンターの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 アイデンティティセンターのリージョン](#)
- [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 アイデンティティセンターのリージョン

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、「[Supported Regions page](#)」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 マネジメントコンソール、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 アイデンティティセンターのワークフォースユーザーのサブスクリプションを IAM アイデンティティセンター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、「[Amazon Q Developer プロファイルとは](#)」を参照してください。

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 コンソールおよび 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 の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。

 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。

- [サポートとのチャット](#)
- [Amazon Q を使用したトラブルシューティングリソース](#)
- [IDE での .NET 変換](#)
- [チャットアプリケーションの Amazon Q \(Slack 用\)](#)
- の「Amazon Q AWS Console Mobile Application」
- [GitLab Duo with Amazon Q](#)
- [Amazon Q for GitHub](#)

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 アイデンティティセンターの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 アイデンティティセンターインスタンスの 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 アイデンティティセンターよりも少ない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場合は、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 アイデンティティセンターインスタンスは、米国西部 (北カリフォルニア)にある。
- Amazon Q Developer コンソールは、米国東部 (バージニア北部) にある。これは、Amazon Q Developer コンソールがサポートする IAM アイデンティティセンター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) に作成されます。
- 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、「[保管中の暗号化](#)」を参照してください。

サブスクリプションの詳細な手順については、「[IAM アイデンティティセンターを使用した開始方法](#)」を参照してください。

Amazon Q Developer Pro サブスクリプションの請求

Pro ティアの請求情報は、個人アカウント (Builder ID) を持つエンドユーザーか、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の管理者かによって異なります。

個人アカウント (Builder ID) ユーザー

個人アカウント (Builder ID) で Pro ティアにサブスクライブする場合は、毎月請求されます。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。例えば、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の課金されます。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 のサブスクリプション解除](#)」を参照してください。

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

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を Pro ティアにサブスクライブしている管理者は、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 の使用料は AWS 組織ごとに請求されます。請求書は、管理アカウントに送付されます。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 を設定していない場合は、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。

請求書は、請求情報とコスト管理コンソールで確認することができます。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、請求情報とコスト管理コンソールの[\[データエクスポート\]](#)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 User Guide」の「[Creating data exports](#)」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 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ステータス情報は、個人アカウント (Builder ID) を持つエンドユーザーか、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の管理者かによって異なります。

個人アカウント (Builder ID) ユーザー

個人アカウント (Builder 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確認できます。

サブスクリプションステータスは、以下のいずれかになります。

- アクティブ – Amazon Q Developer 機能を使用してサブスクリプションが有効化されています。サブスクリプションに対して請求が行われます。
- キャンセル済み – Pro ティアの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することはできません。詳細については、「[Amazon Q Developer Pro のサブスクリプション解除](#)」を参照してください。

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

一連の 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を Pro ティア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確認できます。

ステータスは、[グループ] タブと [ユーザー] タブで若干異なります。

[グループ] タブのステータスは次のとおりです。

- サブスクライブ中 – 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の解除) 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ません。詳細については、「[Amazon Q Developer Pro 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有効化しました。このサブスクリプションに対して請求が行われます。
- 保留中 – ユーザーはサブスクライブしていますが、サブスクリプションを有効化していません。このサブスクリプションに対しては請求は行われません。
- 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の解除) され、ユーザーは Amazon Q Developer 機能にアクセスできません。詳細については、「[Amazon Q Developer Pro 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しているユーザーは表示されません。これらのユーザーを表示するには、Amazon Q コンソール (Amazon Q Developer コンソールではなく) のサブスクリプションページに移動します。このページでは、グループで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 の表示

Note

このセクションは、個人アカウント (Builder ID) には適用されません。

一連の 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を Pro ティアにサブスクライブしている管理者は、それらのユーザーは IDE またはコマンドラインで IAM アイデンティティセンターの開始 URL とリージョンを使用して 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 は組織に固有です。

開始 URL を表示するには

1. AWS マネジメントコンソール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](#)」で定義された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。開始 URL は組織に固有です。

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 (Builder ID) には適用されません。

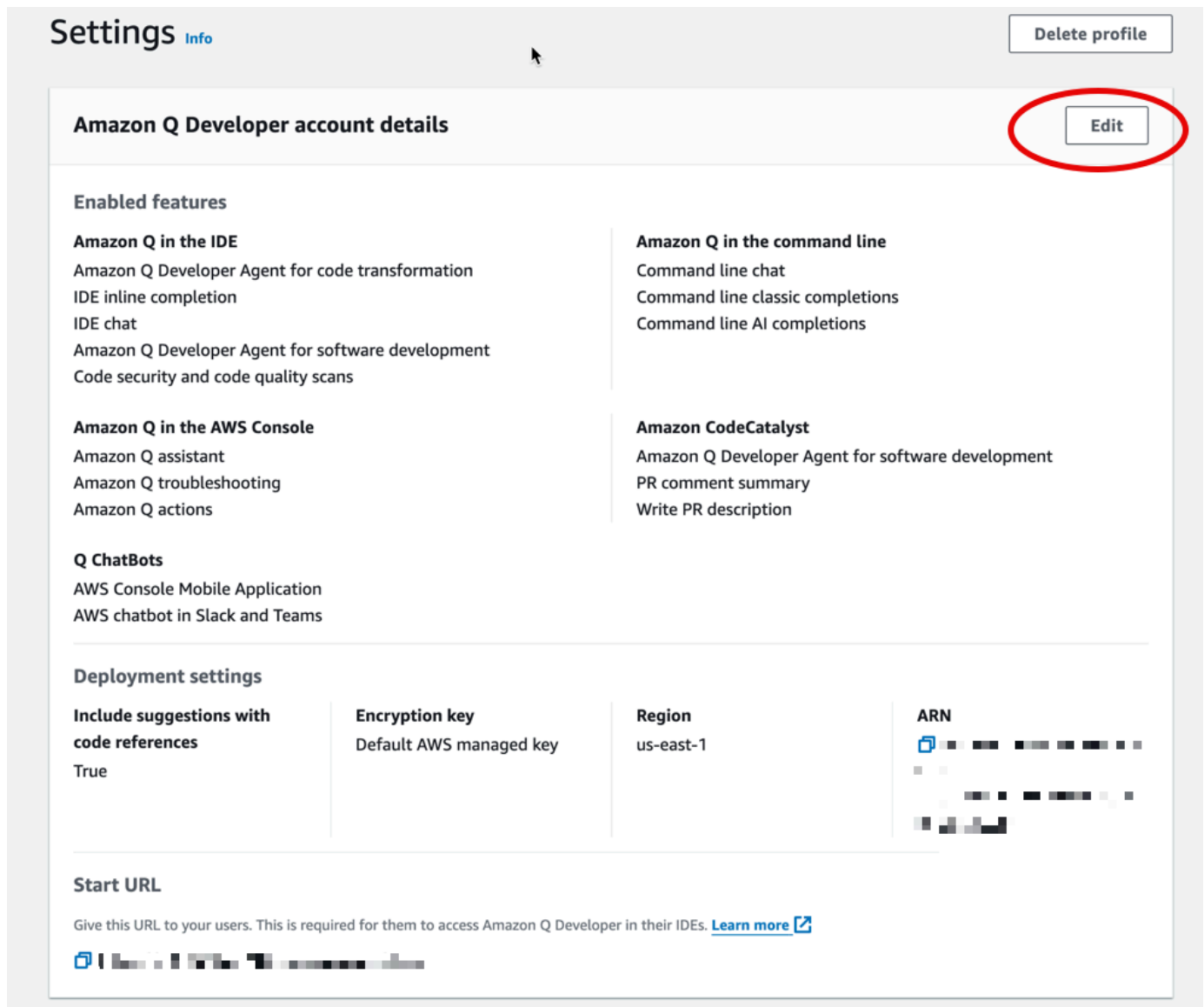
デフォルトでは、Amazon Q Developer は暗号化に AWS マネージドキーを使用します。一部の機能では、カスタマーマネージドキーを設定してデータを暗号化できます。カスタマーマネージドキーによる暗号化をサポートする機能のリストについては、「[Data encryption](#)」を参照してください。

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 マネジメントコンソール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](#)」で定義された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 (詳細) 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 (Builder ID) 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 アイデンティティセンターのワークフォースユーザーの Pro ティア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際、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。(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) アカウント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 アイデンティティセンターのワークフォースユーザーでの使用に必須です。プロファイルを使用せずに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 (Builder ID) 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 コンソールの設定のロックが解除され、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、「[Amazon Q Developer プロファイルとは](#)」を参照してください。

Amazon Q Developer プロファイルを作成するには

- 「[IAM アイデンティティセンターを使用した開始方法](#)」の指示に従ってユーザーをサブスクライブします。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 (Builder ID) 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ぐにキャンセルして、AWS アカウントからすべての Q Developer 設定を削除できます。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。例えば、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。
- (QDevProfile-**region** と呼ばれる) マネージドアプリケーションは、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 アイデンティティセンターインスタンスは、[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

誤ってプロフィールを削除した場合:

- プロフィールを再作成してから、ユーザーを再度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行った設定もリセットする必要があります。IAM アイデンティティセンターインスタンスは削除されないため、ユーザー ID を再作成する必要はありません。
- プロフィールを再作成した後、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。表示できるのは、新しいプロフィールの作成日以降のデータのみです。

Amazon Q Developer プロフィールを削除するには、次の手順に従います。

[開始する前に]

- プロフィー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 プロフィールを削除するには

1. AWS マネジメントコンソール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](#)」で定義されたアクセス許可が必要です。

3. [設定] を選択します。
4. ページの上部付近で、[プロフィールの削除] を選択します。

Amazon Q Developer でのプロフィール共有の有効化

Note

このセクションは、個人アカウント (Builder ID) には適用されません。

[AWS Organizations](#) によって管理される組織内の管理アカウントの管理者は、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 [AWS マネジメントコンソールに、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 マネジメントコンソール、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 アイデンティティセンターインスタンスが設定され、Amazon Q に接続されていること。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ること。
- Amazon Q Developer コンソールにアクセスするための最小アクセス許可を保有していること。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 マネジメントコンソール 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 マネジメントコンソール、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確認してください。

トピック

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

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。
- 自分の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 \(Builder ID\) を使用した開始方法](#)」、「[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「[メンバーアカウントで 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。

ユーザーがアクティベーション E メールを受信しない

問題: ユーザーがアクティベーション E メールを受信しない

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 アイデンティティセンター。
- 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 メールがないか、スパムフォルダや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 アイデンティティセンター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 アイデンティティセンター ユーザーガイド」の「[Add users to your IAM Identity Center directory](#)」を参照してください。

AWS ウェブサイトで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 AWS ウェブサイトでサブスクリプションを使用できない

[AWS マネジメントコンソールおよびアプリやウェブサイトでの Amazon Q の使用](#)しようとする AWS 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 アイデンティティセンター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の詳細については、「[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 アイデンティティセンター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 アイデンティティセンター 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 アイデンティティセンターのアカウントインスタンスの 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 マネジメントコンソールまたは他の 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。
- グループの一員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に最大 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、またはマネージドアプリケーションが削除されていないかを確認します。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。
- ユーザーにアクティブなサブスクリプションがない場合は、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。

ユーザーが IDE でサブスクリプションを使用できない

問題: IAM アイデンティティセンター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て再度サインインしてもらいます。
- グループの一員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に最大 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、またはマネージドアプリケーションが削除されていないかを確認します。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、ユーザーには、ビルダー ID でサインインして無料利用枠を使用してもらいます。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を表示できない

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ない。

Pro 階層に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] ページに移動してもそれらのユーザーが表示されません。

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 コンソールでは、グループの一員としてサブスクライブされたユーザーを表示でき、AWS Organizationsで管理される組織内の複数のアカウントのサブスクリプションを表示することもできます。
- 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ます AWS リージョン。IAM アイデンティティセンターインスタンスがデプロイされているリージョンにいまする必要があります。Amazon Q Developer コンソールとプロフィールでリージョンが異な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 (Builder ID) には適用されません。

[AWS Organizations](#) によって管理される組織内の管理アカウントの管理者は、管理アカウントへのサインイン中に、(Amazon Q Developer コンソールではなく) Amazon Q コンソール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 つの集約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

組織全体の可視性を有効にすると、集約されたサブスクリプション情報は、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の管理者は、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場合にも適用されます。

Amazon Q Developer サブスクリプションを組織全体で可視化するには、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実行でき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 User Guide」の「[Terminology and concepts for AWS Organizations](#)」を参照してください。

組織で Amazon Q への信頼されたアクセスを有効にするには、次の手順を実行します。

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 アイデンティティセンターの組織インスタンスを設定し、Amazon Q Developer に接続していること。
- IAM アイデンティティセンター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ること。
- 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ること (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 マネジメントコンソール 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、次のいずれかの操作を実行します。

- 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の表示設定] セクションで、[編集] を選択します。

[オン] を選択します。

- 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の表示設定] 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。

これで Amazon Q への信頼されたアクセスが有効になります。管理アカウントの管理者としてサインインすると、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が (Amazon Q Developer コンソールではなく) Amazon Q コンソールに表示されるようになります。

Amazon Q Developer Pro のサブスクリプション解除

個人アカウント (Builder ID) を持つエンドユーザーか、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の管理者かに応じて、Pro ティアのサブスクリプションを解除する方法について説明します。

個人アカウントのサブスクリプション解除 (Builder ID)

Pro ティアから個人アカウント (Builder 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、[Free ティア](#)の制限内で

Free ティアを使用できます)。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。

 Warning

Builder 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、能動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 マネジメントコンソール

IDE

IDE から Builder 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 (Builder 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 メニューで、[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 マネジメントコンソール を使用して にサインインします。この AWS アカウント は、Builder ID を Pro ティアにアップグレードしたときに指定したものです。詳細については、「[個人アカウントのアップグレード \(Builder ID\)](#)」を参照してください。

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

4. [Builder 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「q whoami」と入力して、個人アカウント (Builder ID) 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で「q chat」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. 「**/subscribe --manage**」と入力します。

はブラウザウィンドウで AWS マネジメントコンソール 起動します。

Note

が自動的に起動 AWS マネジメントコンソール 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 マネジメントコンソール を使用して にサインインします。この AWS アカウント は、Builder ID を Pro ティアにアップグレードしたときに指定したものです。詳細については、「[個人アカウントのアップグレード \(Builder ID\)](#)」を参照してください。

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

6. [Builder 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 マネジメントコンソール

から Builder ID のサブスクリプションを解除するには AWS マネジメントコンソール

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 マネジメントコンソール 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 \(Builder ID\)](#)」を参照してください。

2. 米国東部 (バージニア北部) リージョン (IAD) の 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. [Builder 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 アイデンティティセンターの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:

- [AWS Organizations](#) によって管理され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は、管理者がサブスクリプションを作成している場合にのみ、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で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そのユーザーのサブスクライブを解除する必要があります。
- 管理アカウントの管理者は、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、サブスクリプション解除についてメンバーアカウントの管理者と調整を行うことができます。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の管理者としてサインインし、ユーザーのサブスクリプションを直接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の管理者として表示する方法の詳細については、「[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。
- 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] としてマークされ、Amazon Q Developer の機能にアクセスできなくなります。(ただし、[Free ティア](#)の制限内で Free ティアを使用できます)。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。料金は日割り計算されません。

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 マネジメントコンソール 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

個人アカウント (Builder ID) を持つエンドユーザーが、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の管理者かに応じて、Free ティアから Pro ティアにアップグレードする方法について説明します。

個人アカウントのアップグレード (Builder ID)

個人アカウント (Builder ID) を Free ティアから Pro ティア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、「[サービスのティア](#)」および「[Builder ID の制限](#)」を参照してください。

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 Account Management Reference Guide」の「[Create an AWS アカウント](#)」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 (Builder 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。

または

- Amazon Q メニューで、[Q Developer Pro にサブスクライブする] 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 のアップグレードの詳細を確認するページが表示されます。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、「[サブスクリプションの請求](#)」を参照してください。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。このトークンが使用されるのは一度だけです。

4. [アップグレードを確認する] 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が成功しました」というメッセージが表示されます。[Builder ID ユーザー] 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 (Builder ID) が Pro ティアにサブスクライブされます。

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ました」という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「q whoami」と入力して、個人アカウント (Builder ID) 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で「q chat」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. 「**/subscribe**」と入力します。

はブラウザウィンドウで AWS マネジメントコンソール 起動します。

Note

が自動的に起動 AWS マネジメントコンソール 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 のアップグレードの詳細を確認するページが表示されます。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、「[サブスクリプションの請求](#)」を参照してください。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。このトークンが使用されるのは一度だけです。

6. [アップグレードを確認する] 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が成功しました」というメッセージが表示されます。[Builder ID ユーザー] 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 (Builder ID) が Pro ティアにサブスクライブされます。

7. (オプション) Q CLI プロンプトで「**/subscribe**」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 はサブスクライブ済みであることを示します。

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を Pro ティアにアップグレードする

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の管理者は、「[IAM アイデンティティセンターを使用した開始方法](#)」の手順に従ってユーザーを Pro ティアにアップグレードできます。

Kiro へのアップグレード

Amazon Q Developer CLI が Kiro にブランド変更されました。

このブランド変更とはどういう意味ですか？

- 個人アカウントを持つユーザーの場合: 必要に応じて、IDE の Q CLI と Amazon Q 拡張機能を Kiro にアップグレードして、今後 Kiro でのみ利用可能になるすべての更新を活用します。アップグレード時に、拡張機能と Q CLI のインストールが Kiro にブランド変更され、追加の機能も提供されます。
- Amazon Q Developer Pro サブスクリプションの管理者の場合: Kiro コンソールを使用してユーザーのサブスクリプションを管理します。Kiro コンソールは Amazon Q Developer コンソールのリブランドであり、すべて同じ機能を備えています。ユーザーに Amazon Q IDE 拡張機能と CLI インストールを Kiro にアップグレードして、リブランドされたインターフェイスにアクセスし、追加機能を提供してもらいます。アップグレード後、すべての Amazon Q コンポーネントは Kiro に相当するものに自動的に更新されます。

IDE およびコマンドラインでの Kiro と、エンタープライズで Kiro サブスクリプションを管理する方法については、[Kiro ドキュメント](#)を参照してください。

AWS アプリやウェブサイトでの Amazon Q Developer の使用

、AWS マネジメントコンソール、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 マネジメントコンソール、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 サポートとチャットする](#)

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 マネジメントコンソール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 アイデンティティセンターを使用してコンソールにサインインします。IAM アイデンティティセンター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 アイデンティティセンターを使用してサインインすると、IAM アイデンティティセンター ID が Amazon Q Developer Pro にサブスクライブされている場合、自動的に Pro 階層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 のサービスティア – Free および Pro](#)」を参照してください。

Note

「Your account has not been configured to use an Amazon Q subscription」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のヒントについて「[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](#)」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、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

生成 AI ベースの Q アーティファクトの紹介

Amazon Q は、テーブルとグラフの視覚化を使用して質問に対する回答を提供できるようになりました。プロンプトライブラリを使用すると、プロンプトの例を簡単に見つけることができます。Q エクスペリエンスがより使いやすく便利になりました。Q アイコンがナビゲーションバーに再配置されました。Q チャットパネルが左側に開くようになりました。

、AWS マネジメントコンソール、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 の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

- [Amazon Q での Q アーティファクトの使用](#)
- [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](#)
- [Amazon Q チャットパネルを操作する](#)
- [チャット設定](#)
- [プロンプトの例](#)
- [リソースに関する Amazon Q Developer とのチャット](#)
- [Amazon Q にリソースのトラブルシューティングを依頼する](#)
- [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](#)
- [E メール送信に関するチャット](#)
- [テレメトリと運用に関するチャット](#)

Amazon Q での Q アーティファクトの使用

Amazon Q アーティファクトを使用すると、Amazon Q はテーブルとチャートの視覚化で強化されたレスポンスを提供できます。リソースについて自然言語の質問をすると、Amazon Q には、リソースを一目で理解できるアーティファクトが表示されることがあります。

Q エクスペリエンスがより使いやすく便利になりました。検索の横にあるナビゲーションバーから Q に簡単にアクセスできます。Q チャットパネルが左側に開き、全画面表示に展開できます。新しいプロンプトライブラリは、便利なプロンプト例を見つけるのに役立ちます。

開始するには、必要なアクセス許可があることを確認し、プロンプトの例を確認して Amazon Q アーティファクトを最大限に活用します。詳細については、「[前提条件](#)」および「[プロンプトの例](#)」を参照してください。

前提条件

Amazon Q で視覚化を表示するには、「[Amazon Q とのチャットとコストに関するチャットをユーザーに許可する](#)」を参照してください。

仕組み

Note

Amazon Q ビジュアライゼーションに関連付けられたすべてのデータは us-east-1 に保存されます。

AWS マネジメントコンソールで Amazon Q の Q アーティファクトを表示するには:

1. AWS マネジメントコンソールにサインインします。
2. 統合ナビゲーションバーの Q アイコンを選択して Amazon Q にアクセスします。
3. 自然言語を使用してタスクを Amazon Q に記述します。例えば、次のようになります。
 - a. 「実行中の EC2 インスタンスを一覧表示する」
 - b. 「先月、リージョン別にコストのグラフを作成する」
4. Amazon Q がビジュアルインターフェイスが役立つと判断した場合、テーブルまたはチャートの視覚化を使用して、Q チャットの横の新しいパネルにアーティファクトが自動的に表示されます。
 - a. リソースについて質問する場合、パネルには以下が含まれます。
 - i. 指定したプロパティに基づいて分類された、質問したリソースを含むテーブル。
 - ii. サービスコンソールのリソースページにリダイレクトするリソースへのディープリンク。
 - b. グラフの視覚化でコストと請求情報について質問する場合、パネルにはグラフウィジェットが含まれます。

プロンプトの例

次のカテゴリと関連するプロンプトは、Amazon Q アーティファクトで完了できるタスクのタイプの例です。

- リソース情報の表示 – リソース情報をテーブルまたはグラフ形式で視覚化します。
- 請求のレコメンデーションと予測を取得する – 今後 6 か月間の予測コストの折れ線グラフを表示し、過去 6 か月間のインスタンスタイプごとに RDS コストをグラフ化します。
- セキュリティとコンプライアンス – EC2 リソースへのトラフィックとインターネットアクセスを確認し、リージョン間で EC2 インスタンスのインターネット接続を検証します。

推奨されるユースケースのリストについては、Q チャットパネルの右上にある Amazon Q プロンプトライブラリアイコンを選択し、テーブルまたはビジュアライゼーションレスポンスタイプでフィルタリングします。

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 マネジメントコンソール、統合ナビゲーションバーの左上にある 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. 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。
2. 会話に名前を付ける、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 で過去の会話を表示、切り替え、削除できます。

Amazon Q は、特定の会話内で以前に質問された質問と応答の履歴を保持し、応答を通知するためのコンテキストとして使用します。AWS コンソールで Amazon Q チャットとの最大 1,000 件の個別の会話を保存できます。

会話を開始すると、新しい会話として自動的に保存されます。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 は選択したプロンプトの例また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、以前のトピックに関する Amazon Q とのチャットを続行できます。新しい質問をしない非アクティブな会話は、90 日間非アクティブになると削除されます。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. チャットパネルの右上にある時計アイコンを選択します。会話ポップアップが開きます。
2. 再開する会話の名前を選択します。その会話からの以前のメッセージはすべてチャットパネルに表示され、Amazon Q とのチャットを続行できます。

会話を削除するには:

1. チャットパネルの右上にある時計アイコンを選択します。会話ポップアップが開きます。
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。

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。

Amazon Q チャットパネルを操作する

注: Amazon Q チャットパネルとサービスコンソールはいつでも切り替えることができます。

1. Q チャットパネルを全画面モードで展開するには、右上隅の最大化アイコンを選択します。全画面表示モードを切り替えるには、サイズ変更アイコンを選択します。
2. Q チャットパネルを閉じるには、右上隅で < を選択します。視覚化を使用してパネルを閉じるには、右上隅にある X を選択します。
3. チャットパネルのサイズを調整するには、デイバイダーを使用します。
4. チャットパネルを再度開くには、統合ナビゲーションバーの Q アイコンを選択します。
5. ビューを切り替えると、作業は自動的に保存されます。

チャット設定

Amazon Q でチャット設定を表示するには、チャットパネルの右上にある歯車アイコンを選択します。

- リージョン — チャットパネルを開く AWS マネジメントコンソール 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。

プロンプトの例

適切なサービスの検索 AWS のサービス、ベストプラクティスの理解、リソースの状態の確認など、AWS および について Amazon Q に質問できます。Amazon Q がビジュアルインターフェイスが役立つと判断した場合、テーブルまたはグラフの視覚化を含む新しいパネルが自動的に表示されます。

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。

- CloudWatch アラームなしで RDS データベースを一覧表示する
- Lambda 関数の最大ランタイムはどれくらいですか？
- VPC にリソースをいつ配置すべきですか？
- タグ値 `<tag value>` の S3 バケットを一覧表示する
- さまざまな S3 ストレージクラスの GB あたりのコストを示すグラフを作成する
- 過去 3 週間の vCPU 時間あたりの EC2 コストのグラフ化
- 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はどのコンテナサービスを使用するのが最適ですか？
- 最適化レコメンデーションによる削減額の棒グラフを表示する

開始しやすくするために、Q は新しい会話を開始するときにプロンプトを推奨します。プロンプトライブラリでサポートされているプロンプトのリストを表示することもできます。プロンプトライブラリでプロンプトを表示するには、チャットパネルの右上にある書籍アイコンを選択します。

リソースに関する Amazon Q Developer とのチャット

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。Amazon Q は、高度な推論機能を使用してリソースを分析し、インサイトを提供するため、複数のサービスコンソール、API、複雑なスクリプトに依存せずに、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。

- リソースの一覧表示と詳細 – アカウント内のリソースのリストまたは特定の詳細をリクエストします。
- 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、「[Amazon Q にリソースのトラブルシューティングを依頼する](#)」を参照してください。

可能な質問の例については、「[Amazon Q にリソース情報をリクエストする](#)」を参照してください。

トピック

- [仕組み](#)
- [前提条件](#)
- [Amazon Q にリソース情報をリクエストする](#)
- [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 API を呼び出せるようにするには、IAM アイデンティティにそれらの API 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 環境をより深く

理解す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。最初の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、読み取り専用 Q アーティファクトで強化された質問に対する回答を提供できます。例えば、リソースやコスト、請求について質問すると、Amazon Q はテーブルやグラフなどの視覚化を生成し、アカウントリソースの状態をすばやく理解するのに役立ちます。

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 マネジメントコンソール、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 アイデンティティ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 にリソースへのアクセスを許可するためのアクセス許可。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、「[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 に質問する AWS サービスの料金ページを参照してください。

Amazon Q にリソース情報をリクエストする

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。指定された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され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 は、リソースの質問に回答するために追加情報を必要とする場合があります。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?
- `<#####>` で自分の EC2 インスタンスを一覧表示する
- Lambda 関数 `<##>` の設定を取得する
- 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?
- CloudWatch アラームなしで RDS データベースを一覧表示する
- タグ値 `<tag value>` の S3 バケットを一覧表示する
- 先週のサービス別のコストのグラフを表示する
- 最も高価なリソースの上位 10 の棒グラフを表示する
- 予算と予測支出を示すグラフを作成する

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 リソースの数を教えてください。」など、リソース数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応答を生成するときに、その使用を試みます。Amazon Q は Resource Explorer を使用して、すべての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 API を呼び出してリソースを一覧表示し、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のリ

ソースをインデックス化するため、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リクエストすると、Amazon Q は Cloud Control API にフォールバックします。

Resource Explorer を有効にせず、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があります。また、多数のリソースをカウントする場合、Cloud Control API がタイムアウトする可能性もあります。

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「[詳細設定を使用して Resource Explorer をセットアップする](#)」を参照してください。
- ユーザーの IAM アイデンティティには、デフォルトビューに対する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「[検索用の Resource Explorer ビューへのアクセス許可の付与](#)」を参照してください。

Amazon Q にリソースのトラブルシューティングを依頼する

では AWS マネジメントコンソール、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。問題が発生したらチャットパネルを開き、Amazon Q に状況を説明します。例えば、「I can't add an object to my S3 bucket」(S3 バケットにオブジェクトを追加できない) や「My load balancer is returning a 503 error」(ロードバランサーからエラーが返される) と入力します。Amazon Q は、提供された情報を分析して、考えられる根本原因を特定します。次に、問題を効率的に解決するためのカスタマイズされたソリューション、詳細な手順、またはベストプラクティスを提示します。

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ます。

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
Amazon S3	アクセス許可の問題	Why can't I put objects into my S3 bucket? The bucket ID is amzn-s3-demo-bucket. (S3 バケットにオブジェクトを追加できないのはなぜですか? バケット ID は amzn-s3-demo-bucket です。) Why can't I delete the object s3://amzn-s3-demo-bucket-locked/Q-Stream2.jpg? (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?) Why can't I delete an object in S3? (S3 の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	My Glue job with the job name 'Run111B11B11-<...>' and the job run id 'bb_b1b111<...>' in the 'us-west-2' region failed. (ジョブ名 "Run111B11B11-<...>"、ジョブ実行 ID "bb_b1b111<...>" の "us-west-2" リージョンの Glue ジョブが失敗しました。) Why did my Glue job called GlueRun00AA00A00A-<...> fail? (Glue ジョブ GlueRun00

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
		AA00A00A-<...> が失敗したのはなぜですか?)
Amazon Athena	クエリの問題	My Athena query didn't return any results. query ID: 222c22cc-2c022-<...> region id: us-east-2 (Athena クエリから何も結果が返されません。クエリ ID: 222c22cc-2c022-<...>、リージョン ID: us-east-2) I ran an Athena query with an execution ID of 333d33dd-3d33-<...> and a region of us-east-1, and it didn't return any results. (実行 ID が 333d33dd-3d33-<...> でリージョンが us-east-1 の Athena クエリを実行しましたが、結果が返されませんでした。)

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
Amazon ECS	タスクが停止する問題、Fargate ヘルスチェックの問題、エージェントが切断される問題	My ECS task is stopped and I don't know why. The details of the task are: Cluster: my-ecs-cluster, Service: my-ecs-service, Task Definition: my-task-definition, Task ARN: arn:aws:ecs:us-west-2:444444444444:task/my-ecs-cluster/4ee4ee4ee4444<...> (ECS タスクが停止していますが、理由がわかりません。タスクの詳細は以下のとおりです。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/4ee4ee4ee4444<...>) I'm having a problem with my ECS task. The task health check always fails for the task in the 'my-ecs-cluster' cluster and service. (ECS タスクに問題が発生しています。"my-ecs-cluster" クラスターおよびサービス内のタスクでヘルスチェックが常に失敗します。) The Amazon ECS agent on one of my container instances appears to be disconnected. The agent is not responding

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
		or updating its status, which is causing tasks to be stuck in a pending state. (コンテナインスタンスの 1 つで Amazon ECS エージェントが切断されているようです。エージェントが応答せず、ステータスも更新されないため、タスクが保留状態のまま停止しています。)
Amazon EC2 Elastic Load Balancing	ヘルスチェックの問題、504、503、502、500 エラー	Why are the health checks for the target group called 'my-target-group' failing? (ターゲットグループ "my-target-group" のヘルスチェックが失敗するのはなぜですか?) Why am I receiving 503 errors from my load balancer 'my-elb'? (ロードバランサー "my-elb" で 503 エラーが発生しているのはなぜですか?)

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
Amazon EKS	Application Load Balancer (ALB) Ingress Controller の問題、マネージドアドオンの問題	I have an ALB ingress controller in my EKS cluster, and am seeing a failure with the error message 'WebIdentityErr:failed to retrieve credentials'. AWS リージョンは us-west-2 です。 There seems to be an issue with the add-ons in my EKS cluster called my-eks-cluster, in the us-west-2 region. (us-west-2 リージョンにある my-eks-cluster という EKS クラスターのアドオンに問題が発生しているようです。)

AWS サービス	Amazon Q が解決に役立つ問題の種類	プロンプトの例
Amazon ECR	セカンダリアカウントのアクセスの問題	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。Specifically, I need to allow account 2222222222 to push and pull images from the repository named "my-ecr-repo" in my account (111111111111) in the region (us-west-2). (別の AWS アカウントから Amazon ECR イメージリポジトリへアクセスするための許可を付与できません。具体的には、リージョン (us-west-2) にある私のアカウント (111111111111) の "my-ecr-repo" というリポジトリに対して、アカウント 222222222222 がイメージをプッシュおよびプルできるよう許可する必要があります。)

Amazon Q がリソースのトラブルシューティングを行うには、「[リソースに関する Amazon Q Developer とのチャット](#)」で説明されているものと同じアクセス許可が必要です。

コストに関するチャット

Amazon Q Developer は、AWS アプリケーションの理解、構築、拡張、運用に役立つ生成人工知能 (AI) を活用した会話アシスタントです。Amazon Q Developer は、自然な会話を通じて AWS コストを管理するのに役立つ強力な機能を提供します。Cost Explorer から履歴コストと予測コストを分析し、Cost Optimization Hub と AWS Compute Optimizer からコスト削減の推奨事項を発見し、Savings Plans と予約の機会を理解し、AWS の製品属性やサービスの料金についてすぐに回答

を得ることができます。Amazon Q Developer は、特定の質問 (「先月の EC2 インスタンスの純非ブレンドコストは何だったか」など) に回答することも、複雑な分析やオープンエンド分析 (先週のコスト削減の最大の要因は何だったか」など) を実行することもできます。Amazon Q Developer は、クエリ構文を学習したり、複数のコンソールページをナビゲートしたりするのではなく、独自の単語で質問をしたり、AWS アカウントからの実際のデータに裏付けられた正確な回答を提供したり、呼び出された APIs とコンソール内の情報の場所を正確に表示したりすることで、AWS コストデータとやり取りする方法を変換します。

Amazon Q Developer のコスト管理機能の詳細については、[「コスト管理ユーザーガイド」の「Amazon Q Developer で生成 AI を使用してコストを管理する」](#)を参照してください。AWS

できること

Amazon Q Developer では、次のことができます。

- コストを分析する – 過去の支出パターン、コスト傾向、予測コストについて質問します。たとえば、「先月の EC2 コストはいくらですか？」と入力します。または「先週コストが増加したのはなぜですか？」
- 最適化の機会を見つける – Cost Optimization Hub、AWS Compute Optimizer、Savings Plans のレコメンデーションについて質問することで、AWS の支出を削減する方法を見つけます。たとえば、「最高のコスト最適化の機会は何ですか？」と入力します。または「どの EC2 インスタンスがオーバプロビジョニングされていますか？」
- 料金を理解する – AWS サービスの料金に関する回答をすぐに取得できます。たとえば、「us-east-1 での c8g.2xlarge インスタンスのコストはいくらですか？」と入力します。または「ダブリンの S3 に 1 PB を格納するにはいくらかかりますか？」
- 支払いステータスの確認 — 最近の請求書を一覧表示し、支払い残高を確認します。たとえば、「過去 6 か月間の請求書を一覧表示する」と「未払いの支払い残高はありますか？」などです。
- コストを視覚化する – 過去のコストと使用状況、サービス料金、予算などのカスタムグラフとグラフを生成します。例えば、「各リージョンの支出額のグラフを表示する」や「先月の EC2-Other コストを分類するグラフを作成する」などです。

Amazon Q Developer は、質問のフレーズ方法に適応します。目的が正確にわかっている場合は、特定の質問をしたり、自由回答形式の探索的な質問をしたり、Q に調査を依頼したりできます。Q は会話全体でコンテキストを保持するため、フォローアップの質問をしてより深く掘り下げたり、分析を特定の方向に導いたりできます。

仕組み

コストについて Amazon Q Developer に尋ねると、Q は AWS Cost Explorer、Cost Optimization Hub、AWS Compute Optimizer、およびその他の AWS のサービスからデータを取得します。Q は計算を実行し、パターンを分析し、実際の使用状況と支出データに基づいてインサイトを提供します。Q は、応答ごとに、実行した API コール、使用されたパラメータ、および利用可能な場合は AWS マネジメントコンソールの一貫するビューへのリンクを表示することで、応答に到達した方法の透明性を提供します。これにより、データを検証し、さらに詳しく調べることができます。

開始方法

AWS のコストについてチャットするには、以下が必要です。

- 適切な IAM アクセス許可 – IAM ID には、Amazon Q とチャットし、請求データにアクセス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 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。
- Cost Explorer オプトイン – AWS アカウントで AWS Cost Explorer 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、[Cost Explorer の有効化](#)」を参照してください。AWS

Amazon Q Developer のコスト管理機能を最大限に活用するために、AWS Cost Optimization Hub や AWS Budgets などの追加サービスを有効にすることもできます。詳細については、「[コスト管理ユーザーガイド](#)」の「[Amazon Q Developer のコスト管理機能の概要](#)」を参照してください。AWS

開始するには、以下の手順を実行します。

1. <https://console.aws.amazon.com> で AWS マネジメントコンソールにサインインします。
2. コンソールナビゲーションバーの右側にある Amazon Q アイコンを選択します。
3. 次のようなコストについて質問します。
 - 「先月のコストはいくらですか？」
 - 「最高のコスト最適化の機会は何ですか？」
 - 「Linux を実行する c8g.2xlarge インスタンスは us-east-1 でいくらかかりますか？」
 - 「先週のコストの円グラフをリージョン別に表示」

Slack や Microsoft Teams などのチャットアプリケーションで Amazon Q Developer を設定することもできます。チャットアプリケーションで Amazon Q Developer を使用方法の詳細については、「」を参照してください[Amazon Q Developer in chat applications](#) [でのチャット](#)。

質問例

Amazon Q Developer に尋ねることができるコストに関する質問の例を次に示します。

コスト分析

- 「先月のコストはいくらですか？」
- 「過去 6 か月間の EC2 支出の傾向を表示します」
- 「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？」
- 「先週コストが増加したのはなぜですか？」
- 「先月の支出データを分析し、最も重要なインサイトをください」

コスト最適化

- 「最高のコスト最適化の機会は何ですか？」
- 「どの EC2 インスタンスがオーバースプロビジョニングされていますか？」
- 「アイドル状態のリソースはありますか？」
- 「どの Savings Plans を購入すればよいですか？」

予算と異常のモニタリング

- 「予算を超えたチームはありますか？」
- 「コストの異常はありますか？」

料金見積り

- 「us-east-1 での c8g.2xlarge インスタンスのコストはいくらですか？」
- 「ダブリンの S3 に 1 PB を格納するにはいくらかかりますか？」
- 「マルチ AZ および 300 GB gp2 ストレージを搭載した t4g.xlarge RDS インスタンスの月額コストはいくらですか？」

- 「小規模な EC2 インスタンス、API ゲートウェイ、最大 5 GB の SQL データベース、CloudFront でホストされている基本的な JS フロントエンドを備えた基本的な 3 層ウェブアプリを構築する料金はいくらですか？」

コストの可視化

- 「過去 12 か月間のサポート料金を月単位でグラフ化」
- 「今月の日別のインスタンスタイプ別の EC2 コストの面グラフを表示する」
- 「us-east-1 で階層別の S3 ストレージ料金のグラフを作成する」
- 「過去 3 か月間の Savings Plans カバレッジと使用率の折れ線グラフ」
- 「過去 3 週間の vCPU 時間あたりの EC2 コストをグラフ化」

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を分析し、AWS ネットワークセキュリティサービスの不足や誤設定を特定して、ネットワークセキュリティ体制を強化するための推奨事項を把握するのに役立ちます。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検出結果を理解し、修復手順を実施して、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 にたずねると、リソースに関する具体的な情報、関連するセキュリティ検出結果、詳細な修復手順、および AWS マネジメントコンソールで詳細を把握するためのリンクが応答として返されます。

Amazon Q による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「[Get insights with Amazon Q Developer](#)」を参照してください。

前提条件

AWS ネットワークセキュリティに関するチャットは、AWS マネジメントコンソールおよび[設定されたチャットアプリケーション](#)で行うことができます。

Amazon Q でネットワークセキュリティに関する質問の回答を得るには、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関するチャットを行うには、お使いの 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、「[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。

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 でネットワークセキュリティに関するチャットを行うには、お使いの AWS アカウントで 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必要があります。以下の手順に従って、AWS Shield ネットワークセキュリティディレクターを有効にします。

1. 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 (<https://console.aws.amazon.com/nsd/>) 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

以下は、ネットワークセキュリティについて Amazon Q にたずねることができる質問の例です。

- 上位のネットワークセキュリティ検出結果を特定して
- 環境内のネットワークセキュリティを要約して
- システムに DDoS 攻撃のリスクはありますか？
- ネットワークセキュリティを強化するにはどうすればよいですか？
- WAF で保護されていないリソースはありますか？
- 一般的なウェブの脆弱性から保護されていないリソースはどれですか？
- EC2 インスタンスに共通する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 WebACL はありますか？

E メール送信に関するチャット

Amazon Q は、Amazon Simple Email Service (Amazon SES) で E メール送信を設定するのに役立ちます。これにより、送信配信率とエンゲージメント率を最適化し、送信問題のトラブルシューティ

ングを行うことができます。Amazon SES について Amazon Q に尋ねると、そのレスポンスには、アカウントの送信 ID、設定セット、およびその他の Amazon SES リソースに関する情報が含まれます。また、E メール送信パターンや、Gmail や Yahoo などのメールボックスプロバイダーからの応答パターンに関する質問に回答することもできます。

前提条件

Amazon SES に関するチャットは、AWS マネジメントコンソール および [設定されたチャットアプリケーション](#)で行うことができます。

Amazon Q が E メール送信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

E メール送信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。また、質問する Amazon SES リソースにアクセスするためのアクセス許可も必要です。

質問例

Amazon Q に尋ねることができる E メール送信に関する質問の例を次に示します。

- E メール送信用の SES の設定を完了するには、何かする必要がありますか？
- 配信性能が最も高い送信 ID を教えてください。
- E メールの配信性能を Yahoo に送信する方法を教えてください。
- 送信を改善するための推奨事項はありますか？
- 配信性能が突然向上または悪化した最近のイベントがあったかどうか教えてください。

テレメトリと運用に関するチャット

Amazon Q は CloudWatch テレメトリデータと運用データを分析して、AWS 環境の管理を支援します。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。Amazon Q は質問されると、正確なサポートを提供するためにリソース名や時間範囲などの具体的な情報の入力を求めることがあります。

AWS サービスのヘルスチェック: 指定された AWS サービスのリソースの状態を評価し、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 から Y までの時間に DynamoDB テーブルのトラブルシューティングを支援してください。
- X から Y までの時間に Amazon S3 に関連する異常を調査してください。

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になるテレメトリを特定し、アラーム/アラート/ページの背後にある原因を診断できるようにします。

- X という名前のアラームが発生したのはなぜですか？

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、サービスの全体的な状態を判断して、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 テレメトリデータと運用データを分析する方法について詳しくは、「Amazon CloudWatch ユーザーガイド」の「[CloudWatch 調査](#)」を参照してください。

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。AWS マネジメントコンソールでは、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、脆弱性の理解と対応に役立てることができます。

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 API 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応答を生成します。

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 でのデータの使用方法の詳細については、「[データ保護](#)」を参照してください。

 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、「[あるプロバイダーのプラグインとユーザーが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 マネジメントコンソール。

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。応答を受け取ったら、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 CloudZero 間の接続を有効にします。プラグインを設定したら、Amazon Q チャットの質問の先頭に **@cloudzero** 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の設定](#)」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、「[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、「[管理者にプラグインの設定を許可する](#)」を参照してください。

認証情報の取得

開始する前に、CloudZero アカウントの次の情報をメモしてください。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 アカウント設定で確認できます。詳細については「CloudZero ドキュメント」の「[Authorization](#)」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
```

シークレットを作成する方法については、「AWS Secrets Manager User Guide」の「[Create a secret](#)」を参照してください。

サービスロール

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、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 AWS マネージド 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-1:111122223333: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-1:111122223333:secret:secret-id"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
                }
            }
        }
    ]
}

```

Amazon Q にサービスロールの引き受けを許可するには、サービスロールに以下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": "111122223333",
          "aws:SourceArn": "arn:aws:codewhisperer:us-east-1:111122223333:profile/profile-id"
        }
      }
    }
  ]
}
```

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、AWS Secrets Manager に保存されている認証情報を使用して、CloudZero とのやり取りを有効にします。

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。CloudZero 認証情報は、Secrets Manager シークレットに保存されます。

新しい設定を作成する場合は、以下の情報を入力します。

- a. [CloudZero API キー] で、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。作成し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がある場合は、[AWS Secrets Manager シークレット] ドロップダウンメニューからシークレットを選択します。シークレットには、前の手順で指定した 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

手順 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場合、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。作成し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、[サービスロール](#) 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。

7. [設定の保存] を選択します。
8. プラグインページの [設定済みプラグイン] セクションに CloudZero 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の設定

プラグインを使用するには、以下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、「[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。
- q:UsePlugin アクセス許可

IAM ID に設定済みの 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 を使用する場合で、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、プラグイン ARN を更新する必要があります。

CloudZero プラグイン ARN をを見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定済みの 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に CloudZero 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、「[あるプロバイダーのプラグインとユーザーがチャットできるようにする](#)」を参照してください。

CloudZero プラグインを使用してチャットする

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頭 **@cloudzero** に入力します。Amazon Q からのフォローアップの質問や質問への回答には、**@cloudzero** も含める必要があります。

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 は、最善の回答を提供するために、ユーザーが何をしようとしているかに関する詳細を求める場合があります。
 - **@cloudzero how do I use CloudZero?**
 - **@cloudzero how do I get started with CloudZero?**
- コストインサイトの一覧 — コストインサイトの一覧を取得したり、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 マネジメントコンソール。

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 Datadog 間の接続を有効にします。プラグインを設定したら、Amazon Q チャットの質問の先頭に **@datadog** 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の設定](#)」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、「[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、「[管理者にプラグインの設定を許可する](#)」を参照してください。

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ください。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 サイトの概要](#)」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 キー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、以下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-applicaition-key>"
}
```

シークレットを作成する方法については、「AWS Secrets Manager User Guide」の「[Create a secret](#)」を参照してください。

サービスロール

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 AWS マネージド 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-1:111122223333: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-1:111122223333:secret:secret-id"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
                }
            }
        }
    ]
}

```

Amazon Q にサービスロールの引き受けを許可するには、サービスロールに以下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": "111122223333",
          "aws:SourceArn": "arn:aws:codewhisperer:us-east-1:111122223333:profile/profile-id"
        }
      }
    }
  ]
}
```

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、AWS Secrets Manager に保存されている認証情報を使用して、Datadog とのやり取りを有効にします。

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] で、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。Datadog 認証情報は、Secrets Manager シークレットに保存されます。

新しい設定を作成する場合は、以下の情報を入力します。

- a. [Datadog API キー] で、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] で、Datadog アカount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。作成し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がある場合は、[AWS Secrets Manager シークレット] ドロップダウンメニューからシークレットを選択します。シークレットには、前の手順で指定した 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

手順 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場合、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。作成し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、[サービスロール](#) 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。

8. [設定の保存] を選択します。
9. プラグインページの [設定済みプラグイン] セクションに Datadog 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の設定

プラグインを使用するには、以下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、「[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。
- q:UsePlugin アクセス許可

IAM ID に設定済みの 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、プラグイン ARN を更新する必要があります。

Datadog プラグイン ARN をを見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定済みの 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に Datadog 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、「[あるプロバイダーのプラグインとユーザーがチャットできるようにする](#)」を参照してください。

Datadog プラグインを使用してチャットする

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、**@datadog** も含める必要があります。

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 は、最善の回答を提供するために、ユーザーが何をしようとしているかに関する詳細を求める場合があります。
- **@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「[モニター ステータス ページ \(レガシー\)](#)」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。リストに記載されているモニターに関するフォローアップの質問をすることができます。
- **@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 マネジメントコンソール。

プラグインを使用して、Wiz の問題の特定と取得、最もリスクの高いアセットの評価、脆弱性や漏洩の把握を行うことができます。応答を受け取ったら、問題に対処する方法など、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 Wiz 間の接続を有効にします。プラグインを設定したら、Amazon Q チャットの質問の先頭に `@wiz` 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の設定](#)」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、「[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、「[管理者にプラグインの設定を許可する](#)」を参照してください。

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ください。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 – Wiz にアクセスする URL。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 endpoint 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 API 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「[Client ID and Client secret](#)」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、以下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>"
}
```

シークレットを作成する方法については、「AWS Secrets Manager User Guide」の「[Create a secret](#)」を参照してください。

サービスロール

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、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 AWS マネージド 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-1:111122223333: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-1:111122223333:secret:secret-id"
    },
    {
```

```

        "Effect": "Allow",
        "Action": [
            "kms:Decrypt"
        ],
        "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id",
        "Condition": {
            "StringEquals": {
                "kms:ViaService": "secretsmanager.us-east-1.amazonaws.com"
            }
        }
    }
}
]
}

```

Amazon Q にサービスロールの引き受けを許可するには、サービスロールに以下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": "111122223333",
          "aws:SourceArn": "arn:aws:codewhisperer:us-east-1:111122223333:profile/profile-id"
        }
      }
    }
  ]
}

```

```
}  
}  
}  
]  
}
```

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、AWS Secrets Manager に保存されている認証情報を使用して、Wiz とのやり取りを有効にします。

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] で、Wiz にアクセスする API エンドポイントの 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。Wiz 認証情報は、Secrets Manager シークレットに保存されます。

新しい設定を作成する場合は、以下の情報を入力します。

- a. [クライアント ID] で、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] で、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。作成し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がある場合は、[AWS Secrets Manager シークレット] ドロップダウンメニューからシークレットを選択します。シークレットには、前の手順で指定した 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

手順 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場合、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。作成し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、[サービスロール](#) 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。

8. [設定の保存] を選択します。
9. プラグインページの [設定済みプラグイン] セクションに Wiz 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の設定

プラグインを使用するには、以下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、「[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。
- q:UsePlugin アクセス許可

IAM ID に設定済みの 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 を使用する場合で、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、プラグイン ARN を更新する必要があります。

Wiz プラグイン ARN をを見つけるには、Amazon Q Developer コンソールのプラグインページに移動し、設定済みの 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に Wiz 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、「[あるプロバイダーのプラグインとユーザーがチャットできるようにする](#)」を参照してください。

Wiz プラグインを使用してチャットする

Amazon Q の Wiz プラグインを使用するには、Wiz 問題に関する質問の冒頭に「@Wiz」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、@Wiz も含める必要があります。

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 の 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。また、最も重大な問題の上位 10 件までを一覧表示するよう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 – 問題の中でセキュリティの脅威をもたらしている脆弱性や漏洩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 の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 はコンソールアクションを記録し、生成 AI を使用して同等の AWS CLI コマンドとコードを任意の言語と形式で提案します。

サービスのティア

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 マネジメントコンソール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、2 つのサブネット (パブリック 1 つとプライベート 1 つ)、セキュリティグループ、NACL、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、AWS のさまざまなサービスを使用している場合でも、Console-to-Code ではこうしたアクションを 1 回の記録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、以下のサービスで AWS マネジメントコンソールを使用する場合にアクションを記録できます。

- Amazon DynamoDB
- AWS IoT
- Amazon Cognito
- 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 マネジメントコンソール 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. 次のいずれかを行います。

- 個々の CLI コマンドをコピーするには、コマンドの左側にあるコピーボタンを選択します。
- 個々の CLI コマンドを実行するには AWS CloudShell、コマンドの左側  にある CloudShell アイコンを選択します。これにより CloudShell が開き、すぐに実行できる CLI コマンドが入力されます。
- 一連の CLI コマンドを表示または実行するには、コマンドを選択し、選択したすべてのコマンドをコピーするには CLI をコピーするか、CloudShell を開いてすべてのコマンドを入力します。

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 マネジメントコンソール、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についてトラブルシューティングを行います。

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

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 サポートとチャットする](#)」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 マネジメントコンソール、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. ウィンドウが表示され、エラーに関する最初の情報が提供されます。次に、エラーを解決するために実行できる一連の手順が提供されます。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 サポートとチャットする

Amazon Q Developer を使用して、AWS Support Center Consoleなど AWS マネジメントコンソールの任意の場所からサポートケースを作成して サポートに問い合わせを行うことができます。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。

利用できる サポートのタイプは、AWS アカountのサポートレベルに応じて異なります。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以外のユーザー

のみが Amazon Q で サポートに問い合わせることができます。AWS サポートの詳細については、「AWS サポートユーザーガイド」の「[Getting started with 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、「[Amazon Q にリソースのトラブルシューティングを依頼する](#)」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。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 を使用して サポートに問い合わせることができます。
- 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許可があること。詳細については、「[Manage access to サポート Center](#)」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 AWS のサービスの一部である Amazon Q 機能について サポートに問い合わせるには、Amazon Q ではなく、他の AWS のサービスのサポートケースを作成します。例えば、Amazon VPC Reachability Analyzer で Amazon Q のネットワーク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に Amazon VPC を選択します。

Amazon Q Developer または Amazon Q Business の機能について サポート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、次のステップを使用します。

1. 次の 2 つの方法からいずれかを選択して Amazon Q で サポートケースを作成できます。
 - 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。サポートに問い合わせるには、[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。

[\[3\] Troubleshoot instances with failed status checks - Amazon Elastic Compute Cloud](#) 



Thank you for your feedback. If you need further assistance related to this issue, you may contact support.

[Create a support case](#)

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。

Sure, I've drafted the following support case for you. Review details and make required changes before continuing. I will also add our recent conversation to the case description once submitted.

Support Level	Enterprise Support	Change 
Description	The customer is unable to connect to their instance after a recent deployment. They have tried restarting the instance and generating new key pairs but are still unable to connect. No other changes were made to the network configuration. The issue is problematic and the customer would like to chat with support.	
Case type	Technical	
Service	Elastic Compute Cloud (EC2 - Linux)	
Category	Instance Issue	
Severity	General guidance	
Additional Contacts	None	
Attachments	 Attach Files You can attach up to 3 files. Each file can be up to 5 MB.	
Cancel Edit Submit		

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. サポートに問い合わせるには、使用する手段を選択します。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場合、サポートエージェントはお客様の AWS アカウントに関連付けられている E メールアドレス宛にご連絡します。
 - 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、コードの生成と更新、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で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](#)
- [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	はい	はい	あり
エージェントコーディング	はい	はい	はい	あり
MCP サーバー	はい	はい	はい	あり
チャットのコンテキスト	はい	はい	はい	あり

機能	VSCode	JetBrains	Eclipse	Visual Studio
インラインチャット	はい	はい	あり	いいえ
チャットのワークスペースコンテキスト	はい	はい	はい	あり
インライン提案	はい	はい	はい	あり
変換	はい	あり	なし	はい
	はい	はい	あり	いいえ

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。

- [Eclipse 用の Amazon Q をダウンロードする](#)
- [Visual Studio Code 用の Amazon Q をダウンロードする](#)
- [JetBrains IDE 用の Amazon Q 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 アイデンティティセンター ユーザーガイド」の「[How to extend the session duration for Amazon Q in the IDE](#)」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 でサポートされている Eclipse の最小バージョンは 2024-06 (4.32) です。
- Amazon Q でサポートされている JetBrains IDEs (IntelliJおよびを含むPyCharm) の最小バージョンは 2024.3 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. [Amazon Q プラグイン](#)を Eclipse にインストールします。

2. IDE の右上隅にある Amazon Q アイコンを選択します。
3. IDE の下部で 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、Eclipse IDE に戻ります。
5. Amazon Q を使用し始めるには、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。

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. [Amazon Q プラグイン](#) を Eclipse にインストールします。
2. IDE の右上隅にある Amazon Q アイコンを選択します。
3. IDE の下部で 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、Eclipse IDE に戻ります。
8. Amazon Q を使用し始めるには、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。

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. [Amazon Q プラグイン](#) を JetBrains IDE にインストールします。
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. [Amazon Q プラグイン](#) を JetBrains IDE にインストールします。
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 アイデンティティセンターで認証を行い、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 の下部で [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で、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. VS Code に [Amazon Q 拡張機能](#) をインストールします。
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. VS Code に [Amazon Q 拡張機能](#) をインストールします。
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 アイデンティティセンターで認証を行い、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 の下部で 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 のメインメニューから [拡張機能] を展開し、[AWS Toolkit] 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 [有効にする] を選択します。
4. [Free ティア] 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 のメインメニューから [拡張機能] を展開し、[AWS Toolkit] 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 [有効にする] を選択します。認証するには、Professional ティア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 アイデンティティセンターの詳細については、「IAM アイデンティティセンターユーザーガイド」の「[What is IAM Identity Center?](#)」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 ティア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、[接続] 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 ティア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、認証に使用する IAM アイデンティティセンタープロファイルの名前を入力します。

- b. [開始 URL] テキストフィールドに、IAM アイデンティティセンターの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、Connect 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、前の手順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 の下部で [Amazon Q] を選択し、[Q Developer 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で、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 プロファイルの変更] 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

- [IDE で Amazon Q の操作](#)
- [タスクの例](#)
- [質問例](#)
- [Amazon Q からの応答に関する問題の報告](#)
- [Amazon Q Developer でのコードレビュー](#)
- [Amazon Q Developer を使用した IDE でのコード変換](#)
- [Amazon Q Developer でのコードの説明と更新](#)
- [Amazon Q Developer とのインラインでのチャット](#)
- [IDE 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](#)
- [IDE での Amazon Q チャットのモデルの選択](#)

IDE で Amazon Q の操作

チャットの使用

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で 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。プロンプトにコンテキストを追加したり、チャットセッション全体でファイル、フォルダ、またはワークスペース全体をコンテキストとして指定したりできます。詳細については、「[チャットへのコンテキストの追加](#)」を参照してください。

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。

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

次のコマンドは、Amazon Q とのチャットを管理するのに役立ちます。

- /clear - このコマンドを使用して、現在の会話をクリアします。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つつ、会話の簡潔な概要が作成されます。
- /help - このコマンドを使用して、Amazon Q ができることとできないこと、質問例、使用可能な機能の概要を確認します。

エージェントコーディング

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、開発時にチャットします。エージェントコーディングは、IDE でデフォルトでオンになっています。

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。

Amazon Q にコードの改善を依頼すると、ファイルは直接更新されます。差分で変更を表示し、元に戻すオプションがあります。

Amazon Q がタスクを検討または作業している間、チャットパネルに指示を引き続き追加し、作業に組み込むことができます。

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、単独で実行します。

自然言語でのチャット

Amazon Q Developer は、IDE でチャットするときの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、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 IDE で Amazon Q との会話を開始します。Amazon Q が言語を自動的に検出し、適切な言語で応答を返します。

タスクの例

コード機能の開発

Note

この機能は、このドキュメントおよび IDE では `/dev` と呼ばれていました。

Amazon Q は、統合開発環境 (IDE) でのコード機能の開発、プロジェクトへのコード変更、ソフトウェア開発タスクに関する質問に回答するのに役立ちます。実行するタスクを説明すると、Amazon Q は現在のプロジェクトまたはワークスペースのコンテキストを使用して、コードベースに適用できる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。

ユニットテストの生成

Note

この機能は、このドキュメントおよび IDE では `/test` と呼ばれていました。

Amazon Q はユニットテストを生成できるため、ソフトウェア開発ライフサイクル全体でテストを自動化できます。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。

ドキュメントの生成

Note

この機能は、このドキュメントおよび IDE では /doc と呼ばれていました。

Amazon Q は、コードの READMEs やその他のドキュメントを生成して、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。新しいドキュメントを作成し、コードベースで既存のドキュメントを更新できます。

コードレビュー

Note

この機能は、このドキュメントおよび IDE では /review と呼ばれていました。

Amazon Q 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。この機能の使用方法的詳細については、「」を参照してください [Amazon Q Developer でのコードレビュー](#)。

コードの変換

Amazon Q 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 によって変更が生成され、ユーザーはその変更を確認してファイルに適用できます。詳細については、「[コードの変換](#)」を参照してください。

質問例

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、次のサブジェクト領域に関連する質問に回答するために特に役立ちます。

- 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 でのコードレビュー

Amazon Q Developer では、開発サイクル全体を通じて、コードベースのセキュリティ上の脆弱性やコード品質の問題をレビューし、アプリケーションのセキュリティを改善できます。コードベース全体を確認したり、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、単一のファイルを確認したりできます。コードを記述するときに評価する自動レビューを有効にすることもできます。

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 デイテクターは](#)、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、デイテクターが追加されると、コードが最新のポリシーに準拠するよう、レビューは自動的に新しいデイテクターを組み込みます。

この機能でサポートされている IDE については、「[サポートされている IDE](#)」を参照してください。サポートされている言語については、「[コードレビューの言語サポート](#)」を参照してください。

トピック

- [仕組み](#)
- [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 を使用したコードの問題への対処](#)
- [コードの問題をフィルタリングする](#)
- [Amazon Q Developer のコードレビューでのコードの問題の重要度](#)

仕組み

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、最近のコード変更を確認するか、ファイルまたはプロジェクト全体を確認できます。レビューを開始するには、IDE でコードフォルダを開き、チャットパネルからコードを確認するように Amazon Q に依頼します。

デフォルトでは、コードの確認を Amazon Q に依頼するだけで、IDE のアクティブなファイル内のコード変更のみがレビューされます。コードの変更は、ファイルの `git diff` コマンドの出力によって決まります。差分ファイルが存在しない場合、Amazon Q はコードファイル全体を確認します。ファイルが開いていない場合、レビューするプロジェクト内のコード変更を検索します。

同様に、プロジェクトまたはワークスペース全体を確認するように Amazon Q に依頼すると、まずコードの変更を確認しようとします。差分ファイルが存在しない場合は、コードベース全体を確認します。

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レビューします。

- **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。

- シークレットの検知 — コードでの機密情報や秘密情報の漏洩を防ぎます。Amazon Q は、コードファイルとテキストファイルをレビューして、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検出します。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 の問題スキャン — 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) コードファイルをレビューして、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、効率性の基準を満たしているかどうかを確認します。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の問題を表示します。
- 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 がコードをレビューするために使用するディテクターの一覧については、「[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

[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
ソースコードのサイズ	200 KB	50MB

Amazon Q Developer でコードレビューを開始する

Amazon Q は、ファイルまたはコードベース全体を確認したり、記述時にコードを自動的に確認したりできます。

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

- [ファイル、プロジェクト、またはワークスペースを確認する](#)
- [タスクとプロンプトの例](#)
- [コードを書く際にレビューする](#)

ファイル、プロジェクト、またはワークスペースを確認する

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を確認させることができます。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。

Amazon Q がレビューを完了したら、問題を調査して、問題を修正するためのコード修正を取得できます。詳細については、「[コードの問題の対処](#)」を参照してください。

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、IDE の次の手順を実行します。

JetBrains

1. IDE でレビューするファイルまたはプロジェクトを開きます。
2. Amazon Q アイコンをクリックして、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。該当する場合、Amazon Q は特に指定がない限り、デフォルトでのみコードの変更を確認します。

4. 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、以下を入力できます。
 - a. **Review my code changes** – Amazon Q はコードベースのコード変更を確認します
 - b. **Run a code review on this entire file** – Amazon Q は変更だけでなく、ファイル内のすべてのコードを確認します。
 - c. **Review this repository** – Amazon Q は変更だけでなく、コードベース全体を確認します。

コードレビューシナリオおよび関連するプロンプトの詳細については、[「プロンプトの例」](#)を参照してください。

5. Amazon Q はファイルまたはプロジェクトのレビューを開始します。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。
6. 問題が検出されると、コードの問題タブが開き、Amazon Q が検出した問題のリストが表示されます。
7. コードの問題の詳細については、コードの問題パネルに移動します。移動すると、以下の操作を実行できます。
 - a. 検出結果を選択すると、脆弱性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ます。
 - b. コードの問題の説明を表示するには、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正を提案します。
 - 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。ファイル内にコードの変更が表示され、チャットパネルには変更を元に戻すオプションが表示されます。
 - d. 自然言語を使用して、問題の詳細を尋ねたり、提案された修正の説明を得たり、代替の解決策を尋ねたりすることもできます。
8. コードの問題の対処については、[「Amazon Q Developer を使用したコードの問題への対処」](#)を参照してください。

Visual Studio Code

1. IDE でレビューするファイルまたはプロジェクトを開きます。
2. Amazon Q アイコンをクリックして、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。該当する場合、Amazon Q は特に指定がない限り、デフォルトでのみコードの変更を確認します。
4. 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、以下を入力できます。
 - a. **Review my code changes** – Amazon Q はコードベースのコード変更を確認します
 - b. **Run a code review on this entire file** – Amazon Q は変更だけでなく、ファイル内のすべてのコードを確認します。
 - c. **Review this repository** – Amazon Q は変更だけでなく、コードベース全体を確認します。

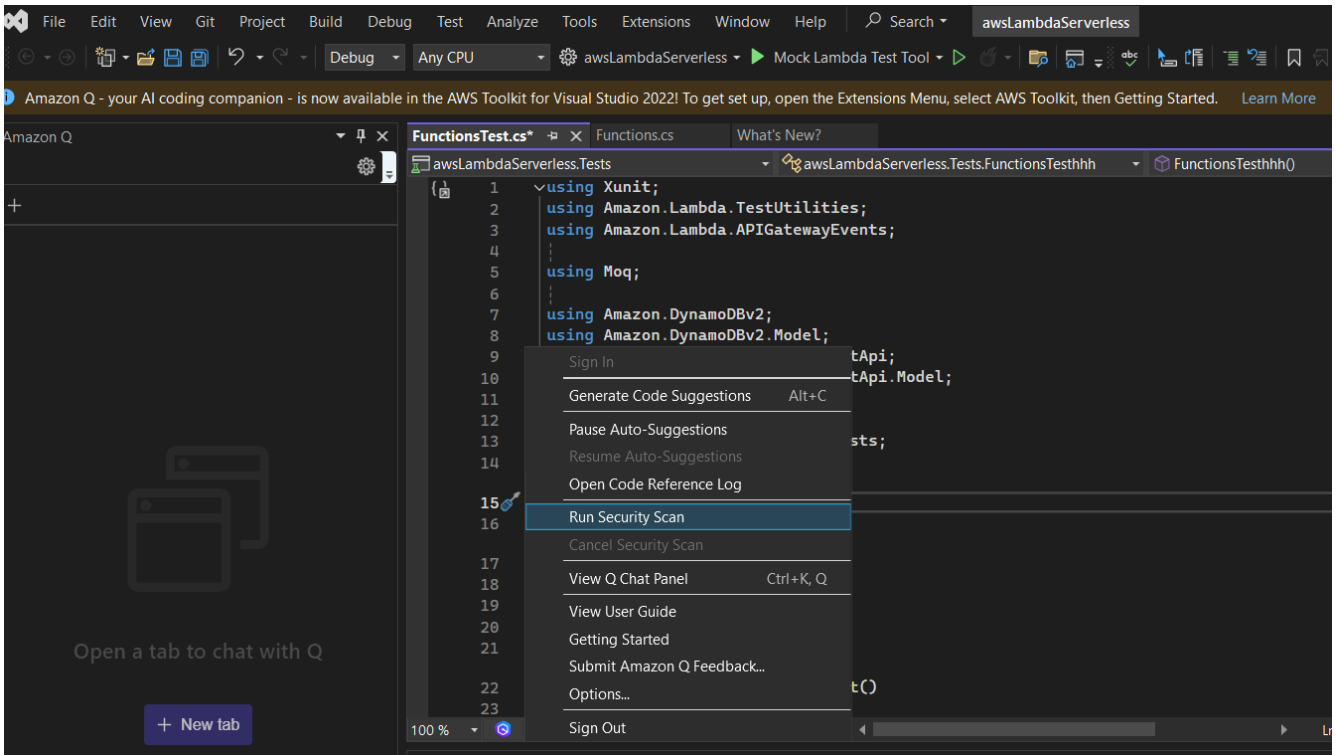
コードレビューシナリオおよび関連するプロンプトの詳細については、[「プロンプトの例」](#)を参照してください。

5. Amazon Q はファイルまたはプロジェクトのレビューを開始します。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。
6. 問題が検出されると、コードの問題タブが開き、Amazon Q が検出した問題のリストが表示されます。
7. コードの問題の詳細については、コードの問題パネルに移動します。移動すると、以下の操作を実行できます。
 - a. 検出結果を選択すると、脆弱性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ます。
 - b. コードの問題の説明を表示するには、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正を提案します。
 - 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。ファイル内にコードの変更が表示され、チャットパネルには変更を元に戻すオプションが表示されます。
 - d. 自然言語を使用して、問題の詳細を尋ねたり、提案された修正の説明を得たり、代替の解決策を尋ねたりすることもできます。
8. コードの問題の対処については、[「Amazon Q Developer を使用したコードの問題への対処」](#)を参照してください。

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 を使用したコードの問題への対処](#)」を参照してください。

タスクとプロンプトの例

コードレビューを開始するときは、いくつかのシナリオが考えられます。以下は、コードレビューを開始する方法と、必要なレビューを実行するように Amazon Q に求める方法の概要です。

- 1 つのファイルのコード変更のみを確認するには:

- IDE で ファイルを開き、**Review my code**
- **Review the code in <filename>** と入力します。
- コードファイル全体を確認するには:
 - 変更せずにファイルを開き、「」と入力します。**Review my code**
 - 変更を含むファイルを開き、「」と入力します。**Review my entire code file**
 - **Review all the code in <filename>** と入力します。
- リポジトリ内のすべてのコード変更を確認するには:
 - IDE でリポジトリを開き、**Review my code**
- 変更だけでなく、リポジトリ全体を確認するには:
 - IDE でリポジトリを開き、**Review my repository**

コードを書く際にレビューする

 Note

Amazon Q の自動レビューは、[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 デイテクター](#)によるルールベースのレビューです。Amazon Q は、ユーザーが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レビューし、問題が検出されるとすぐにコードの問題を表示します。Amazon Q が自動レビューを実行する際、インプレースコード修正は生成されません。

Amazon Q を使用する際、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、次の手順を実行します。

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 ウィンドウの下部で、[Amazon Q] を選択します。

Amazon Q タスクバーが開きます。

2. [自動レビューを一時停止] 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。

Amazon Q Developer を使用したコードの問題への対処

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 Visual Studio Code のコードの問題に対処する](#)
- [Visual Studio のコードの問題に対処する](#)

JetBrains および Visual Studio Code のコードの問題に対処する

JetBrains および Visual Studio Code のコードの問題に対処するには、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。

以下のアクションを実行できます。

- インプレースコード修正を生成する
- 問題を説明して新しいコードを取得する
- 問題を無視する、または同様のすべての問題を無視する

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。

ファイル内のコードの問題を自動的に修正するには:

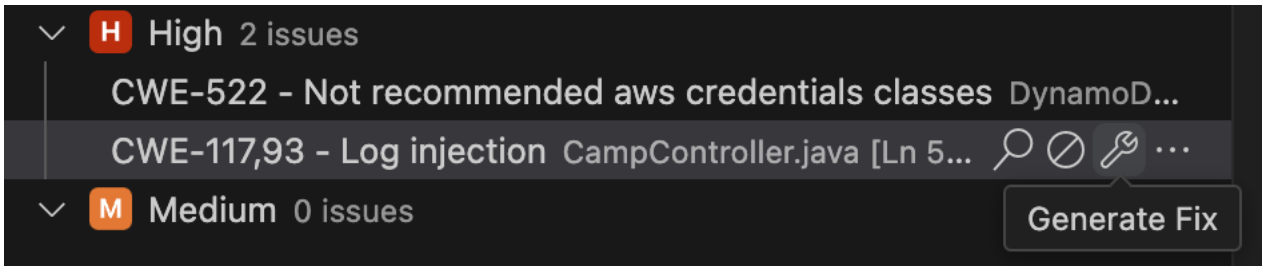
JetBrains

1. 問題ツールウィンドウで、Amazon Q Code Issues タブで、対処するコード問題を選択します。
2. パネルが開き、コードの問題に関する詳細情報が表示されます。該当する場合は、コードの問題の特定に使用された Amazon Q デイテクターの詳細が表示されます。
3. パネルの下部で、修正を選択します。
4. 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。
5. ファイル内にコードの変更が表示され、チャットパネルには変更を元に戻すオプションが表示されます。

Visual Studio Code

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. レンチアイコンを選択します。

次の図は、Visual Studio Code のコードの問題のレンチアイコンを示しています。



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. ファイル内にコードの変更が表示され、チャットパネルには変更を元に戻すオプションが表示されます。

コードの問題を説明して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、ファイルに追加するコードを添付した修復オプションを提供します。

コードの問題の説明を取得するには:

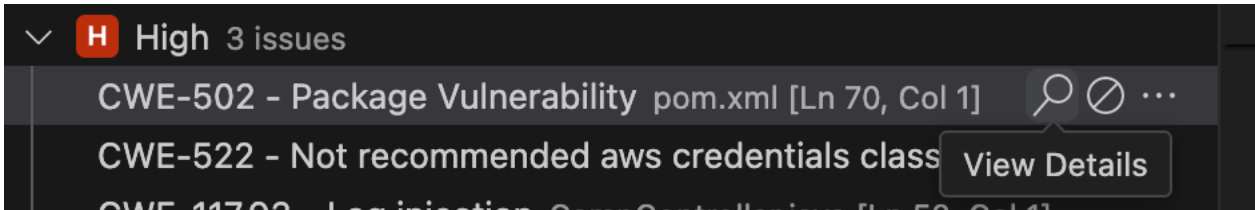
JetBrains IDEs

1. 問題ツールウィンドウで、Amazon Q Code Issues タブで、対処するコード問題を選択します。
2. パネルが開き、コードの問題に関する詳細情報が表示されます。該当する場合は、コードの問題の特定に使用された Amazon Q デイテクターの詳細が表示されます。
3. パネルの下部で、説明を選択します。
4. チャットパネルで、Amazon Q は問題の詳細を提供し、ファイルに挿入することができるコードを使用して問題を修正する方法を提案します。
5. 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。更新されたコードを追加するときは、必ず脆弱なコードを削除してください。

Visual Studio Code

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ます。

次の図は、Visual Studio Code のコードの問題の虫眼鏡アイコンを示しています。



3. チャットパネルで、Amazon Q は問題の詳細を提供し、ファイルに挿入することができるコードを使用して問題を修正する方法を提案します。
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。更新されたコードを追加するときは、必ず脆弱なコードを削除してください。

コードの問題を無視する

検出されたコードの問題が該当しない場合は、問題を無視するか、同様のすべての問題 (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、次の手順を実行します。

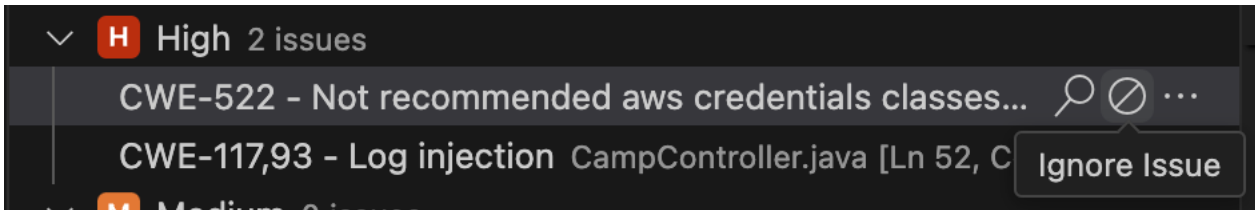
JetBrains

1. 問題ツールウィンドウで、Amazon Q Code Issues タブで、無視するコード問題を選択します。
2. パネルが開き、コードの問題に関する詳細情報が表示されます。パネルの下部で、無視を選択します。コード問題はコード問題パネルから削除されます。
3. Ignore All を選択して、同じ CWE でこのコードの問題やその他のコードの問題を無視することもできます。

Visual Studio Code

1. コードの問題タブで、無視するコードの問題を選択します。
2. 無視アイコンを選択します。

次の図は、Visual Studio Code のコードの問題の無視アイコンを示しています。



3. コード問題はコード問題パネルから削除されます。
4. 同様の問題を無視するには、省略記号アイコンを選択し、表示される類似問題を無視するボタンを選択します。

Visual Studio のコードの問題に対処する

Amazon Q で検出された問題を Visual Studio で表示するには、Visual Studio のメインメニューで [表示] の見出しを展開し、[エラー一覧] を選択して、Visual Studio の [エラー一覧] を開きます。

コードの問題の情報をを使用して、コードを更新します。コードを更新したら、コードを再度レビューして、問題が修正されている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 の [エラー一覧] で Amazon Q の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。[エラー一覧] で Amazon Q が検出したコードの問題を表示するには、[エラー一覧] の見出しの [警告] オプションを選択する必要があります。

エラー一覧でコードの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] 列ヘッダーで [フィルター] アイコンをクリックし、[Amazon Q] を選択して Amazon Q のコードの問題をフィルタリングします。

コードの問題をフィルタリングする

Note

コードの問題は JetBrains IDE と Visual Studio Code でのみフィルタリングできます。

コードの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の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して、パネルに選択した重要度の問題のみを表示できます。

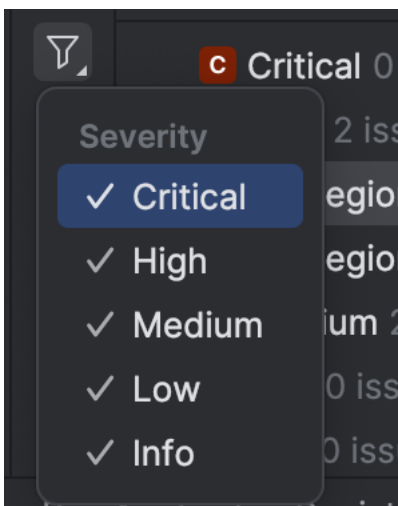
また、コードの問題パネルでコードの問題の表示方法をコントロールすることもできます。問題は、重要度またはファイルの場所に基づいてグループ化できます。

コードの問題をフィルタリングするには、次の手順を実行します。

JetBrains IDEs

1. Amazon Q Code Issues タブで、フィルターアイコンを選択します。
2. 重要度レベルのポップアップメニューが開きます。

次の図は、のコードの問題タブの重要度メニューを示しています IntelliJ IDEA。

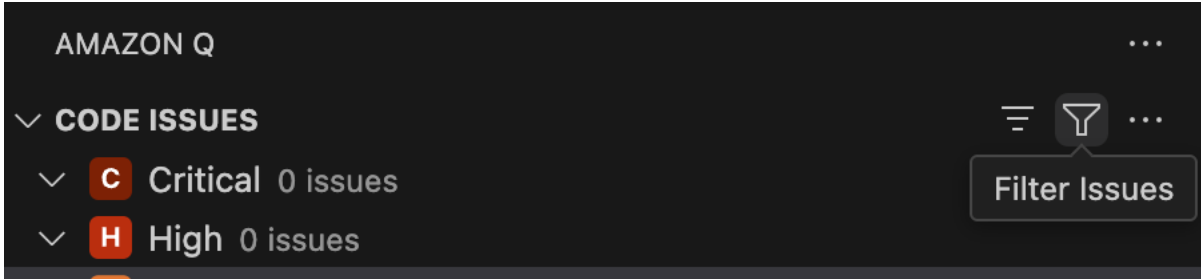


3. フィルタリングする重要度レベルを選択または選択解除し、OK を選択します。選択した重要度の問題のみが Amazon Q Code Issues パネルに表示されます。

Visual Studio Code

1. [コードの問題] パネルで、フィルターアイコンを選択します。

次の図は、Visual Studio Code の [コードの問題] タブのフィルターアイコンを示しています。



2. [問題のフィルタリング] メニューが開きます。

フィルタリングする重要度レベル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。選択した重要度の問題のみがコードの問題パネルに表示されます。

コードの問題をグループ化するには、次の手順を実行します。

JetBrains IDEs

1. Amazon Q Code Issues パネルから、グループ化アイコンを選択します。
2. Group By ポップアップ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。

Visual Studio Code

1. [コードの問題] パネルで、グループ化アイコンを選択します。

次の図は、Visual Studio Code の [コードの問題] タブのグループ化アイコンを示しています。



2. [問題のグループ化] 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 のコードレビューでのコードの問題の重要度

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基づいて、対処し追跡する問題の優先順位をつけることができますようにします。以下のセクションでは、コードの問題の重要度の判断方法と、各重要度レベルが意味する方法について説明します。

重要度の算出方法

コードの問題の重要度は、問題を検出したディテクターによって決まります。[Amazon Q Detector Library](#) の各ディテクターには、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、そのコンテキストで検出結果をどのように悪用できるか (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) 、および取得できるアクセスの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。

重要度決定マトリックス

アクセスのレベル	労力レベル	緊急度
システムまたはその出力の完全なコントロール	システムへのアクセスが必要	高
システムまたはその出力の完全なコントロール	高レベルの労力によるインターネット	重大

アクセスのレベル	労力レベル	緊急度
システムまたはその出力の完全なコントロール	インターネット経由	重大
機密情報へのアクセス	システムへのアクセスが必要	中
機密情報へのアクセス	高レベルの労力によるインターネット	高
機密情報へのアクセス	インターネット経由	高
システムがクラッシュまたは遅くなる可能性がある	システムへのアクセスが必要	低
システムがクラッシュまたは遅くなる可能性がある	高レベルの労力によるインターネット	中
システムがクラッシュまたは遅くなる可能性がある	インターネット経由	中
追加のセキュリティを提供する	悪用不可	情報
追加のセキュリティを提供する	システムへのアクセスが必要	情報
追加のセキュリティを提供する	高レベルの労力によるインターネット	低
追加のセキュリティを提供する	インターネット経由	低
ベストプラクティス	悪用不可	情報

重要度の定義

重要度ラベルは次のように定義されています。

重大 - このコードの問題は、さらに悪化しないように直ちに修復する必要があります。

コードに関する重大な問題は、攻撃者がシステムをコントロールしたり、中程度の労力で動作を変更したりできることを示しています。重大な検出結果は、最大限の緊急性で対処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。

高 - この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コントロールしたり、高い労力で動作を変更したりできることを示しています。重要度の高い検出結果は、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 - このコードの問題は、中期的な優先事項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、またはシステムの利用不可につながる可能性があります。できるだけ早く、関連するリソース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 - このコードの問題には、独自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。

情報レベルの検出結果には、品質や読みやすさの向上、または代替 API 操作に関する提案が含まれます。すぐに対処する必要はありません。

Amazon Q Developer を使用した IDE でのコード変換

Amazon Q Developer は、言語およびオペレーティングシステム (OS) レベルのアップグレードと変換を自動的に実行することで、統合開発環境 (IDE) 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 によって変更が生成され、ユーザーはその変更を確認してファイルに適用できます。

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。その後、Amazon Q で実行する変換の種類に該当するトピックを参照してください。

変換をサポートする IDE と Amazon Q のインストール方法について詳しくは、「[IDE での Amazon Q Developer の使用](#)」を参照してください。

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

Note

AWS Transform カスタムが Java アップグレードで利用可能になりました。バージョンアップグレード、SDK 移行などを処理し、すべての実行で改善されるエージェント AI。 [開始方法](#)

Amazon Q は、Java アプリケーションの次の種類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 へのデータベース移行のための埋め込み SQL 変換

開始するには、実行する変換の種類に該当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 を使用したコマンドラインでのコード変換](#)
- [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 による 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 – Amazon Q が 1 か月間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 での .NET 変換](#)を含む、IDE で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 か月間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: 1,000 行のコード	
1 か月あたりのコード行	無料利用枠: 2,000 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 25 ジョブ	
1 か月あたりのジョブ	Pro ティア: 1,000 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することができます。Amazon Q がコードのアップグレードで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 API 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。

コードを変換する前に、Amazon Q はまずソース言語バージョンでコードを構築し、コードの変換に必要な情報が揃ってい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にターゲット JDK との互換性を持たせるための必要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 でのコード変換の仕組みの詳細については、「[Amazon Q Developer が Java 言語アップグレードのコードを変換する仕組み](#)」を参照してください。

トピック

- [サポートされている Java アップグレードと IDE](#)
- [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仕組み](#)

サポートされている Java アップグレードと IDE

Amazon Q の変換は、現在、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。同じ Java バージョンへのコード変換には、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 アップグレード

ソースコードバージョン	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 IDE で Java アップグレードをサポートしています。

- JetBrains IDE のモジュール
- Visual Studio Code のプロジェクトとワークスペース

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、Maven 上に構築されている。
- プロジェクトは Maven を使用して IDE で正常にビルドできる。Maven 3.8 以降が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。
- アプリケーションは UTF-8 文字のみを使用している。アプリケーションで非 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。

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)

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およびファースト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ーの依存関係とは、組織が管理し、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可能なライブラリ、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際に、ファーストパーティー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ー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 (例: 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、サードパーティーの依存関係とそのバージョンを YAML ファイルで指定して、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ます。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、使用する変換の指定。
- `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ています。
- `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ています。
- `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ています。
- `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` - 依存関係がファーストパーティーかサードパーティーか (`FIRST_PARTY` または `THIRD_PARTY` のいずれかで指定)。

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

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、開いたチャットパネルでアプリケーションを変換するように Amazon Q に依頼します。
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、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。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、[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 \(オプション\)](#)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。

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、開いたチャットパネルでアプリケーションを変換するように Amazon Q に依頼します。
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、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。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、[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 \(オプション\)](#)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。
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仕組み

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にターゲット JDK との互換性を持たせるための必要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、ターゲット Java バージョンへのコードのアップグレードを試みます。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 から 17
- Java 8 から 21
- Java 11 から 17
- Java 11 から 21
- Java 17 から 21

Amazon Q は、コードをターゲット 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。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、サードパーティーの依存関係とそのバージョンを YAML ファイルで指定して、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、廃止された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 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する。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、サーバー側で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、コードをサーバー側に送信して変換を続行します。

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 生成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 は生成するコードを確認し、セキュリティ上の懸念に対応するように更新します。

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 30 日後まで使用できます。

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、[特微量開発](#)や [Workspace コンテキスト](#)などの機能を使用して、より多くのプロジェクトをコンテキストとして含め、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、AWS Database Migration Service 埋め込み SQL を (AWS DMS) を使用して 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。AWS DMS の DMS Schema Conversion を使うと、ターゲットデータベースに適用できるように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ことができます。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。埋め込み SQL は手動で変換するのではなく、IDE で Amazon Q を使用して自動的に変換できます。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に対応するバージョンに変換します。

現在 Amazon Q では、PostgreSQL に移行する Oracle データベースの Java アプリケーションに埋め込まれた SQL を変換できます。SQL コードを変換するオプションは、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場合のみ IDE で表示されます。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、以下の前提条件を満たしていることを確認します。

- 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するため、変換の要件を満たすには、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。
- AWS DMS Schema Conversion を使用してデータベーススキーマを変換するプロセスを済ませておきます。詳細については、「データベース移行ガイド」の「[Migrating Oracle databases 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、プロジェクト構造の設定で、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 のドキュメントの「[Project structure settings](#)」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に対応す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、変換する埋め込み SQL が含まれている Java コードベースを開きます。
2. Amazon Q アイコンをクリックして、チャットパネルを開きます。
3. チャットパネルでアプリケーションを変換するように Amazon Q に依頼します。
4. Java アプリケーションが SQL 変換の条件を満たしている場合、実行する変換の種類を選択するように求められます。**SQL conversion** と入力します。
5. 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られます。ファイルを取得する手順がチャットで示されます。

6. 埋め込み SQL が含まれるプロジェクトとデータベーススキーマファイル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 Amazon Q が取得した詳細が正確であることを確認します。
8. Amazon Q が 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. SQL コードを変換した後、ファイルに加えられた更新が差分として提示されます。差分に示された変更を確認し、変更を受け入れてコードを更新します。

Amazon Q では、変更の詳細を示す変換の概要も提供されます。

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 を使用したコマンドラインでのコード変換

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 はローカル環境でコードをビルドして、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 の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、サーバー側で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、コードをサーバー側に送信して変換を続行します。

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 生成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 は生成するコードを確認し、セキュリティ上の懸念に対応するように更新します。

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、「[Amazon Q Developer のコマンドラインで変換を実行する](#)」を参照してください。

変換を設定し、Amazon Q Developer Pro に対して認証するには、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_8、JAVA_11、JAVA_17、または JAVA_21 を入力できます。**<your-target-java-version>** には、JAVA_17 または JAVA_21 を入力できます。--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証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表示するには、以下を実行します。

```
qct -h
```

変換ジョブの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、「[コマンドラインでのジョブ履歴の表示](#)」を参照してください。

Amazon Q Developer の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でコードを変換するには、次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る](#) こと。
- 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る](#) こと。
- コマンドライン環境に Python がインストールされていること。これは、コマンドラインツールのインストール方法です。現在サポートされている Python の最小バージョンは 3.12 です。
- macOS または Linux で変換を実行していること。
- アプリケーションのサイズは 2 GB 以下であること。
- Amazon Q でアップグレードする特定の依存関係がある場合、[依存関係のアップグレードファイル](#)を設定済みであること。

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 アイデンティティセンターを認証して、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を持ってい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す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 [ユーザーにコマンドラインでの変換の実行を許可する](#) で定義されたアクセス許可があることを確認します。

Amazon Q Developer サブスクリプションを介した IAM アイデンティティセンターの認証

IAM アイデンティティセンターで認証するには、ユーザーが管理者によって [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](#) されており、ユーザーが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 を確認することができます。詳細については、[Amazon Q Developer で使用する開始 URL の表示](#) 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、「[アクセス許可を追加する](#)」を参照してください。

[ステップ 4: 設定および認証](#) で開始 URL を指定します。

ステップ 2: ツールのインストール

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、「[バージョン履歴](#)」を参照してください。

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するディレクトリでターミナルウィンドウを開き、以下を実行します。

```
python -m venv qct-cli
```

3. 仮想環境を有効にするには、次の手順を実行します。

```
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-1.2.2-py3-none-any.whl
```

Linux_x86_64

```
pip install <path/to/unzipped-tool>/Linux_x86_64/amzn_qct_cli-1.2.2-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、1.2.2 をダウンロードした[バージョン](#)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およびファースト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ーの依存関係とは、組織が管理し、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可能なライブラリ、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際に、ファーストパーティー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ー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 (例: 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、サードパーティーの依存関係とそのバージョンを YAML ファイルで指定して、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ます。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、使用する変換の指定。
- 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ています。
- 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ています。
- 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ています。
- 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 - 依存関係がファーストパーティーかサードパーティーか (FIRST_PARTY または THIRD_PARTY のいずれかで指定)。

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: 設定および認証

変換を開始する前に、IAM アイデンティティセンターで認証し、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 アイデンティティセンターで認証するには、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

ジョンコードの一覧については、「AWS 全般のリファレンス guide」の「[Regional endpoints](#)」を参照してください。

4. 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表示します。

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。プランに対して次の変更をリクエストすることができます。

- プランに含まれる一覧からどのライブラリを Amazon Q がアップグレードするか
 - 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 でどのステップを実行するか
 - 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のみ選択可)
 - プロンプトの例:
 - <dependency1> も <version2> にアップグレードする

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> を変換するコードを含むフォルダへのパスに、<your-target-java-version> を JAVA_17 または JAVA_21 に、それぞれ置き換えます。

```
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ません。また、変更をリクエストする機会也没有ありません。
2. Maven バージョンは、変換を開始する前に検証されます。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 Maven バージョンがない場合は、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 は変換プランに関するフィードバックを提供するようプロンプトが表示されます。変更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は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、qct configure 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。

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 [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](#) を読み、このタイプの変換の前提条件を理解してください。

1. 埋め込み SQL を変換するには、まず [AWS DMS スキーマ変換](#) 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。

必要なファイル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 変換を開始します。<path-to-folder> を変換するコードを含むフォルダへのパスに、<path-to-sql-config-file> をステップ 1 で作成した YAML ファイルへのパスに、それぞれ置き換えます。

```
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一覧を提供します。SQL 変換に使用するスキーマを選択し、YAML ファイルに新しいフィールド schema: <schema-name> 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。

変換の一時停止またはキャンセル

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、再開するまで最大 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、次の手順を実行します。

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一時停止する場合は、1 を入力します。12 時間以内にジョブを再開して、QCT コマンド ``qct transform --source_folder=≤/Path/Given/Originally/To/QCT>`` を使用してコード変換を続行できます。
- コード変換ジョブをキャンセルする場合は、2 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 の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ります。

```
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: '))))
```

このエラーを修正するには、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？

変換用に新しいバージョンのコマンドラインツールをダウンロードしても、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-1.2.2-py3-none-any.whl --force-reinstall
```

Linux_x86_64

```
pip install <path/to/unzipped-tool>/Linux_x86_64/amzn_qct_cli-1.2.2-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、1.2.2 をダウンロードした[バージョン](#)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
1.2.2 (最新)	2026 年 2 月 26 日	AWS Transform custom のプロモーションバナーを QCT CLI に追加しました。変換コマンドの実行とヘルプテキストにバナーが表示されます。バナー出力を抑制する新しい <code>--skip-banner</code> フラグ。
1.2.1	2025 年 9 月 9 日	最初のプロジェクトアップロード中にファーストパーティの親 POM を含めるように Maven 拡張機能を更新
1.2.0	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。

バージョン	リリース日	リリースノート
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公開されており、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イテレーション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ーの依存関係を持つアプリケーションのジョブと失敗を再開する際の問題を解決するためのバグ修正。
0.5.1	2025 年 3 月 13 日	IAM で認証する場合、AWS リージョンを指定する必要はなくなりました。出力ログのジョブステータスに関す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。

バージョン	リリース日	リリースノート
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 では、Java 変換ジョブ履歴の包括的な概要が提示され、IDE とコマンドラインの両方で変換ジョブを追跡して確認できます。

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
- 時間 – 変換が完了するまでにかかった時間

- ジョブ ID – 変換ジョブの一意的識別子
- 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 – 変更の詳細が示された変換の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た変換のみがジョブ履歴に表示されます。機能のリリース日については、「[Amazon Q Developer ユーザーガイドのドキュメント履歴](#)」を参照してください。

IDE 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。

Visual Studio Code の変換ハブによって、Java 変換ジョブ履歴を包括的に確認できます。

変換ハブのテーブルには、過去 30 日での最新の変換ジョブ 10 件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、進行状況を追跡して、不足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、以前にダウンロードした過去のジョブの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のステータスを更新できます。Amazon Q が失敗したジョブを再開できる場合などは、まだサーバーに達していない可能性のある最新のステータスをサーバー側から取得できるよう、失敗したジョブを更新できます。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ために、完了したジョブを更新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

Visual Studio Code の場合、変換ジョブのすべての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な構造になっています。

```
.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` コマンドで変換ジョブ履歴とカスタマイズオプションにアクセスできます。

CLI の場合、変換ジョブの履歴情報は `.aws/qcodetransform/history/` ディレクトリにローカルに保存されます。

`qct history` 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
```

このコマンドは、デフォルトでは、一時停止中または進行中のジョブに加えて、最新の変換ジョブ 10 件を表示します。

--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。例えば、20 件のジョブを表示するには、以下のコマンドを実行します。

```
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 を使用して 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する 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する 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

変換の間、Amazon Q はコードをサービス所有の Amazon S3 バケットに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ある場合もあります。詳細については、「[Amazon S3 バケット URL および ARN の許可リストへの追加](#)」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。

ります。エラーログの情報をを使用して、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が。

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えるようになります。

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。

Server

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るように、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、[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、「[Amazon Q Developer を使用して サポートとチャットする](#)」を参照してください。

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 日間のみ入手できます。変換が完了してから 30 日を超えた場合は、変換を再スタートして差分ファイルをダウンロードしてください。

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する 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。

JetBrains で Amazon Q によって発行されたログを検索するには、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

VS Code で Amazon Q によって出力されたログを検索するには、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変換することができます。Amazon Q 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変換に必要な変更を決定し、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計画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、ダイナミックリンクライブラリ (DLL) や実行可能ファイルなどのビルド可能なコード単位と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 は変換ハブで段階的な変更を提供し、そこで進捗状況を監視できます。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アップデートします。

Amazon Q は、4 つの主要なタスクを実行して .NET アプリケーションを Linux に変換します。

- 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 への変換 – 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 にプロジェクトとパッケージを変換します。
- Linux 互換のためのコードの書き換え – 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して書き換えます。
- Linux 互換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、「[How it works](#)」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変換](#)
- [Amazon Q Developer が .NET アプリケーションを変換する仕組み](#)
- [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 変換の月ごと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変換

Visual Studio で 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変換するには、次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、アプリケーションが以下の前提条件を満たしていることを確認します。

- 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る。
- 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る。
- アプリケーションは UTF-8 文字のみを使用している。アプリケーションで非 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
 - Web 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で 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. [ソリューションエクスプローラー] を選択します。
4. ソリューションエクスプローラーで、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 で変換] を選択します。
5. [Amazon Q Developer で変換] 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の選択] 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。生成され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。

[変換ハブ] が開き、変換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を待機] 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、[変換ハブ] 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] を確認します。[概要を .md としてダウンロード] 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 変換ステータスで入力が必要とする場合は、いくつかのファイルを手動で更新して Linux でアプリケーションを実行する必要があります。

 - a. [アクション] ドロップダウンメニューで、[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 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で [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仕組み

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、ダイナミックリンクライブラリ (DLL) や実行可能ファイルなどのビルド可能なコード単位と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、それらを変換して、変換されたアプリケーションがビルド可能で使えるようにします。

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変換を開始します。最下位レベルの依存関係から順に処理します。Amazon Q が依存関係の変換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変更が含まれます。

- 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、API の特定と繰り返しの置き換え
 - NuGet パッケージと API 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
- 既存のコードを変換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の書き換え

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数、変更されたパッケージ数、変更された API 数など、アプリケーションに対して提案された変更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。また、ビルドログを使用してビルド概要を表示し、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。

変換の概要には Linux 変換ステータスも記載されており、アプリケーションの Linux 互換性を確保す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、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、コードの変更方法に関する推奨事項を確認します。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。

ファイルのインプレースアップデート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処すると、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

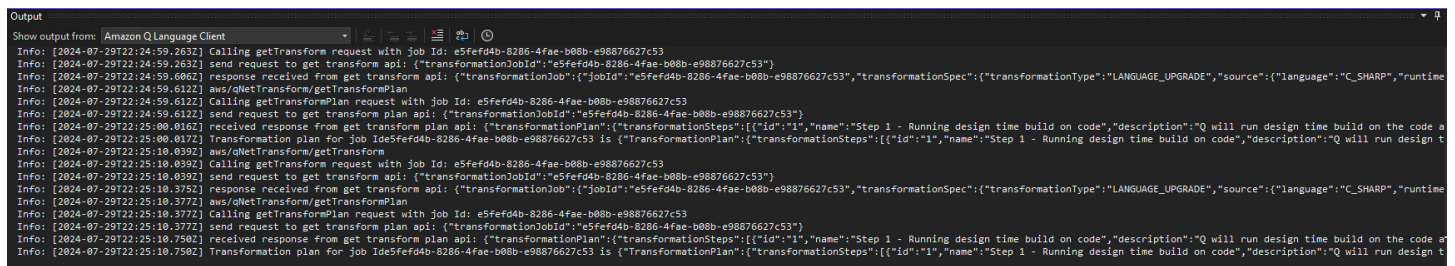
以下のセクションでは、Amazon Q Developer を使用した IDE での .NET 変換に関する一般的な問題のトラブルシューティングを行います。

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 が変換ハブ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。[次のソースの出力を表示:] メニューで、[Amazon Q 言語クライアント] 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。



```
Output
Show output from: Amazon Q Language Client
Info: [2024-07-29T22:24:59.263Z] Calling getTransform request with job Id: e5fef04b-8286-4fae-b08b-e98876627c53
Info: [2024-07-29T22:24:59.263Z] send request to get transform api: ("transformationJobId":"e5fef04b-8286-4fae-b08b-e98876627c53")
Info: [2024-07-29T22:24:59.606Z] response received from get transform api: ("transformationJob":{"jobId":"e5fef04b-8286-4fae-b08b-e98876627c53","transformationSpec":{"transformationType":"LANGUAGE_UPGRADE","source":{"language":"C_SHARP","runtime":
Info: [2024-07-29T22:24:59.612Z] aus/qNetTransform/getTransformPlan
Info: [2024-07-29T22:24:59.612Z] Calling getTransformPlan request with job Id: e5fef04b-8286-4fae-b08b-e98876627c53
Info: [2024-07-29T22:24:59.612Z] send request to get transform plan api: ("transformationJobId":"e5fef04b-8286-4fae-b08b-e98876627c53")
Info: [2024-07-29T22:25:00.016Z] received response from get transform plan api: ("transformationPlan":{"transformationSteps":[{"id":"1","name":"Step 1 - Running design time build on code","description":"Q will run design time build on the code a
Info: [2024-07-29T22:25:00.017Z] Transformation plan for job Id e5fef04b-8286-4fae-b08b-e98876627c53 is ("transformationPlan":{"transformationSteps":[{"id":"1","name":"Step 1 - Running design time build on code","description":"Q will run design t
Info: [2024-07-29T22:25:10.039Z] aus/qNetTransform/getTransform
Info: [2024-07-29T22:25:10.039Z] Calling getTransform request with job Id: e5fef04b-8286-4fae-b08b-e98876627c53
Info: [2024-07-29T22:25:10.039Z] send request to get transform api: ("transformationJobId":"e5fef04b-8286-4fae-b08b-e98876627c53")
Info: [2024-07-29T22:25:10.375Z] response received from get transform api: ("transformationJob":{"jobId":"e5fef04b-8286-4fae-b08b-e98876627c53","transformationSpec":{"transformationType":"LANGUAGE_UPGRADE","source":{"language":"C_SHARP","runtime":
Info: [2024-07-29T22:25:10.377Z] aus/qNetTransform/getTransformPlan
Info: [2024-07-29T22:25:10.377Z] Calling getTransformPlan request with job Id: e5fef04b-8286-4fae-b08b-e98876627c53
Info: [2024-07-29T22:25:10.377Z] send request to get transform plan api: ("transformationJobId":"e5fef04b-8286-4fae-b08b-e98876627c53")
Info: [2024-07-29T22:25:10.758Z] received response from get transform plan api: ("transformationPlan":{"transformationSteps":[{"id":"1","name":"Step 1 - Running design time build on code","description":"Q will run design time build on the code a
Info: [2024-07-29T22:25:10.758Z] Transformation plan for job Id e5fef04b-8286-4fae-b08b-e98876627c53 is ("transformationPlan":{"transformationSteps":[{"id":"1","name":"Step 1 - Running design time build on code","description":"Q will run design t
```

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、C# 言語のサポートされているプロジェクトタイプのみ変換できます。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変換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一覧については、「[ステップ 1: 前提条件](#)」を参照してください。

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。[次のソースの出力を表示:] メニューで、[Amazon Q 言語クライアント] 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 で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 `NODE_TLS_REJECT_UNAUTHORIZED` は、重要なセキュリティチェックを制御します。`NODE_TLS_REJECT_UNAUTHORIZED` を「0」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。
- 期限切れの証明書が許可されます。
- ホスト名が一致しない証明書が許可されます。
- その他の証明書検証エラーが無視されます。

プロキシが自己証明書を使用している場合は、`NODE_TLS_REJECT_UNAUTHORIZED` を無効にする代わりに、次の環境変数を設定できます。

```
NODE_OPTIONS = -use-openssl-ca  
NODE_EXTRA_CA_CERTS = Path/To/Corporate/Certs
```

それ以外の場合は、`NODE_TLS_REJECT_UNAUTHORIZED` 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。

Windows で `NODE_TLS_REJECT_UNAUTHORIZED` を無効にするには、次の手順を実行します。

1. スタートメニューを開き、環境変数を検索します。
2. [システム環境変数を編集する] を選択します。
3. [システムのプロパティ] ウィンドウで、[環境変数] をクリックします。
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 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 [プロンプトに送信] 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 とのインラインでのチャット

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、提案を求めるコードを強調表示し、

小さな入力画面で指示を入力します。Amazon Q はユーザーに代わってコードを生成し、メインコーディングウィンドウ内に差分として表示します。その後、変更を受け入れるか拒否するかをユーザーが選択できます。

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の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 (「このコードを説明して」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。

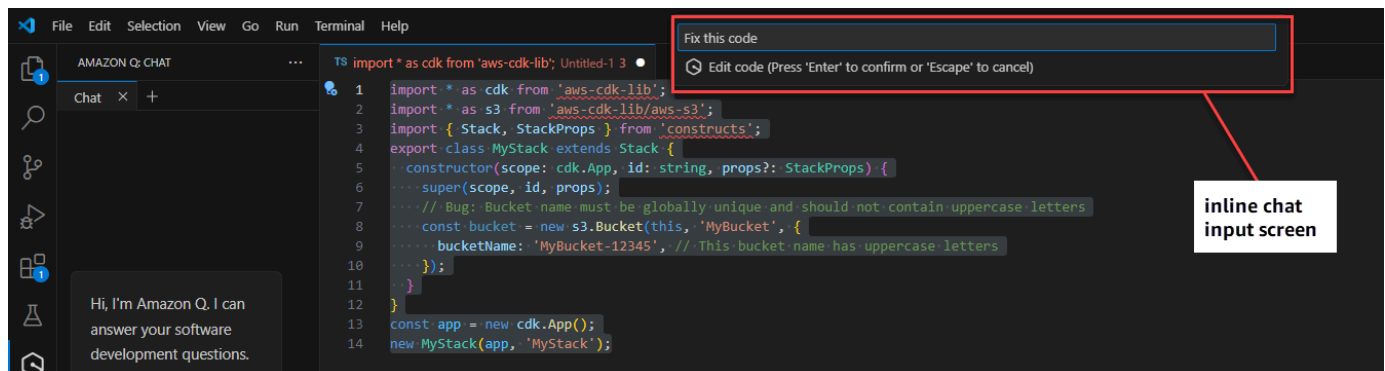
Amazon Q がインラインチャットを介してコード提案を生成する場合、現在のファイルのコードが考慮されます。他のファイルやプロジェクトのコードは考慮されません。

Amazon Q インラインチャットの動作

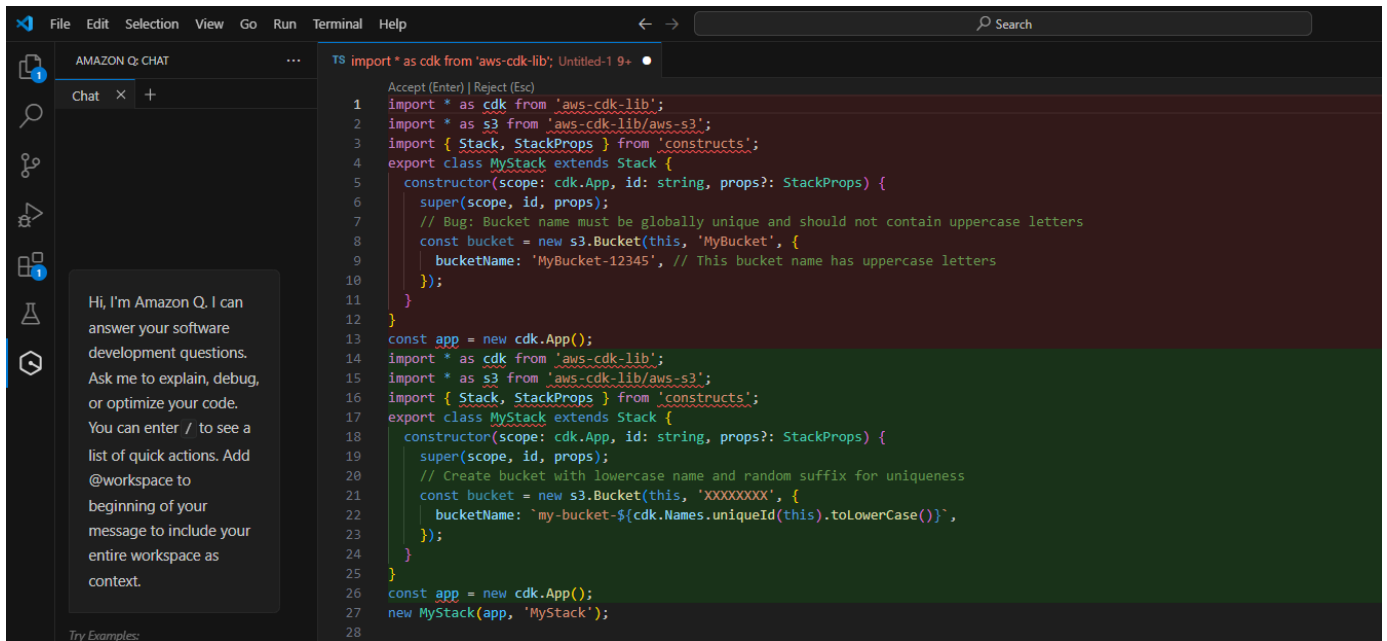
インラインチャットセッションは次のように展開されます。

1. 提案を求めるコードを強調表示し、使用している IDE に基づいて次のオプションから選択します。
 - Visual Studio Code と JetBrains では、`#+I` (Mac) または `Ctrl+I` (Windows) を押します。
 - Eclipse では、`#+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。

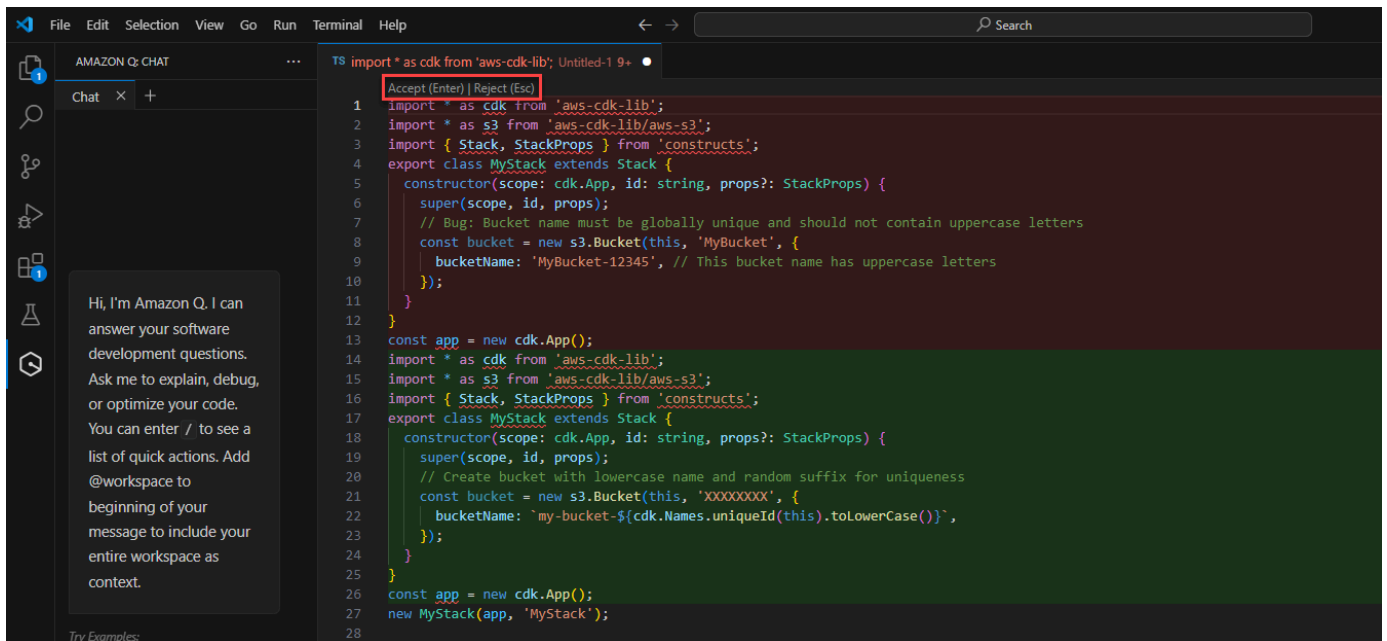
これにより、メインコーディングウィンドウの上部に小さな入力画面が起動し、**Fix this code** などのプロンプトを入力できます。



2. Amazon Q がコードを生成し、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、キーボードの該当するキー (Enter または Esc) を押します。



トピックおよび質問の例

インラインチャットでは、常にコードが回答として返されます。そのため、次のようなプロンプトを入力できます。

- このコードを記録して

- このコードをリファクタリングして
- この関数のユニットテストを記述して

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、下部に提案されたコードが表示されます。差分を横に並べて表示することはサポートされていません。

IDE 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 とチャットする際は、Amazon Q が回答を調整および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ためのコンテキストピッカーポップアップを起動します。また、@ と入力し、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始めて、オートコンプリート機能を使用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で提供するには、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、プロジェクトの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に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 @ と入力すると、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、選択したものを回答のコンテキストとして使用します。
- **画像** – Amazon Q では、画像をプロンプトのコンテキストとして追加できます。これは、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で役立ちます。画像は JPEG、PNG、GIF、または WebP 形式で、最大サイズは 3.75 MB、8,000 x 8,000 ピクセルです。コンテキストに固定された画像を含め、1 つのメッセージに最大 20 個の画像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、設定されている場合、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。

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 [設定を開く] を選択します。

3. [ワークスペースインデックス] 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。インデックス設定を更新するには、[ワークスペースインデックスのワーカースレッド] 設定で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を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ス処理ユニット (GPU) を使用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 とチャットする際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、プロンプトを毎回入力し直すことなく、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。

プロンプトをプロンプトライブラリに保存するには、次の手順を実行します。

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] を選択します。(検索するには下にスクロールする必要があります)。
4. [プロンプト名] に **Create_sequence_diagram** などのプロンプト名を入力し、Enter キーを押します。プロンプト名にスペースを含めることはできません。

Amazon Q は、`~/.aws/amazonq/prompts` フォルダに `Create_sequence_diagram.md` 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。例えば、次のようになります。

```
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、次の手順を実行します。

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. `Create_sequence_diagram` など、保存したプロンプトを選択します。
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。例えば、次のようなプロンプトを作成できます。

```
@Create_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

コンテキストのピン留めは現在、VS Code IDE でのみ使用できます。

コンテキストの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。ピン留めされたコンテキスト項目は、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**、**@folder** 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。

ピン留めには 2 つの方法があります。頻繁に参照する項目を手動でピン留めする方法と、Amazon Q が応答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方法です。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の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、デフォルトのピン留めコンテキストのみで新しくチャットが開始されます。

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、次の手順を実行します。

1. IDE で、Amazon Q チャットパネルを開きます。
2. チャットで **@workspace**、**@file**、**@folder**、**@prompt** などのコンテキストコマンドを使用した後、目的のコンテキストをクリックしてピン留めします。

または、**[@ コンテキストをピン留め]** 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することもできます。

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ピン留めする方法

コンテキスト項目をピン留めするには、3 つの方法があります。

1. ピンコンテキストメニューの使用:
 - チャットパネルの **[@ コンテキストをピン留め]** 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、選択した項目をピン留めします。
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**、**@file**、**@folder**、または **@prompt**) 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ピン留めします。

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の削除

- 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、ユーザーが追加したピン留めコンテキスト項目およびシステムが追加した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 とチャットする際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を記述したものです。例えば、すべての Python コードが型ヒントを使用する必要があること、またはすべての Java コードで Javadoc コメントを使用する必要があることを示すルールを作成できます。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、ルールに従って回答を生成します。チャットへのコンテキストの追加の詳細については、「[IDE の Amazon Q Developer チャットへのコンテキストの追加](#)」を参照してください。

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、次の手順を実行します。

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. [新しいルールを作成] 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. エディタにルールの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、次の手順を実行します。

1. IDE で、プロジェクトのルートフォルダを開きます。
2. プロジェクトのルートフォルダで、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. `project-root/.amazonq/rules` で、プロジェクトルールのファイルを作成します。これは、Markdown ファイルである必要があります。例えば、次のようになります。

`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. 詳細なプロンプトをファイルに追加します。例えば、次のようになります。

```
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 Markdown ファイルをさらに追加します。

これで、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、次の手順を実行します。

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で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、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 チャット用のメモリバンクの生成

Amazon Q は、プロジェクトの構造、テクノロジースタック、製品情報のクイックインデックスを提供するメモリバンクファイルを自動的に生成できます。この機能は、プロジェクト内のキーファイルを分析して、Amazon Q が質問するたびにプロジェクト全体を分析することなくコードベースを理解するのに役立つ概要ファイルを作成します。

メモリバンクファイルを生成すると、Amazon Q は次の自動生成されたファイルを含むmemory-bankサブフォルダ `.amazonq/rules` を作成します。

- `product.md` – プロジェクトとその機能の概要。
- `structure.md` – プロジェクトのアーキテクチャ、フォルダの編成、主要コンポーネント。
- `tech.md` – テクノロジースタック、フレームワーク、依存関係、コーディング標準。
- `guidelines.md` – プロジェクトの開発標準とパターン。

これらのファイルは、Amazon Q とチャットするときにコンテキストとして自動的に使用され、プロジェクトに関する背景情報を提供します。

プロジェクトのメモリバンクを生成する

メモリバンクを生成するには、次の手順を実行します。

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、ルールボタンを選択します。
3. メモリバンクの生成を選択します。
4. 新しいチャットタブが開き、Amazon Q がプロジェクトを分析してメモリバンクファイルを作成します。
5. 完了したら、ルールボタンを選択してファイルを表示できます。

質問するときに、コンテキストとして使用する個々のファイルを選択または選択解除できます。

6. プロジェクトが変更された場合は、Amazon Q で新しいメモリバンクファイルを生成してコンテキストを更新できます。これを行うには、ルールボタンを選択し、メモリバンクを再生成を選択します。

メモリバンク生成をカスタマイズする

カスタムプロジェクトルールを作成することで、メモリバンクファイルの生成方法をカスタマイズできます。たとえば、生成されたファイルの言語または形式を指定するルールを作成できます。

When generating the memory bank files like product.md, structure.md, and tech.md, always generate content in Spanish and include detailed code examples.

メモリバンクルールをプロジェクトの `project-root/.amazonq/rules` フォルダのファイルに保存します。

カスタムプロジェクトルールの作成の詳細については、「」を参照してください[Amazon Q Developer チャットで使用するプロジェクトルールの作成](#)。

Amazon Q Developer でのチャット履歴の圧縮

IDE で Amazon Q Developer とやり取りすると、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、より関連性の高い応答を返すのに役立つ重要なコンテキストとなります。ただし、基盤となるモデルへのリクエストごとに使用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、以下の情報が含まれます。

- 現在の質問またはリクエスト
- 会話での以前のメッセージ
- ユーザーが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

コンテキストウィンドウがこの容量制限に近づくと、Amazon Q は会話の以前の部分を参照できなくなる可能性があります。

チャット履歴の圧縮の仕組み

チャット履歴を圧縮すると、会話の重要な情報を保持しながら、使用するコンテキストの量を減らすことができます。圧縮は以下の仕組みで行われます。

1. Amazon Q が会話の履歴を分析します。
2. 重要なポイント、質問、決定事項の簡潔な概要が作成されます。
3. この概要が、コンテキストウィンドウの詳細な会話履歴に置き換えられます。
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。

圧縮によって、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ます。コンテキストウィンドウの制限に達しても、新しいチャットを開始する必要はありません。

チャット履歴の圧縮の使用

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以下の手順に従います。

1. チャットの入力フィールドに **/compact** と入力します。
2. Amazon Q が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が表示されます。

現在の会話を続けたいが、応答時間が遅くなっている、または関連性の低い回答が返ってきていると感じた場合は、手動圧縮を使用します。

自動で少しずつ圧縮

コンテキストウィンドウが容量の約 80% に達すると、圧縮を推奨する通知が表示されます。この通知には以下の要素が含まれています。

- 圧縮が推奨される理由の説明
- 圧縮をすぐに実施するためのボタン

圧縮後

圧縮後の動作は以下のとおりです。

- 現在のセッションが終了するまで、完全な会話履歴がチャットインターフェイスに表示されます。
- Amazon Q は、(完全な履歴ではなく) 圧縮された概要を使用して応答を生成します。
- コンテキストウィンドウに含まれるのは詳細な履歴ではなく圧縮された概要になります。

- 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

チャット履歴の消去

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消去できます。

1. チャットの入力フィールドに **/clear** と入力します。
2. 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がすべて削除されます。

履歴を圧縮するか消去するかの選択

以下の状況では圧縮を選択します。

- 現在の会話トピックを続行したい場合
- 以前のコンテキストが現在のタスクに関連している場合
- 会話の全般的な方向性と知識を維持したい場合

以下の状況では履歴を消去します。

- まったく新しいタスクまたはトピックを開始する場合
- 以前の会話が無関係になった場合
- 以前のコンテキストが新しい応答に影響を与えないようにしたい場合
- 会話から機密性の高い情報を削除したい場合

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。これらの会話は表示、検索、削除することができます。また、Markdown 形式や 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。

会話はワークスペースごとに個別に保存されるため、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。

会話を表示、検索、削除、エクスポートする場合は、次の手順を使用します。

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、[チャット履歴を表示] ボタンを選択します。
 - `ctrl+F` (Windows および Linux の場合) または `# F` (Mac の場合) 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ています。
 - 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 件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、`/clear` と入力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、[チャット履歴を表示] ボタンを選択します。
 - `ctrl+F` (Windows および Linux の場合) または `# F` (Mac の場合) 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 形式または HTML 形式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既に開始されている状態で、チャットパネルの右上にある [エクスポート] ボタンを選択し、タブに表示されてい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、[チャット履歴を表示] ボタンを選択します。
- `ctrl+F` (Windows および Linux の場合) または `# F` (Mac の場合) 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 形式または HTML 形式のファイルにエクスポートします。

デフォルトでは、ファイルは `q-dev-chat-yyyy-mm-dd.md/html` という名前でプロジェクトのルートに保存されます。

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 Visual Studio Code の IDE 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

[Action] (アクション)	Visual Studio Code ショートカット
コマンドを実行	Shift + Cmd + Enter (Mac)
	Shift + Ctrl + Enter (Windows)
	Shift + Meta + Enter (Linux)
コマンドを拒否	Shift + Cmd + R (Mac)
	Shift + Ctrl + R (Windows)
	Shift + Meta + R (Linux)
生成を停止	Shift + Cmd + Backspace (Mac)

[Action] (アクション)	Visual Studio Code ショートカット
	Shift + Ctrl + Backspace (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 の標準キーボードショートカットのカスタマイズインターフェイスから変更できます。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に合わせて自動的に更新されます。

ショートカットの確認方法

Amazon Q には、キーボードショートカットを確認するいくつかの方法があります。

- ツールヒント: コマンド実行ボタン、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の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できます。

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、フォーカスが 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 の更新が必要になる場合があります。

IDE 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 に使用させ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 での 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	200k
Claude Sonnet 4 (デフォルト)	200k

IDE でのチャットに使用するモデルの選択

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、以下の手順に従います。

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、モデルメニューの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、希望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とは、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コンテキスト (会話履歴、明示的または自動的なコンテキストなど) の量を指します。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、「[チャットへのコンテキストの追加](#)」を参照してください。

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。

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。

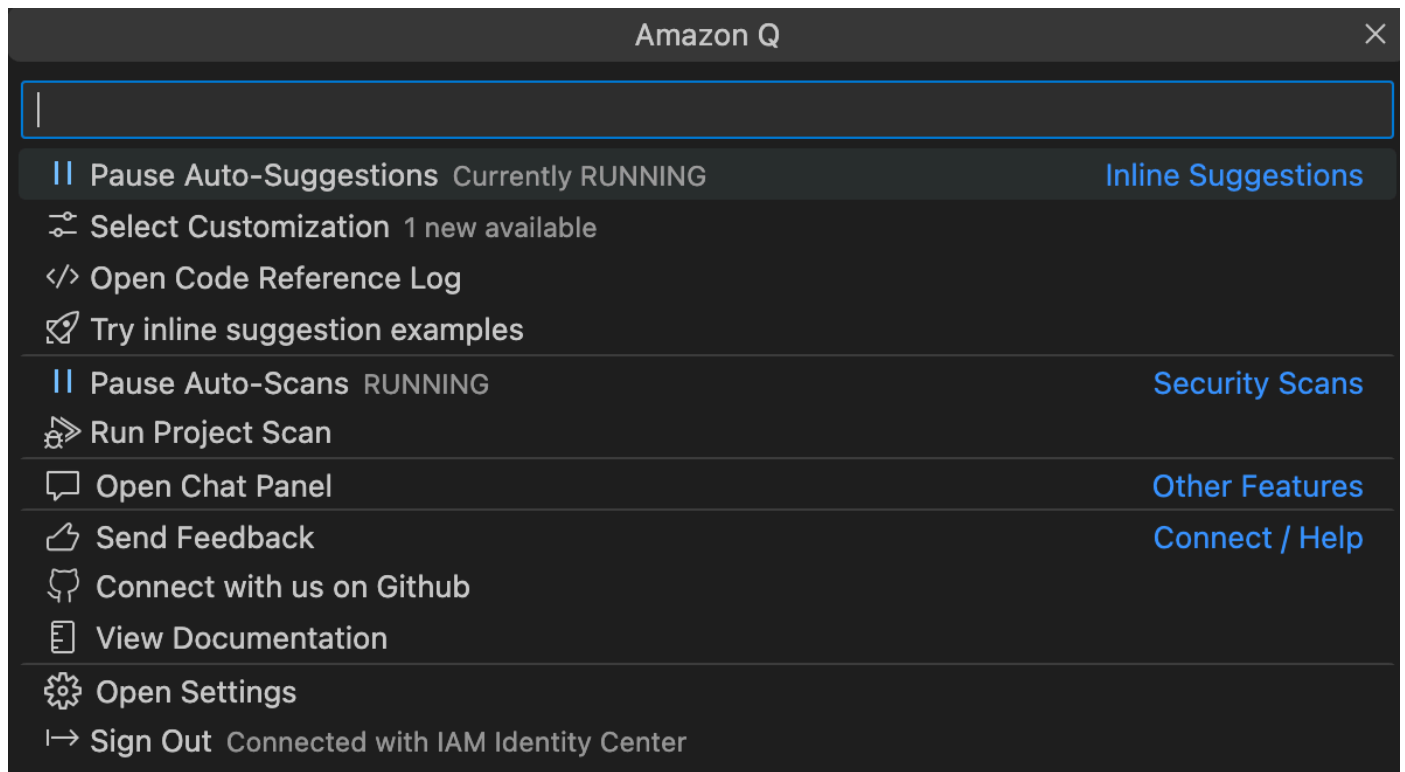
Visual Studio Code

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。

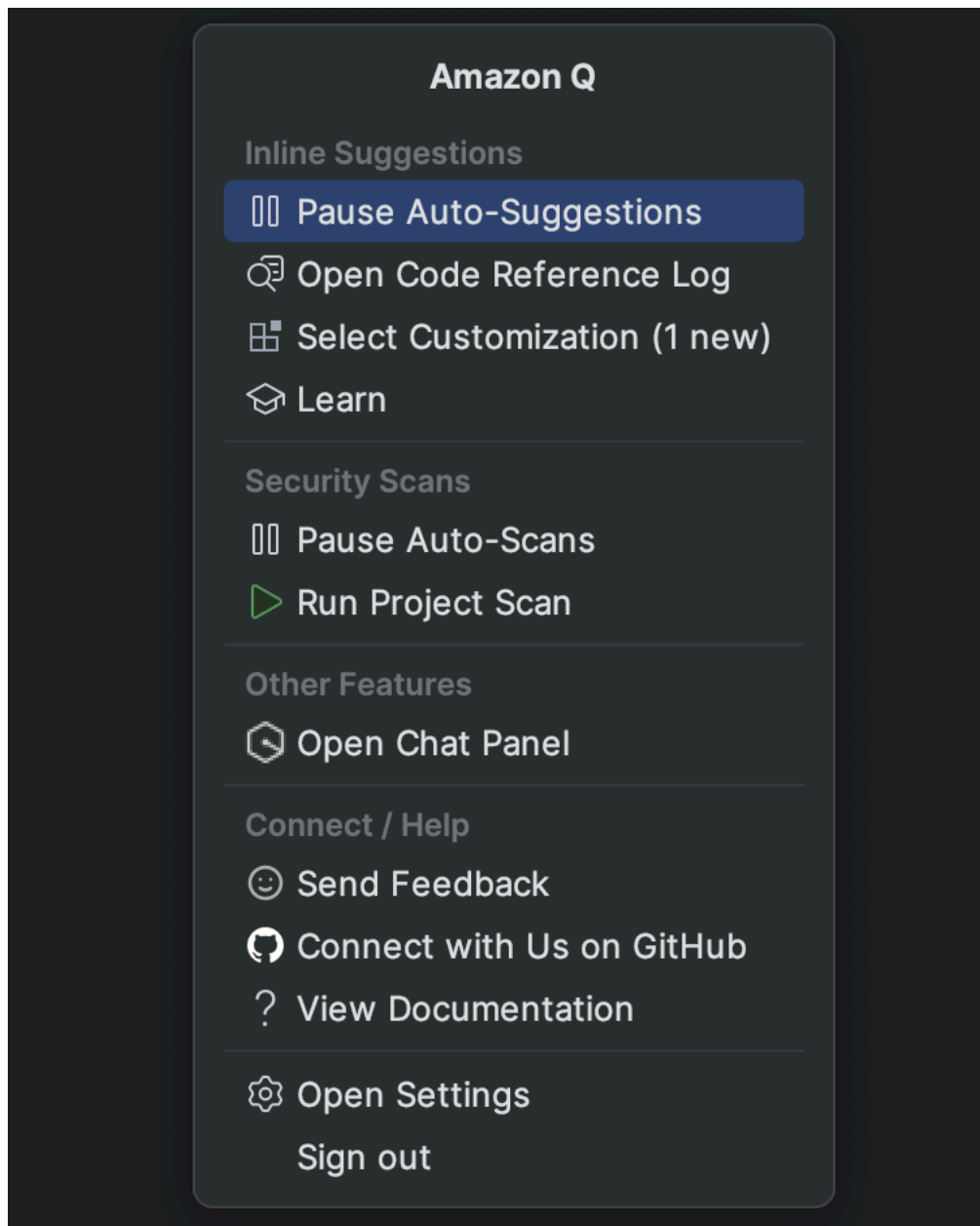
次の図は、VS Code の Amazon Q タスクバーを示しています。



JetBrains

1. JetBrains IDE で、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] から [自動提案の再開] を選択します。

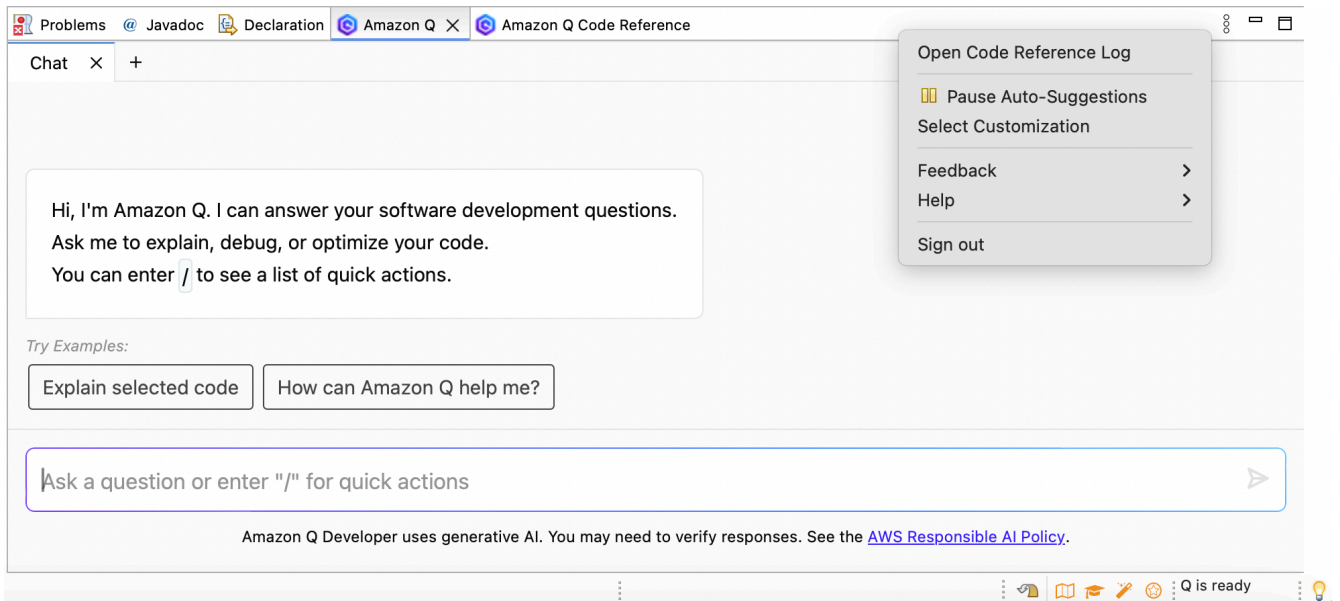
次の図は、JetBrains IDE の Amazon Q タスクバーを示しています。



Eclipse

1. Eclipse IDE で、IDE の右上隅にある Amazon Q アイコンを選択します。
2. Amazon Q の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、Eclipse IDE の Amazon Q タスクバーを示しています。

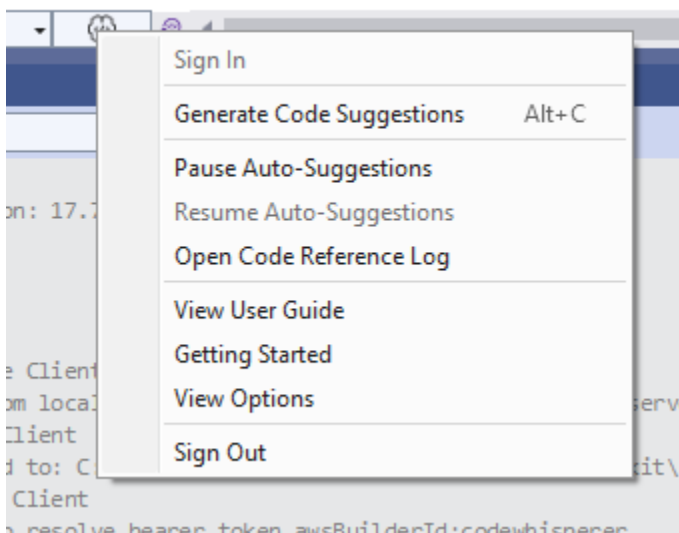


3. [自動提案の一時停止] から [自動提案の再開] を選択します。

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。

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。

```
basics > boto-whisper-demo.py
1  import boto3
2  from boto3.session import Session
3  import unittest
4  from boto
```

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。

```
basics > boto-whisper-demo.py
1  import boto3
2  from boto3.session import Session
3  import unittest
4  from botocore.exceptions import ClientError
5  import logging
6  import time
7
8  # set up logging
9  logging.basicConfig(level=logging.INFO)
```

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。

```
basics > boto-whisper-demo.py > ...
8  # set up logging
9  logging.basicConfig(level=logging.INFO)
10
11  #Create a new session
12  session = Session()
13
14  # define constants
15  DEFAULTREGION = 'us-east-1'
```

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。

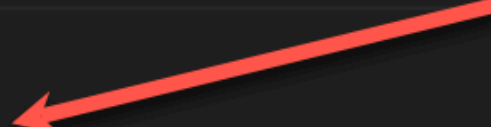
```
8  # set up logging
9  logging.basicConfig(level=logging.INFO)
10
11 #Create a new session
12 session = Session()
13
14 # define constants
15 DEFAULT_REGION = 'us-east-1'
16 TEST_BUCKET_NAME = 'my-test-bucket' + str(int(time.time()))
17 TEST_TABLE_NAME = 'my-test-table' + str(int(time.time()))
18
19 # AWS Clients with session
20 s3 = session.client('s3', region_name=DEFAULT_REGION)
   dynamodb = session.client('dynamodb', region_name=DEFAULT_REGION)
```



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。

```
18
19 # AWS Clients with session
20 s3_client = session.client('s3', region_name=us-east-1)
21 dynamodb_client = session.client('dynamodb', region_name=us-east-1)
22
23 def create_s3_bucket():
   """
   Creates a new S3 bucket
   """
   try:
       s3_client.create_bucket(Bucket=TEST_BUCKET_NAME)
   except ClientError as e:
       logging.error(e)
       return False
   return True
```



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。

```
40 def create_dynamodb_table(table_name, region=None):  
    # global dynamodb # Use the global dynamodb client created with the session  
    print(f"Using region: {region}")  
    print(f"DynamoDB endpoint URL: {dynamodb.meta.endpoint_url}") # Print the endpoint URL  
    try:  
        print(f"Creating table in region: {region}") # Add this line to debug  
        if region is None or region.lower() == 'us-east-1':  
            response = dynamodb.create_table(  
                TableName=table_name,  
                KeySchema=[  
                    {  
                        'AttributeName': 'id',  
                        'KeyType': 'HASH' # Partition key  
                    }  
                ],  
                BillingMode='PAY_PER_REQUEST',  
                ProvisionedThroughput=1000  
            )  
    except Exception as e:  
        print(f"Error creating table: {e}")  
    return response
```

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):
72         self.s3 = session.client('s3', region_name=DEFAULT_REGION)
73         self.dynamodb = session.client('dynamodb', region_name=DEFAULT_REGION)
74         self.s3_resource = session.resource('s3', region_name=DEFAULT_REGION)
75         self.dynamodb_resource = session.resource('dynamodb', region_name=DEFAULT_REGION)
76
77     def tearDown(self):
78         self.s3.delete_bucket(Bucket=TEST_BUCKET_NAME)
79         self.dynamodb.delete_table(TableName=TEST_TABLE_NAME)
80
81     def test_create_s3_bucket(self):
82         self.assertTrue(create_s3_bucket(TEST_BUCKET_NAME, DEFAULT_REGION))
83
84     def test_create_dynamodb_table(self):
85         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():
88         create_s3_bucket(TEST_BUCKET_NAME, region='us-east-1')
89         create_dynamodb_table(TEST_TABLE_NAME, region='us-east-1')
```

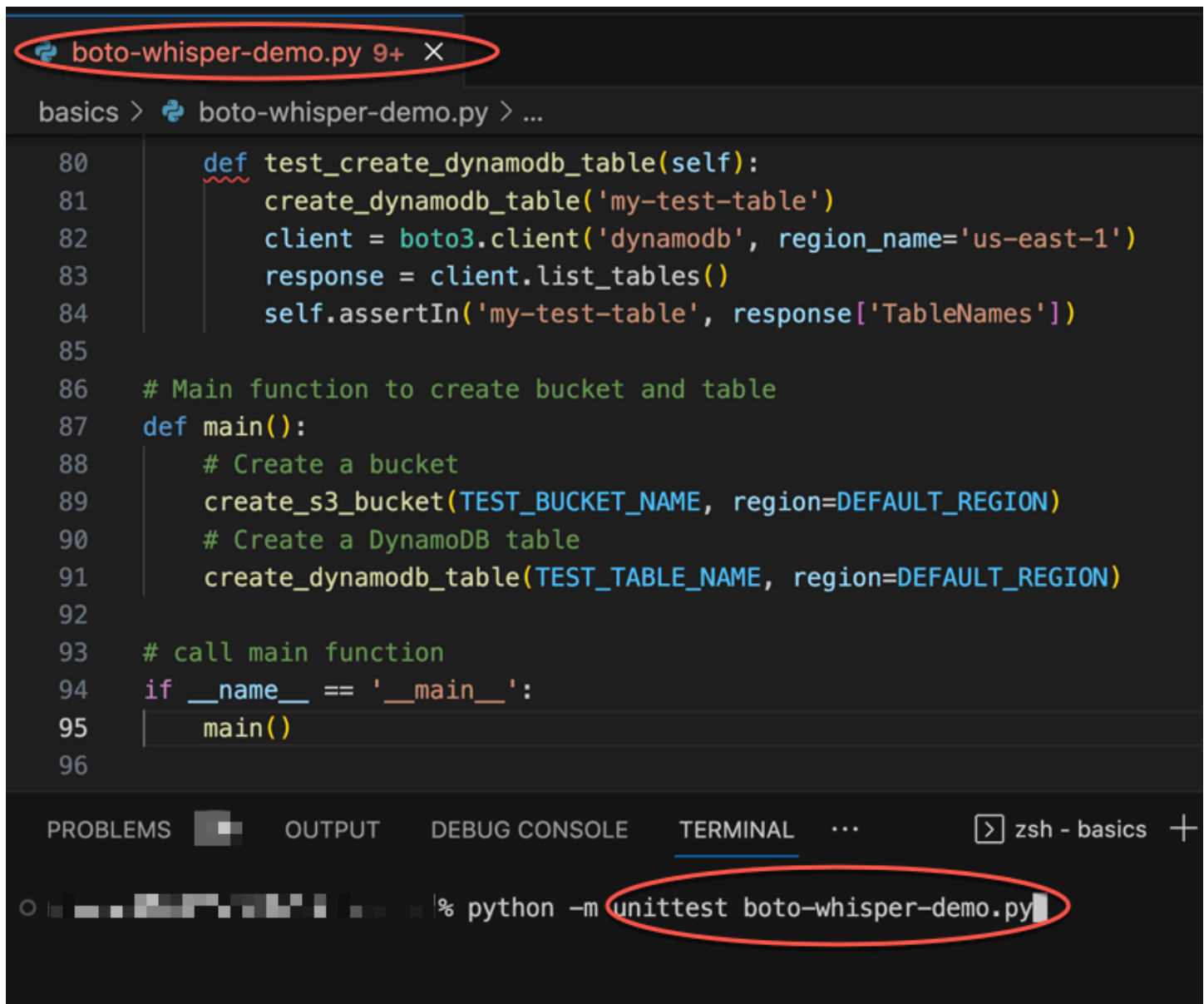
残っているのはメインガードだけで、Amazon Q はそのことを理解しています。

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。


```
# Main function to create bucket and table
def main():
    # Create a bucket
    create_s3_bucket(TEST_BUCKET_NAME, region=DEFAULT_REGION)
    # Create a DynamoDB table
    create_dynamodb_table(TEST_TABLE_NAME, region=DEFAULT_REGION)

# call main function
if __name__ == '__main__':
    main()
```

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。



The screenshot displays the Amazon Q Developer IDE interface. At the top, a tab for `boto-whisper-demo.py` is highlighted with a red circle. Below the tab, the code editor shows a Python script with the following content:

```
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():
88     # Create a bucket
89     create_s3_bucket(TEST_BUCKET_NAME, region=DEFAULT_REGION)
90     # Create a DynamoDB table
91     create_dynamodb_table(TEST_TABLE_NAME, region=DEFAULT_REGION)
92
93 # call main function
94 if __name__ == '__main__':
95     main()
96
```

At the bottom of the IDE, the **TERMINAL** tab is active, showing a command prompt with the command `python -m unittest boto-whisper-demo.py` entered. The command is highlighted with a red circle.

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、Amazon Q 関連のアクセス許可を SageMaker AI 実行ロールに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 ティアのどちらを使用しているかによって異なります。

Amazon Q を Amazon SageMaker AI Studio 用に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「[Set up Amazon Q Developer for your users](#)」を参照してください。

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書き留め、以下のセクションのいずれかの対応する指示に従ってください。

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. ウィンドウの下部にある Amazon Q パネルから、開始方法を選択します。

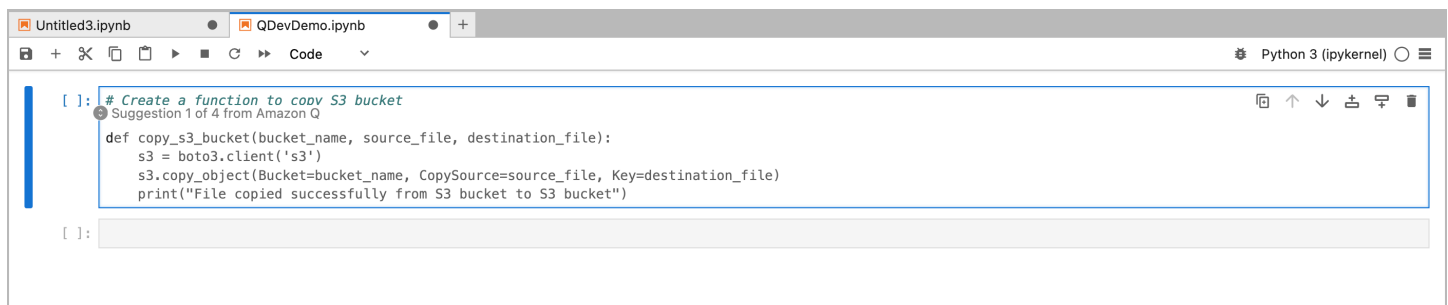
3. ポップアップウィンドウで、[コードをコピーして進む] を選択します。
4. 開始方法ページで、E メールアドレスまたは Google アカウントを使用して Builder ID にサインインまたはサインアップします。詳細については、「[個人アカウント \(Builder ID\) を使用した開始方法](#)」を参照してください。

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. 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。
6. 次の画面で、パスワードを選択して確認し、[AWS ビルダー IDを作成] を選択します。
7. 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。



```
[ ]: # Create a function to copy S3 bucket
      Suggestion 1 of 4 from Amazon Q
      def copy_s3_bucket(bucket_name, source_file, destination_file):
          s3 = boto3.client('s3')
          s3.copy_object(Bucket=destination_file, CopySource=source_file, Key=destination_file)
          print("File copied successfully from S3 bucket to S3 bucket")

[ ]:
```

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 を使用する AWS Glue ように を設定します。

1. [AWS Glue Studio ノートブックをセットアップ](#)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。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 Identity and Access Management 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	コマンド + 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: Option +] Windows: Alt +]
前の提案	MacOS: Option + [Windows: Alt + [
提案を拒否する	ESC、バックスペース、または入力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。

Eclipse でキー割り当てを変更するには、Eclipse のドキュメントで「[Changing the Key Bindings](#)」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	Windows: Alt + C
キーバインドの <code>AWSToolkit.CodeWhisperer.GetSuggestion</code>	
提案を受け入れる	タブ
次の提案	Windows: Alt + .
キーバインドの <code>Edit.NextSuggestion</code>	
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

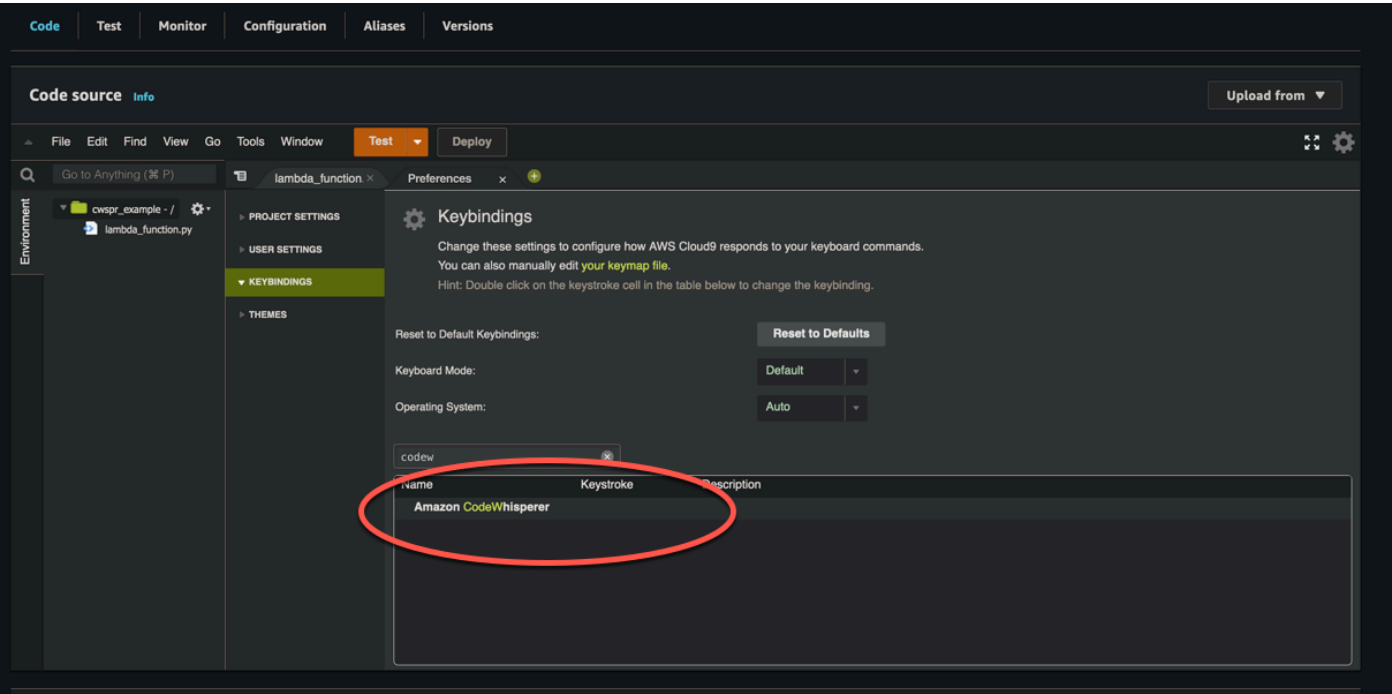
[Action] (アクション)	キーボードショートカット
Amazon Q を手動で開始する	MacOS: Option + C Windows: Alt + C
提案を受け入れる	タブ
次の提案	下矢印
前の提案	上矢印
提案を拒否する	ESC

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。

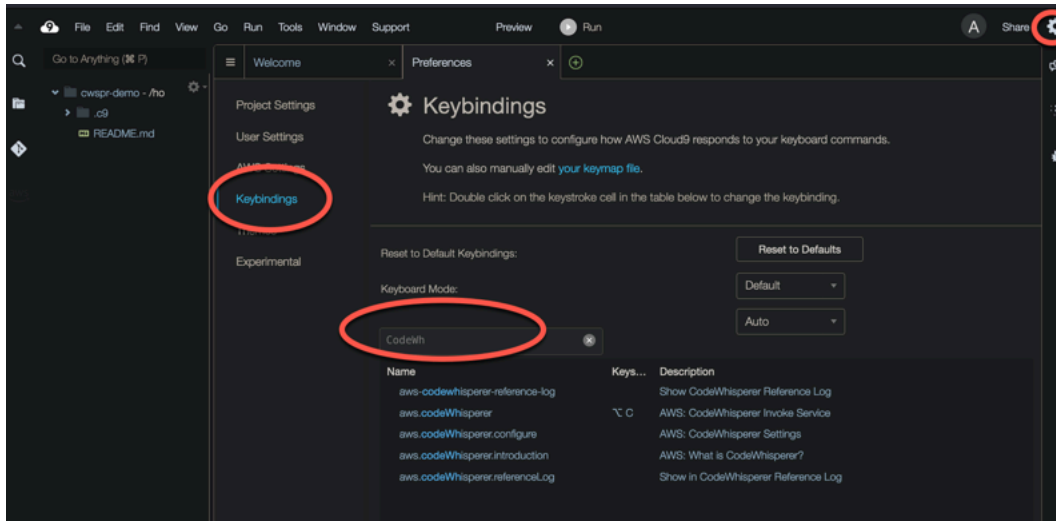


AWS Cloud9

[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提案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。

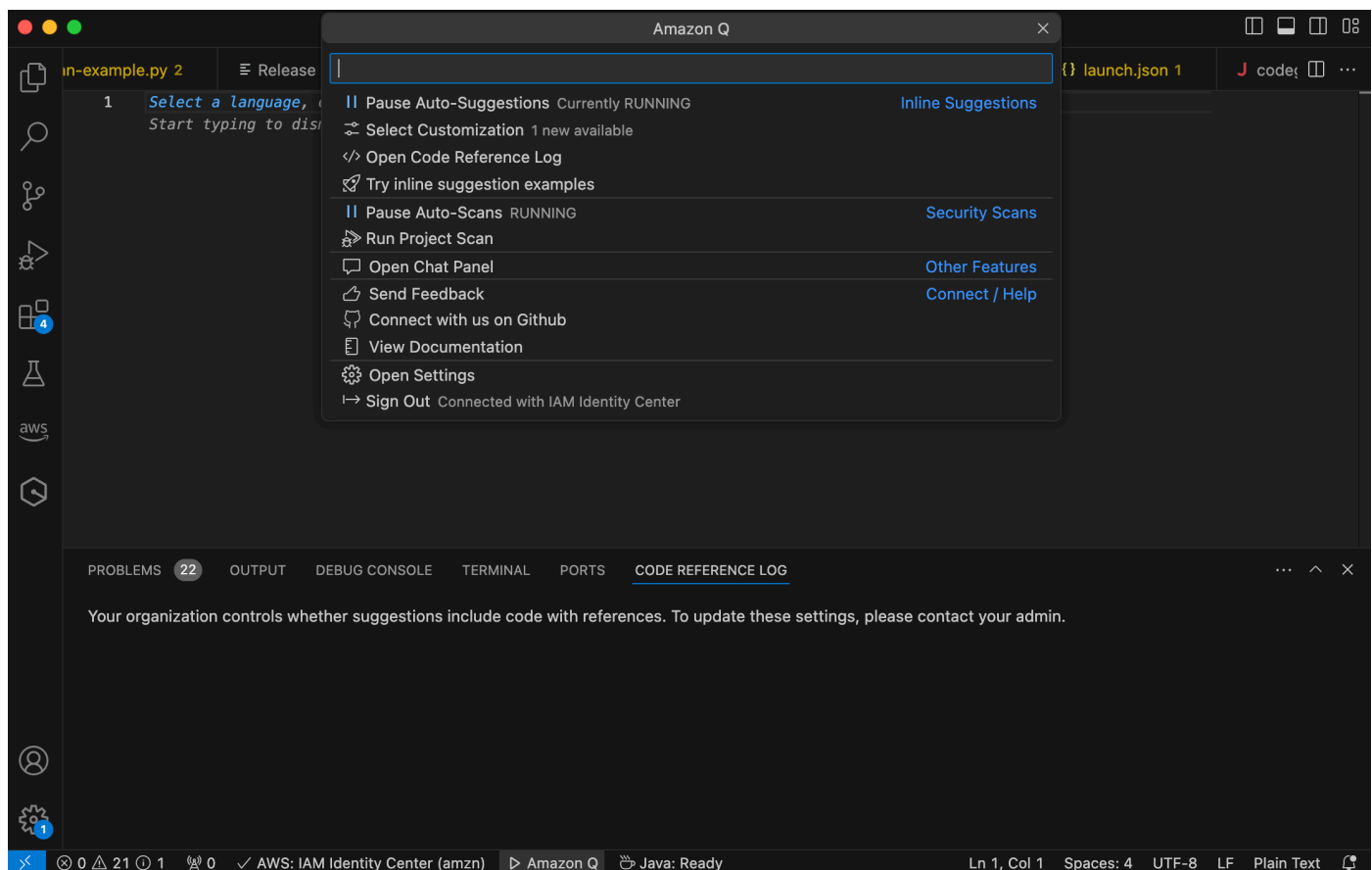
1. VS Code と Amazon Q 拡張機能の両方の最新バージョンを使用していることを確認してください。
2. VS Code で、IDE ウィンドウの下部にあるコンポーネントトレイから [Amazon Q] を選択します。

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。

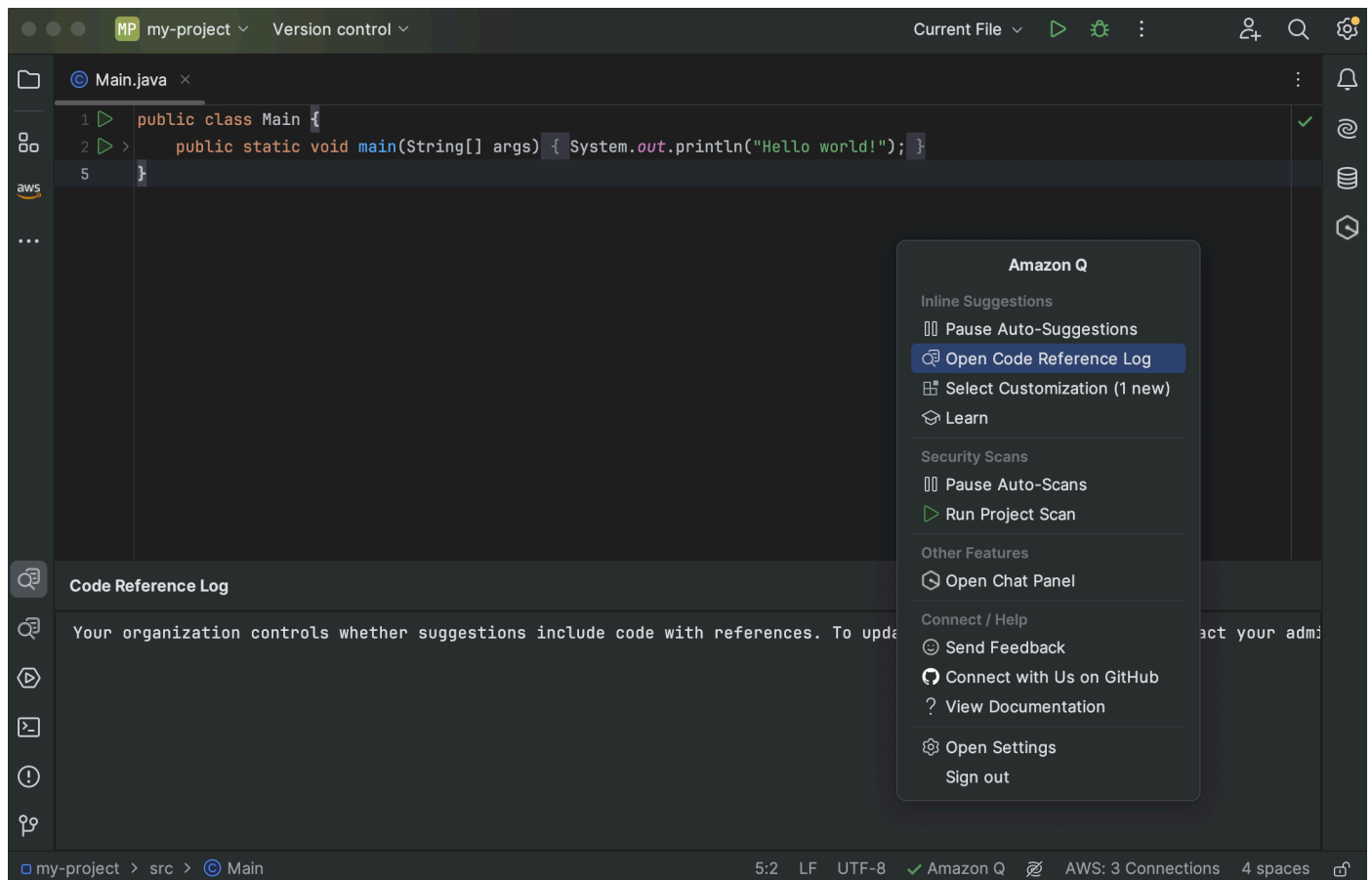
1. JetBrains IDE と Amazon Q プラグインの両方の最新バージョンを使用していることを確認してください。
2. JetBrains で、IDE ウィンドウの下部にあるステータスバーから [Amazon Q] を選択します。

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。

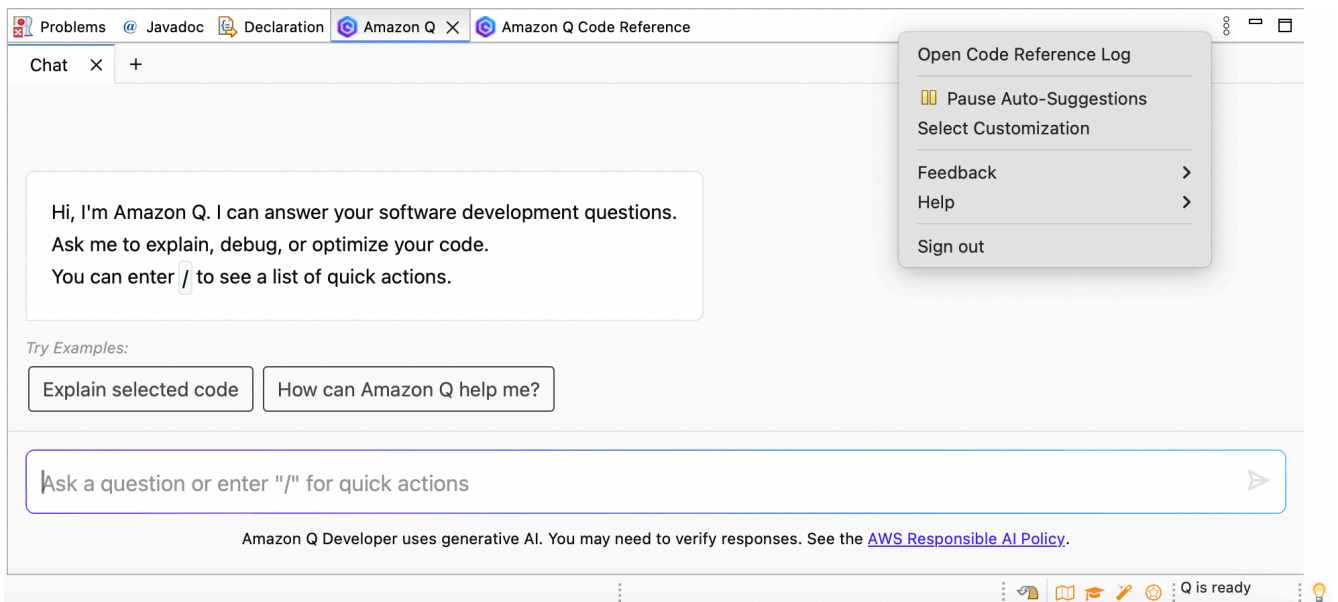


Eclipse

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. Eclipse 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ます。
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 の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、Eclipse IDE の Amazon Q 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。

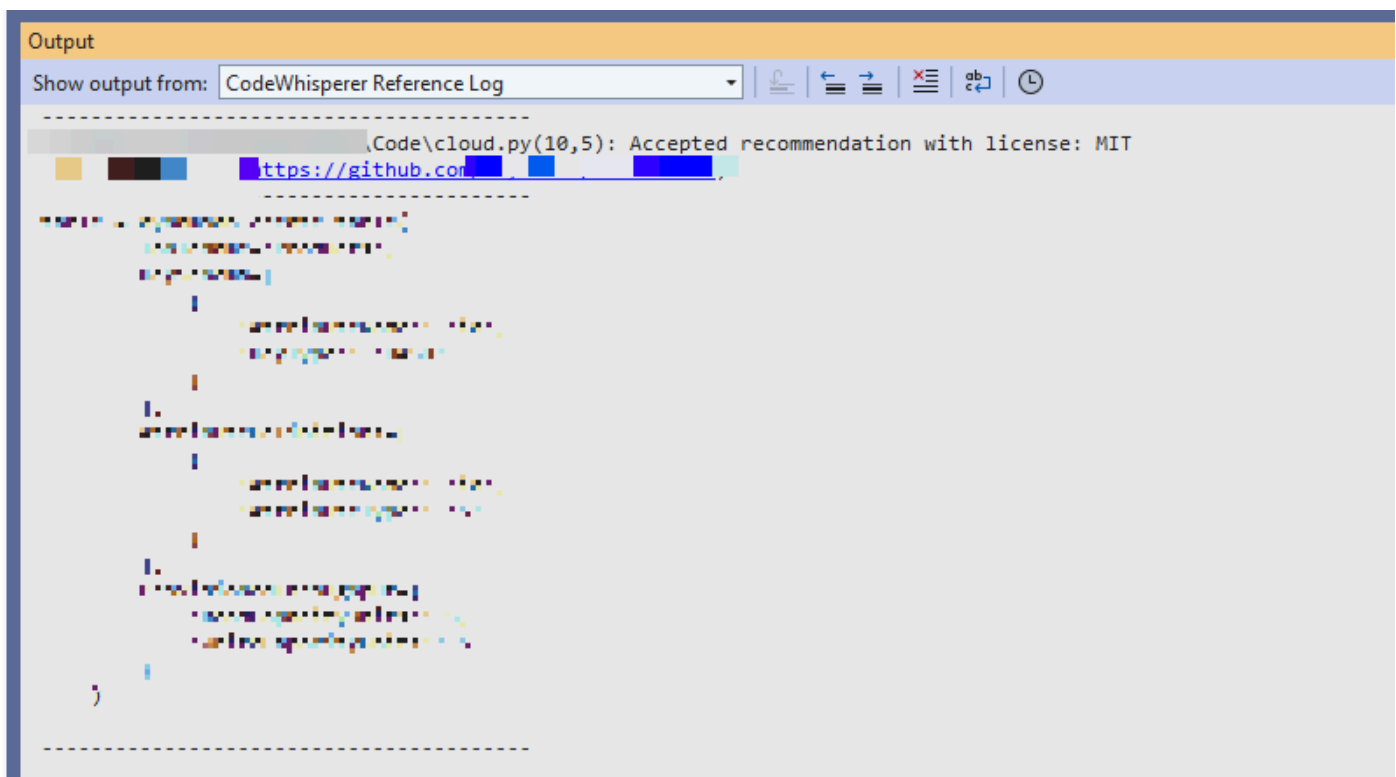
```
# Create function to create a DynamoDB Table
def Suggestion (License: MIT) 1 / 1 Tab to accept
    table = dynamodb.create_table(
        TableName='Products',
        KeySchema=[
            {
                'AttributeName': 'id',
```

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手順は次のとおりです。

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. [リファレンス付きでコードを表示する] 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。

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. [リファレンス付きでコードを表示する] 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。

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. Eclipse 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ます。
2. Eclipse IDE で、[設定] を開きます。
3. 左のナビゲーションペインで、[Amazon Q] を選択します。
4. [リファレンス付きでコードを表示する] 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。
5. [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. 左のナビゲーションバーで [AWS Toolkit] を選択し、[Amazon Q] を選択します。
4. [リファレンス付きで提案を含める] 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. SageMaker AI Studio ウィンドウの上部で、[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 では、SageMaker AI Studio 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。

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++

```

17 int main(int argc, char **argv) {
18     Aws::SDKOptions options;
19     Aws::InitAPI(options); // Should only be called once.
20     {
21         Aws::Client::ClientConfiguration clientConfig;
22
23         clientConfig.region = "us-east-1";
24
25         Aws::SQS::SQSClient sqsClient(clientConfig);
26
27         Aws::Vector<Aws::String> allQueueUrls;
28         Aws::String nextToken; // Next token is used to handle a paginated response.
29         do {
30             Aws::SQS::Model::ListQueuesRequest request;
31
32             // ... (request configuration) ...
33
34             } while (!nextToken.empty());
35     }
36 }

```

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。

```
1  /*
2  1  * Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
3  1  * SPDX-License-Identifier: Apache-2.0
4  1  */
5
6  // Upload an object to Amazon S3 bucket.
7
```

TypeScript

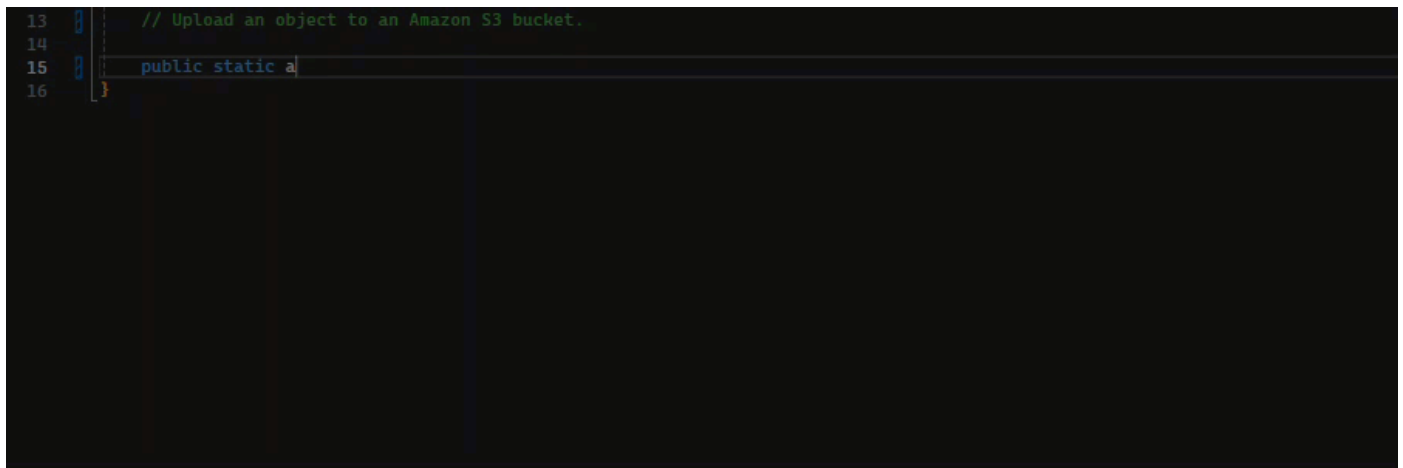
この例では、ユーザーが完全なコメントを入力し、Amazon Q がそれに従ってコードを提供します。



```
TS index.ts  ×
TS index.ts > ...
1  import { S3Client } from "@aws-sdk/client-s3";
2
3  const client = new S3Client({});
4
5  |
```

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。



```
13  // Upload an object to an Amazon S3 bucket.
14
15  public static a
16  }
```

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。

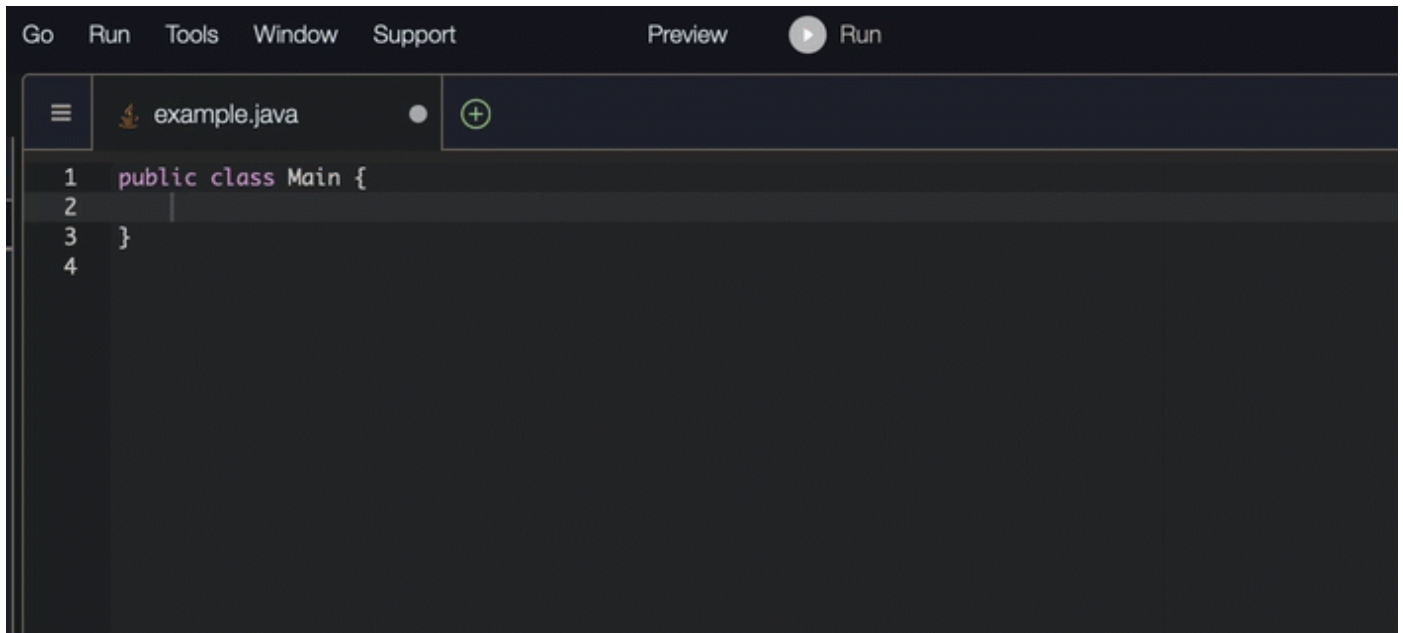
```
1 local access_key_response
2 access_key_response=$(iam_create_user_access_key -u "$user_name")
3 # shellcheck disable=SC2181
4 if [ $? -ne 0 ]; then
5     echo "ERROR: Failed to create access key for user '$user_name'"
6     exit 1
7 fi
8
9
```

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。

```
sagemaker_session = sage.Session()
bucket = sagemaker_session.default_bucket()
runtime = boto3.client("runtime.sagemaker")
s3 = boto3.resource("s3")

# Create a prefix called sampledata.
prefix = "sampledata"

# Create a filename called rawdata.csv
filename = "rawdata.csv"
```

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

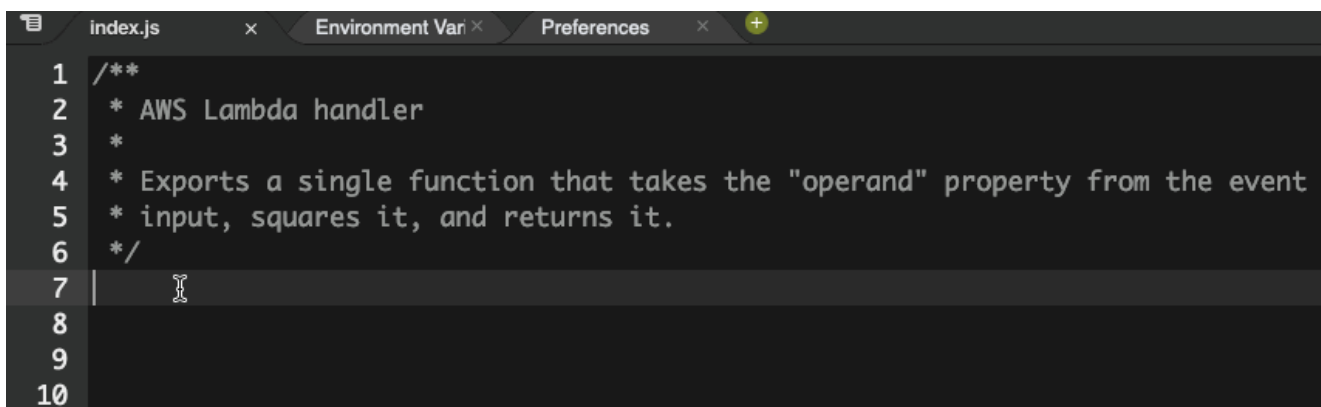
```
32 |  
33 |  
34 | bool AwsDoc::SQS::createQueue(const Aws::String &queueName,  
    |                               const Aws::Client::ClientConfiguration &clientConfigurat
```

C++

```
32 |  
33 |  
34 | bool AwsDoc::SQS::createQueue(const Aws::String &queueName,  
    |                               const Aws::Client::ClientConfiguration &clientConfigurat
```

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。



```
1 /**  
2  * AWS Lambda handler  
3  *  
4  * Exports a single function that takes the "operand" property from the event  
5  * input, squares it, and returns it.  
6  */  
7 |  
8  
9  
10
```


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。

```
def read_from_s3(bucket, key):  
    import boto3  
    s3 = boto3.client('s3')  
    obj = s3.get_object(Bucket=bucket, Key=key)  
    return obj['Body'].read().decode('utf-8')
```

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。

```
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)
```

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 は既にか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):
```

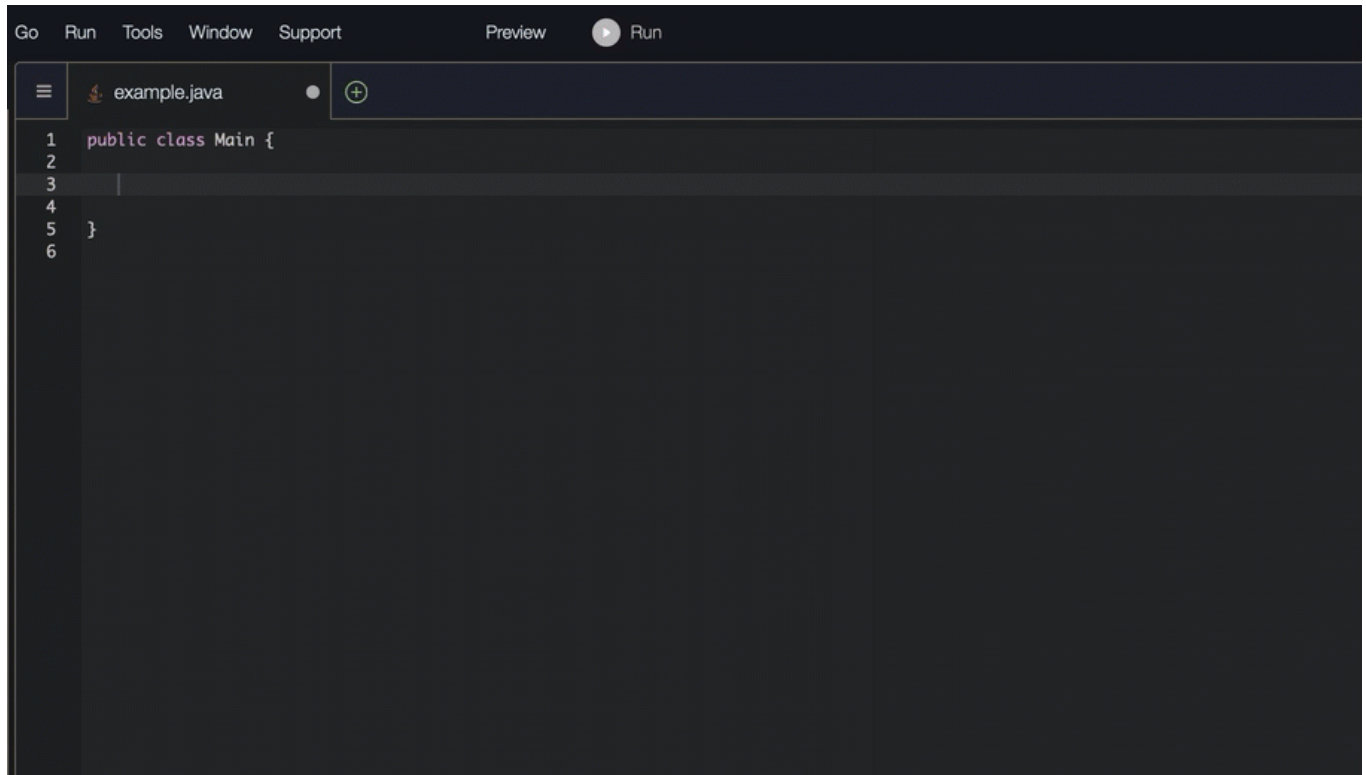
```
    while l <= r:
        mid = l + (r - l) // 2
        if arr[mid] == x:
            return mid
        elif arr[mid] < x:
            l = mid + 1
        else:
            r = mid - 1
```

Java

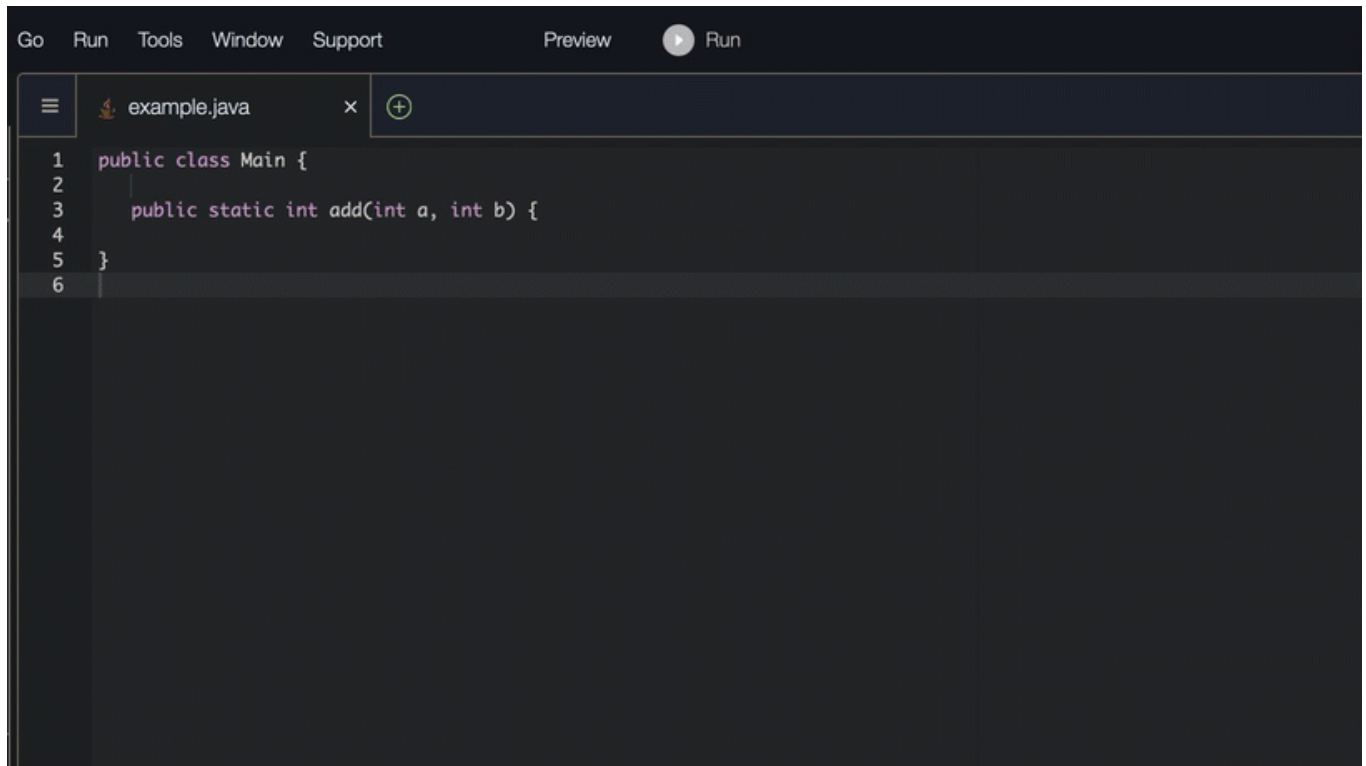
以下のリストでは、Amazon Q がどのように提案を行い、関数を作成するプロセス全体を進めるかについての例を挙げます。

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。

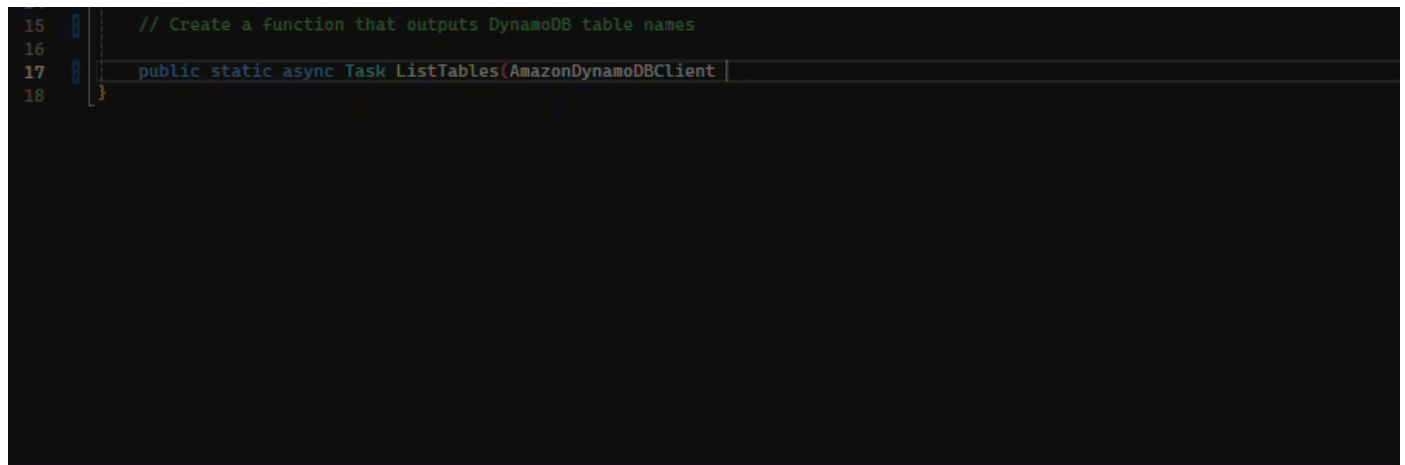


2. 次の図では、Amazon Q からの提案を受け入れる前に、ユーザーが関数の本文にコメントを入力します。次の行で、Amazon Q がコメントに基づいて提案を生成します。



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。

```
/**
 * Upload a large file to an S3 bucket in multiple parts.
 * @param {string} fileName -- The name of the file to upload.
 * @param {string} bucketName -- The name of the bucket to upload to.
 */
```

Python

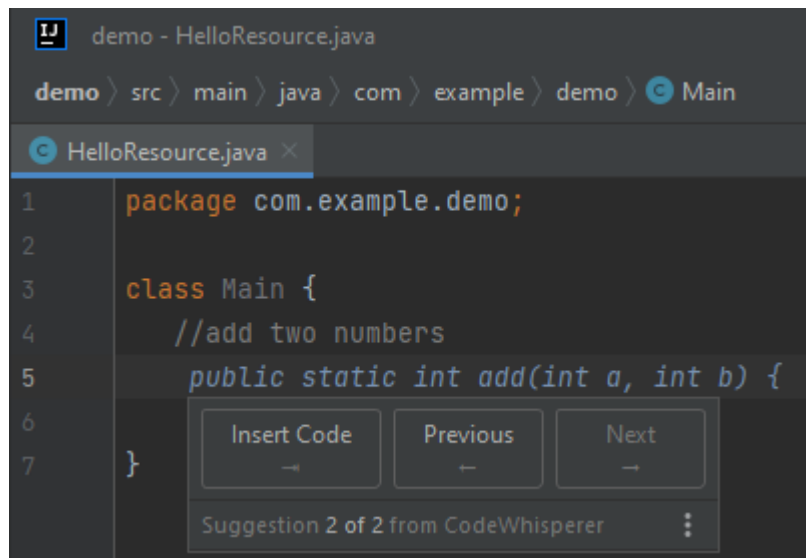
Amazon Q は、入力されたコメントに基づいて関数全体を生成できます。コメントを完成させると、Amazon Q は関数の署名を提案します。提案を受け入れると、Amazon Q は自動的にカーソルを関数の次の部分に進めて提案を行います。提案の合間であっても、追加のコメントやコード行を入力すると、Amazon Q は入力に基づいてリファクタリングします。

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。

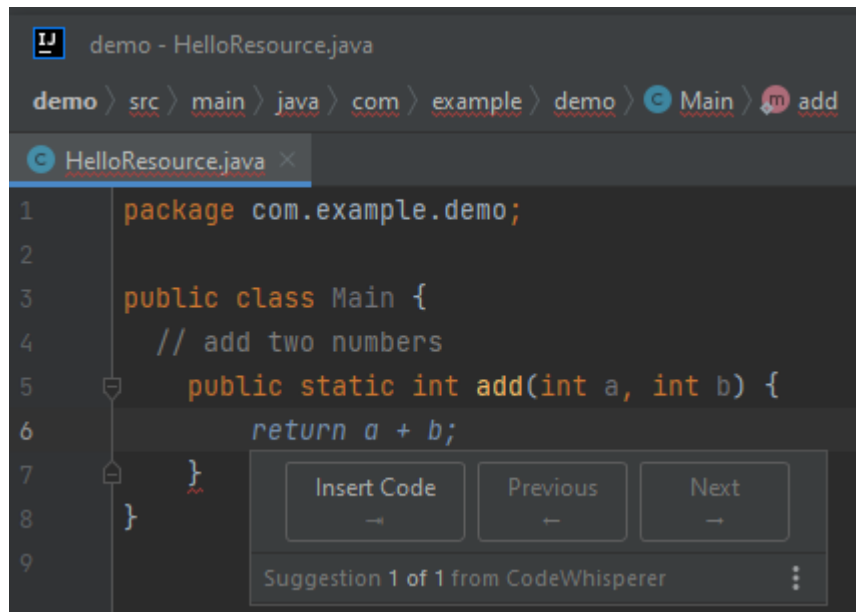
```
1 import boto3
2 ddb_client = boto3.client('dynamodb')
3
```

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。

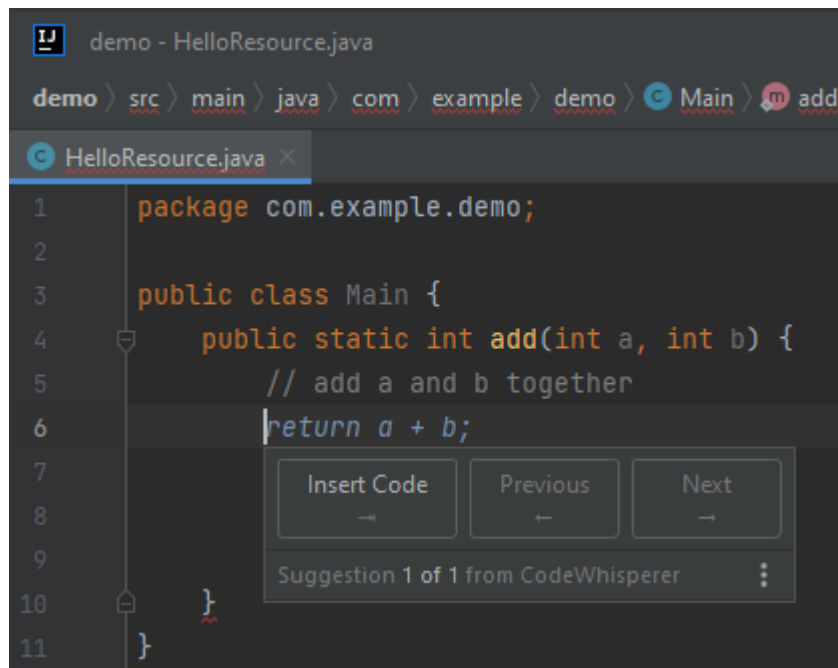
1. 次の画像では、ユーザーがコメントを入力します。コメントの下にある関数署名は、Amazon Q からの提案です。



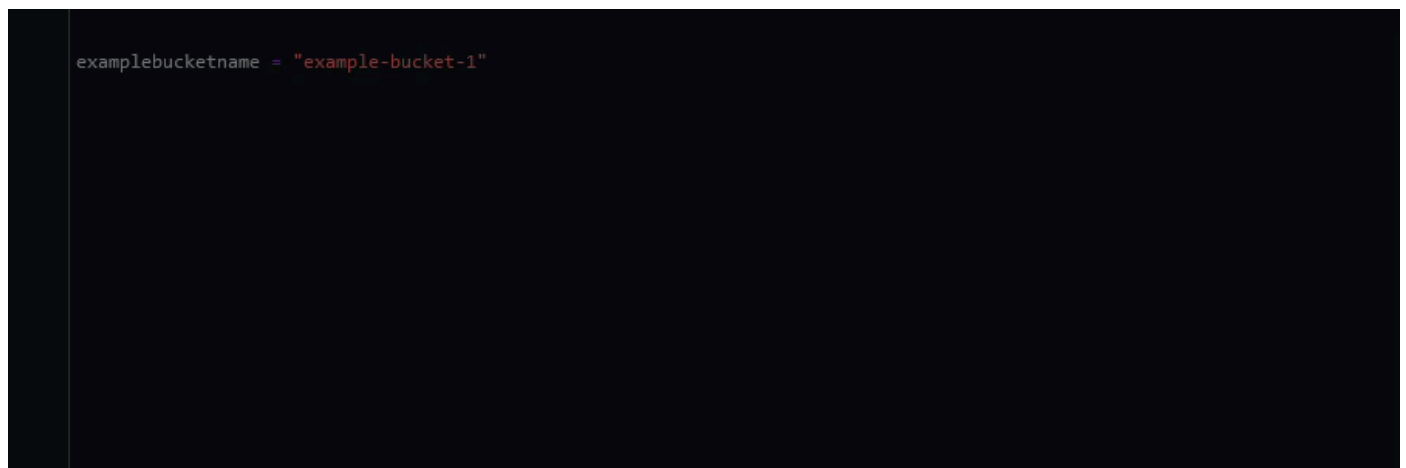
2. 次の画像では、ユーザーが関数署名の Amazon Q 提案を受け入れています。提案を受け入れるとカーソルが自動的に進み、Amazon Q が関数本文について新しい提案を行います。



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

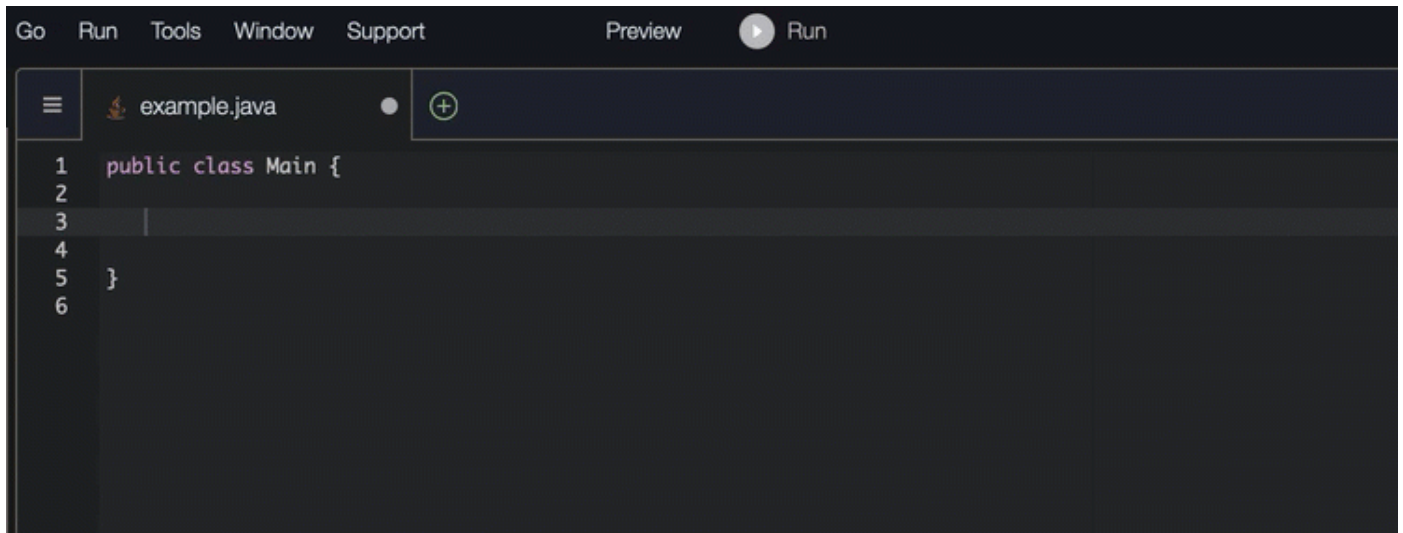
```
9
10 // function to pop the stack
11 int pop(Stack *stack) {
12     if (stack->top == -1) {
13         printf("Stack is empty\n");
14         return -1;
15     }
16     return stack->array[stack->top--];
17 }
18
19 // function to push the stack
20 void push(Stack *stack, int data) {
21     //
22 }
```

C++

```
33
34 bool AwsDoc::RDS::describeDBInstance(const Aws::String &dbInstanceIdentifier,
35                                       Aws::RDS::Model::DBInstance &instanceResult,
36                                       const Aws::RDS::RDSClient &client) {
37     //
38 }
39
```

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。

```
TS index.ts 2 X
TS index.ts > [e] uploadFile
1  import { S3Client } from "@aws-sdk/client-s3";
2
3  const client = new S3Client({});
4
5  /**
6   * Upload local file to bucket
7   */
8  export const uploadFile = async (
```

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。

```
examplebucketname = "example-bucket-1"

def print_bucket_contents(bucket_name):
    """
    Print the contents of a bucket.
    """
    print(f"Printing bucket contents for bucket {bucket_name}")
    for obj in s3.Bucket(bucket_name).objects.all():
        print(obj)
```

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++

```

7  /// <summary>
8  /// This example shows how to attach a policy to an IAM role.
9  /// </summary>
10 /// <param name="roleName">
11 bool AwsDoc::IAM::putRolePolicy(
12     const Aws::String &roleName,
13     const Aws::String &policyName,
14     const Aws::String &policyDocument,
15     const Aws::Client::ClientConfiguration &clientConfig) {
16     Aws::IAM::IAMClient iamClient(clientConfig);
17     Aws::IAM::Model::PutRolePolicyRequest request;
18
19     request.SetRoleName(roleName);
20     request.SetPolicyName(policyName);
21     request.SetPolicyDocument(policyDocument);
22
23     Aws::IAM::Model::PutRolePolicyOutcome outcome = iamClient.PutRolePolicy(request);
24     if (!outcome.IsSuccess()) {
25         std::cerr << "Error putting policy on role. " <<
26             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。

```

1  import {PutObjectCommand, S3Client} from "@aws-sdk/client-s3";
2
3  const client = new S3Client({});
4
5  /**
6   *
7   */
8  export const putObject = async (bucketName, key, body) => {
9      const params = {
10         Bucket: bucketName,
11         Key: key,
12         Body: body,
13     };
14     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。

```

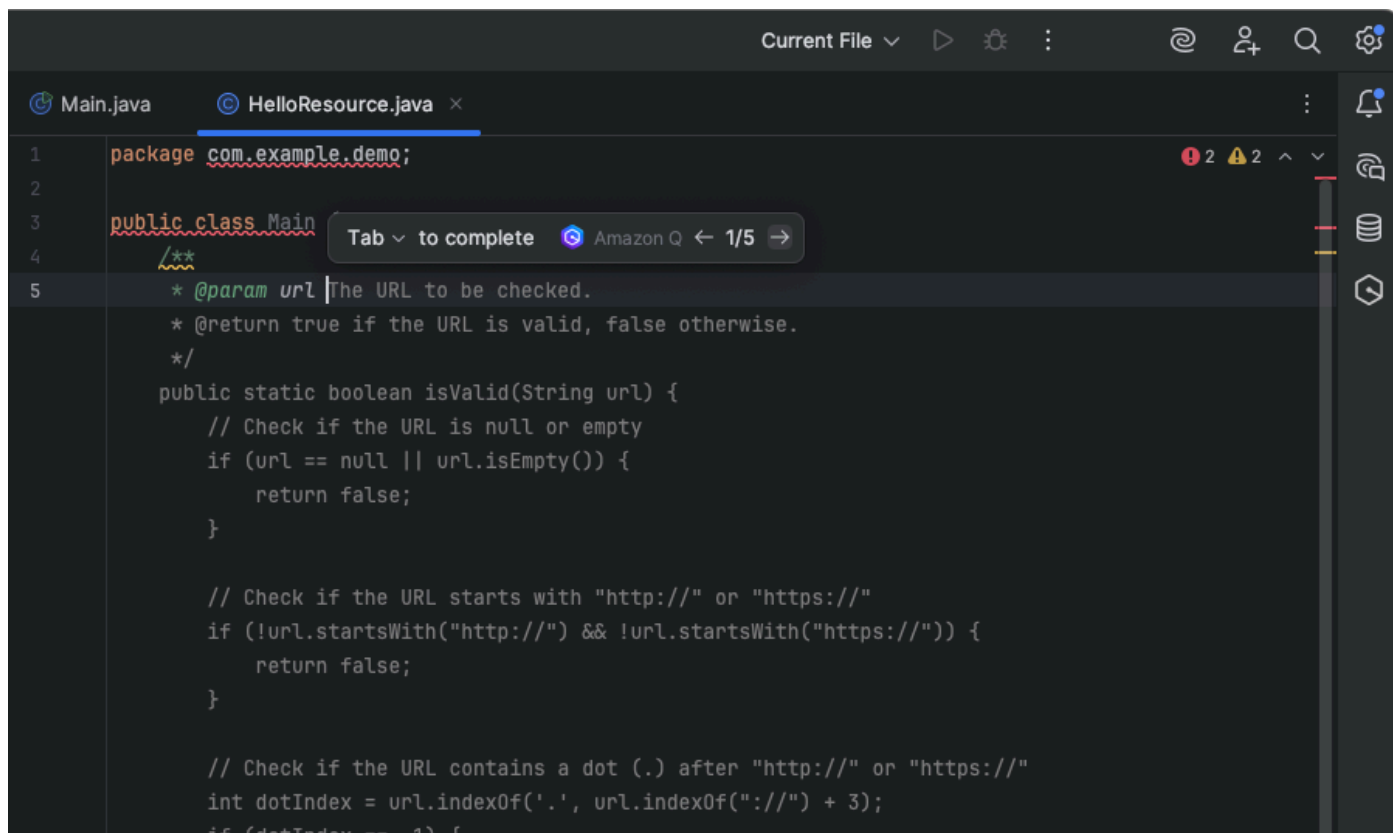
6  |  /// <summary>
7  |  /// Shows how to create a new Amazon S3 bucket.
8  |  /// </summary>
9  |  public static async Task<bool> CreateBucketAsync(IAmazonS3 client, string bucketName)
10 |  {
11 |      try
12 |      {
13 |          var request = new PutBucketRequest
14 |          {
15 |              BucketName = bucketName,
16 |              UseClientRegion = true,
17 |          };
18 |
19 |          var response = await client.PutBucketAsync(request);
20 |          return response.HttpStatusCode == System.Net.HttpStatusCode.OK;
21 |      }
22 |      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。

下の画像では、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。



The screenshot shows an IDE window with two tabs: 'Main.java' and 'HelloResource.java'. The 'Main.java' tab is active, displaying the following code:

```

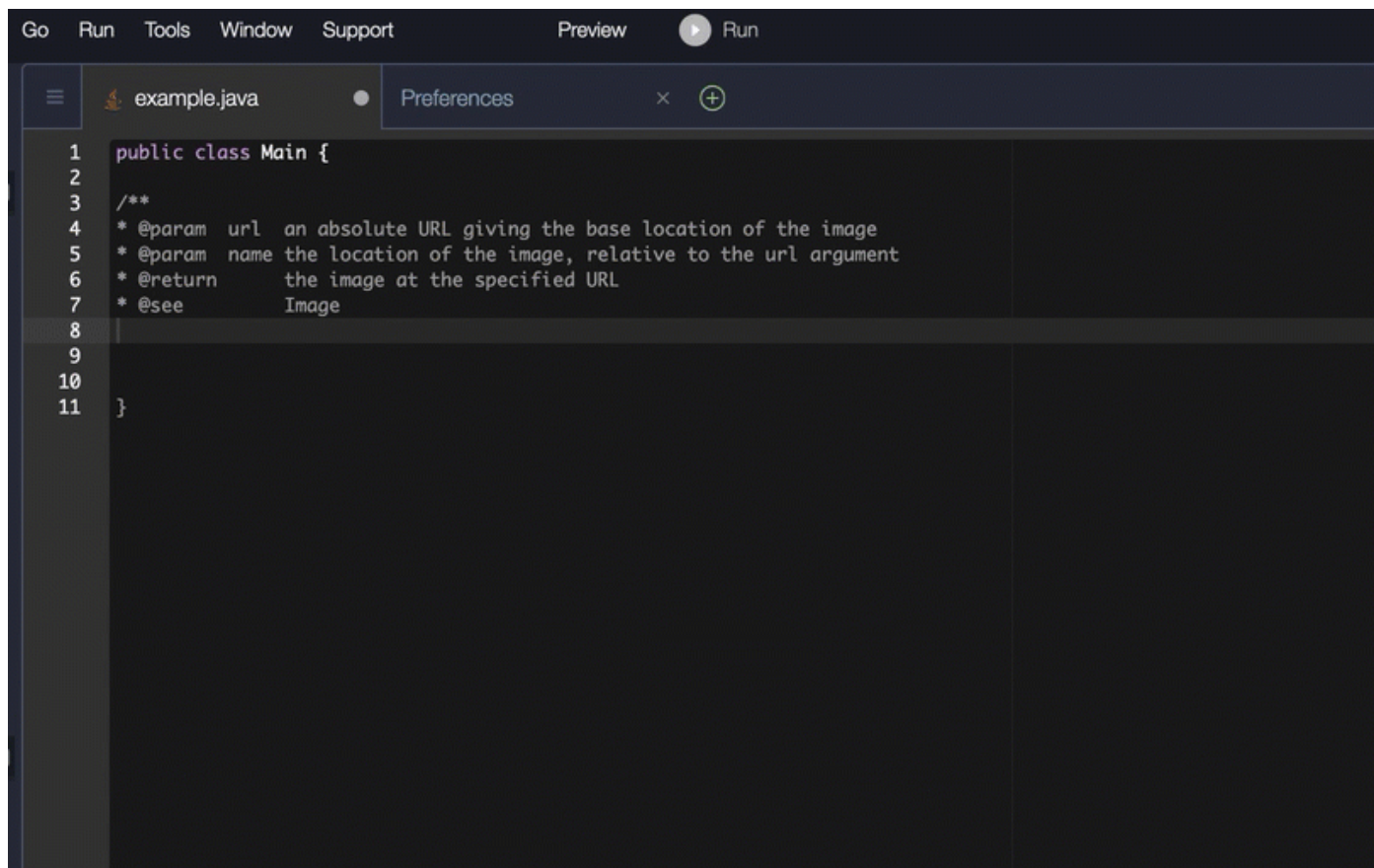
1  package com.example.demo;
2
3  public class Main
4  {
5      /**
6       * @param url The URL to be checked.
7       * @return true if the URL is valid, false otherwise.
8       */
9      public static boolean isValid(String url) {
10         // Check if the URL is null or empty
11         if (url == null || url.isEmpty()) {
12             return false;
13         }
14
15         // Check if the URL starts with "http://" or "https://"
16         if (!url.startsWith("http://") && !url.startsWith("https://")) {
17             return false;
18         }
19
20         // Check if the URL contains a dot (.) after "http://" or "https://"
21         int dotIndex = url.indexOf('.', url.indexOf("://") + 3);
22         if (dotIndex == -1) {

```

At line 5, the user has entered the docstring comment `/**`. An Amazon Q suggestion is shown in a tooltip above the cursor, displaying `Tab to complete` and `Amazon Q ← 1/5 →`. The suggestion list includes the text `* @param url The URL to be checked.`, which is currently selected.

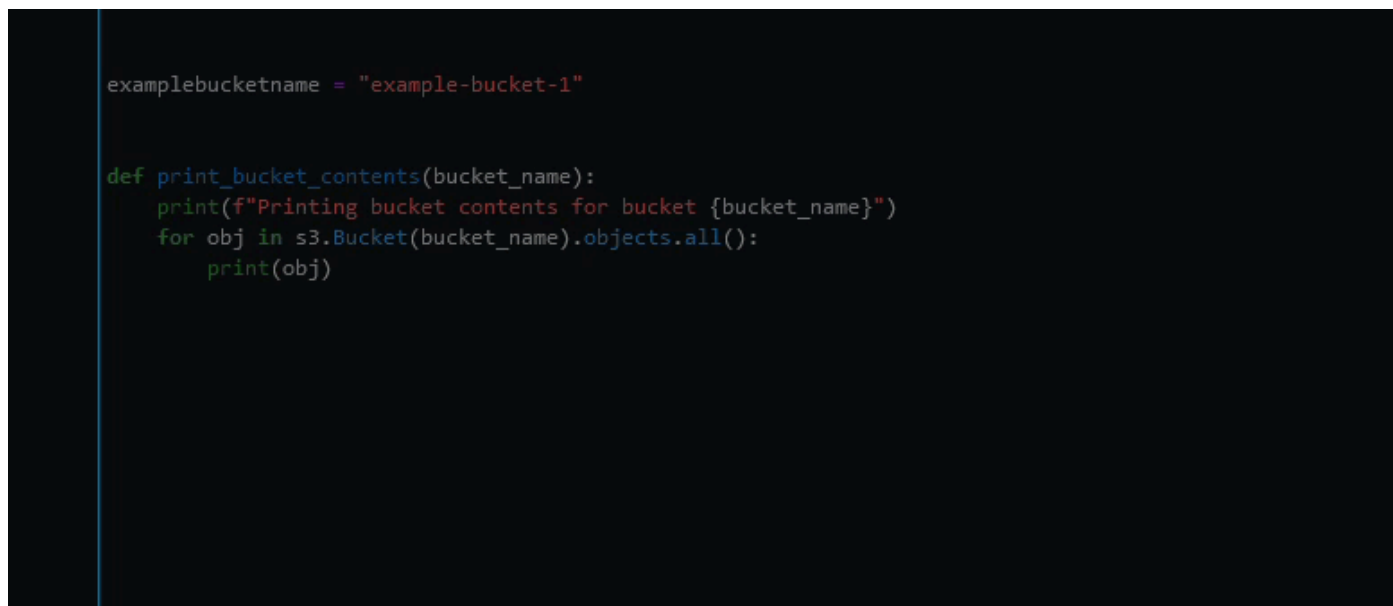
以下の例は、[Oracle のウェブサイトの例](#)から引用しています。

以下の Java 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。

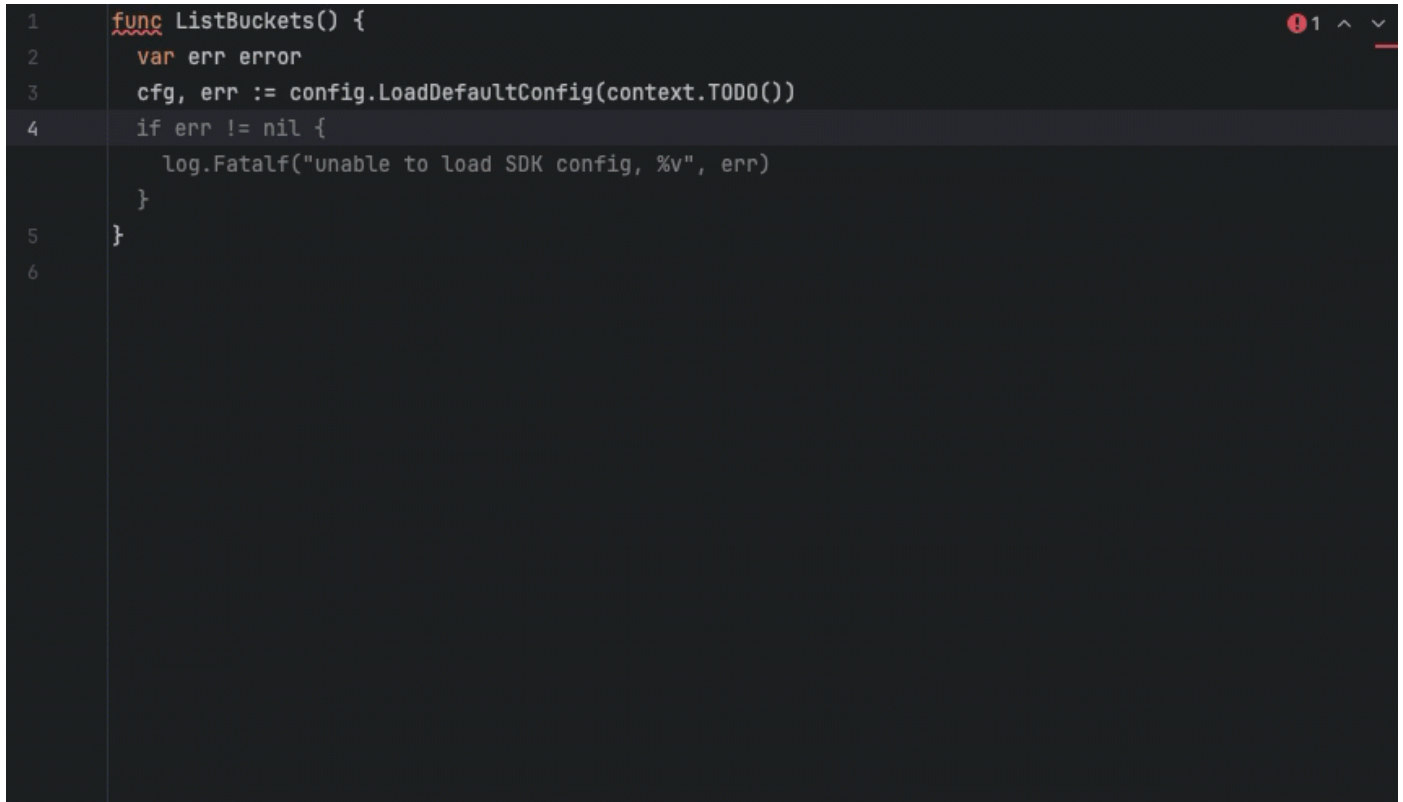


Amazon Q Developer を使用して行ごとの提案を行う

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。



```
1 func ListBuckets() {
2     var err error
3     cfg, err := config.LoadDefaultConfig(context.TODO())
4     if err != nil {
5         log.Fatalf("unable to load SDK config, %v", err)
6     }
}
```

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。


```
1  import "testing"
2
3  func Add(a, b int) int {
4      return a + b
5  }
6  func TestAdd(t *testing.T) {
    if Add(1, 2) != 3 {
        t.Error("1 + 2 did not equal 3")
    }
}
```

C++ and CLion

この例では、Amazon Q は行ごとの提案を行います。

```
1  bool CreateBucket(const Aws::String& bucket_name,
2                      const Aws::Client::ClientConfiguration &clientConfig) {
3      }
4
5
6
```

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 も既にな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。

```
role = get_execution_role()

sagemaker_session = sage.Session()
bucket = sagemaker_session.default_bucket()
runtime = boto3.client("runtime.sagemaker")
s3 = boto3.resource("s3")
```

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によって IDE で使用できます。

- [Chat](#)
- [インラインチャット](#)

これらの機能を使用する際の出力の品質は、言語の人気度によって異なります。

IDE で使用できる Amazon Q のその他の機能については、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
- Shell
- 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

変換がサポートされる言語

変換がサポートされる言語は、コードを変換する環境によって異なります。

JetBrains IDE および Visual Studio Code の場合、Amazon Q は次の言語でコードを変換できます。

- [Java](#)
- [Oracle から PostgreSQL へのデータベース移行のための埋め込み SQL 変換](#)

Visual Studio の場合、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に必要なその他の前提条件について詳しくは、実行する変換の種類に該当するトピックを参照してください。

コードレビューの言語サポート

Amazon Q は、次の言語で記述されたファイルとプロジェクトについて[コードレビュー](#)を作成し、自動コード修正を提示できます。

- 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
 - CloudFormation - 2010-09-09
 - Terraform - 1.6.2 以前
 - AWS CDK - TypeScript と Python

カスタマイズをサポートする言語

Amazon Q は、以下の言語で 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/シェルスクリプト (.sh、.zsh、.bash)
- C (.c、.h)
- C# (.cs)
- C++ (.cpp、.hpp、.h)
- Dart (.dart)
- Go (.go)
- HCL (.hcl)
- HTML (.html、.htm)
- Java (.java)
- JavaScript (.js、.jsx)
- JSON (json)
- Kotlin (.kt、.kts)
- Markdown (.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

Q CLI が Kiro CLI になりました。

詳細については、[Kiro ユーザーガイド](#)を参照してください。

Amazon Q Developer での MCP の使用

モデルコンテキストプロトコル (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、カスタムツールやサービスに接続して Q の機能を拡張できるようになりました。

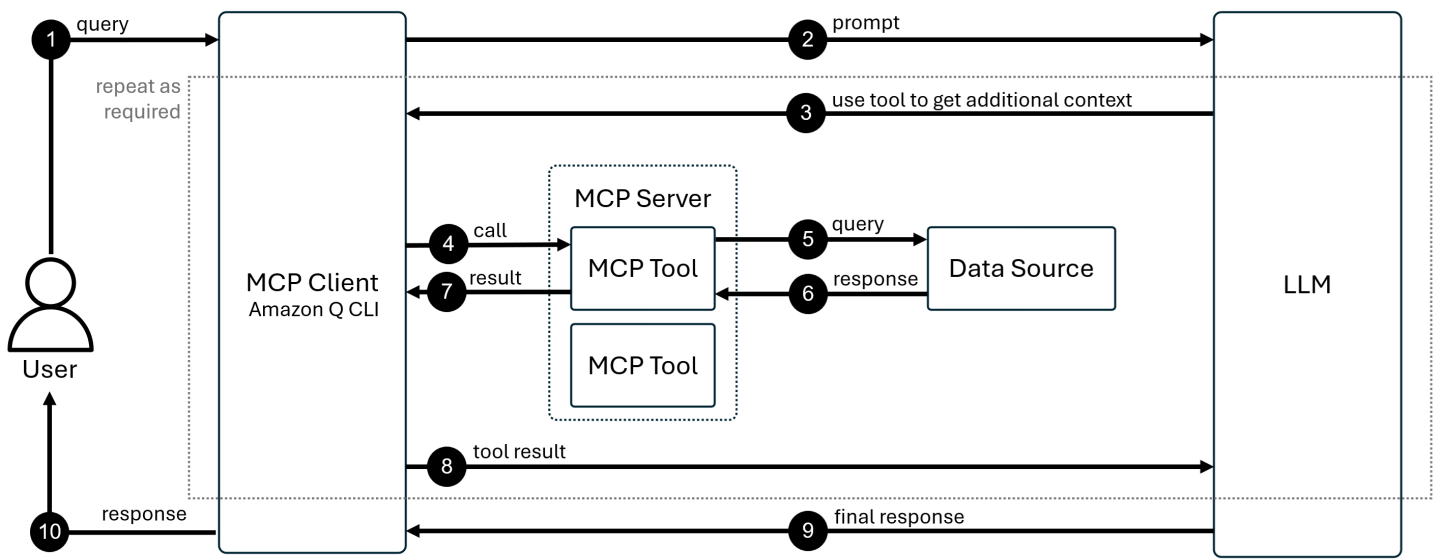
トピック

- [MCP の概要](#)
- [CLI での MCP 設定](#)
- [MCP の設定](#)
- [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](#)
- [Q Developer の MCP ガバナンス](#)

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て、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

使用方法: `qchat mcp [OPTIONS] COMMAND`

MCP 設定コマンド

コマンド	説明
<code>qchat mcp add</code>	設定済みのサーバーを追加または置き換える
<code>qchat mcp remove</code>	MCP 設定からサーバーを削除する
<code>qchat mcp list</code>	設定済みサーバーを一覧表示する
<code>qchat mcp import</code>	別のファイルからサーバー設定をインポートする
<code>qchat mcp status</code>	設定済みサーバーのステータスを取得する
<code>qchat mcp help</code>	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 サーバーの引数

--args パラメータは、エスケープ形式または JSON 配列形式を使用したカンマを含む引数をサポートするようになりました。

```
# Escaped commas
q mcp add --name server --command cmd --args "arg1,arg2\,with\,commas,arg3"

# JSON array format
q mcp add --name server --command cmd --args '["arg1", "arg2,with,commas", "arg3"]'
```

リモート MCP サーバー

Amazon Q Developer CLI は、プロセスとして実行されるローカル MCP サーバーに加えて、HTTP 経由で通信するリモート MCP サーバーをサポートしています。リモートサーバーは、OAuth 認証を使用するか、開くことができます (認証は必要ありません)。

構成

リモート MCP サーバーは、type および url フィールドを使用してエージェント設定ファイルで設定されます。

```
{
  "mcpServers": {
    "find-a-domain": {
      "type": "http",
      "url": "https://api.findadomain.dev/mcp"
    }
  }
}
```

OAuth 認証フロー

OAuth 認証を必要とするリモート MCP サーバーを使用する場合:

1. リモート MCP サーバーを含むエージェントで Q CLI セッションを開始する
2. サーバーは最初、「まだロードされていません」と表示されます。
3. /mcp コマンドを使用して認証を開始する
4. Q CLI は、サーバーに認証が必要であることを示し、URL を提供します。

5. Q CLI セッションを開いたまま、ブラウザで指定された URL を開きます。
6. ブラウザの認証手順に従います。
7. Q CLI ウィンドウに戻る - 認証が成功すると MCP サーバーにサインインします

認証が完了すると、サーバーのツールが利用可能になります。

MCP の設定

Q CLI を使用した MCP サーバーの設定

Amazon Q CLI のグローバルに定義された MCP 設定は、次の場所で処理されます。

```
~/.aws/amazonq/cli-agents
```

Amazon Q Developer CLI は、ローカル MCP サーバー (プロセスとして実行される) とリモート MCP サーバー (HTTP 経由で通信する) の両方をサポートします。リモートサーバーは OAuth 認証を使用するか、認証なしで開くことができます。

詳細については、[Q CLI Github リポジトリのカスタムエージェント設定ガイド](#)と [リモート MCP サーバー](#) を参照してください。

IDE で Q を使用して MCP サーバーを設定する

IDE の Amazon Q のグローバルに定義された MCP 設定は、次の場所で処理されます。

```
~/.aws/amazonq/agents/default.json
```

詳細については、「[IDE での Q Developer の MCP 設定](#)」を参照してください。

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、ツールが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]
```

[value] がタイムアウトする場所 (ミリ秒単位)。この設定は、Amazon Q がサーバーの初期化を待ってからインタラクションを開始するまで待機する時間を制御します。

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 の各ツールには以下のものが含まれます。

- 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み取れる説明
- 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: 何をするかを記述するだけで、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する: `/prompts`
- プロンプトを使用する:
 - @ *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。

IDE での Q Developer の MCP 設定ファイルについて

GUI を使用して IDE の Q Developer に MCP サーバーを追加すると、設定は次の 2 つのファイルのいずれかに保存されます。

- グローバルスコープ: `~/.aws/amazonq/default.json`

- ローカルスコープ: `.amazonq/default.json`

ただし、レガシーの理由から、他の 2 つの場所に MCP 設定情報を配置することもできます。

- グローバルスコープ: `~/.aws/amazonq/mcp.json`
- ローカルスコープ: `.amazonq/mcp.json`

Q Developer は、MCP サーバーのワークスペースレベルの設定、そのアクセス許可、および保存されている設定を優先します。

Note

mcp.json ファイルで MCP 設定をすでに設定していて、初めて MCP 設定 GUI を使用している場合は、その設定が GUI に表示されます。

レガシー mcp.json ファイルのサポートは、グローバル default.json 設定ファイルの `useLegacyMcpJson` フィールドによって有効になります。デフォルトでは、このフィールドは `true` に設定されます。詳細については、Q Developer CLI GitHub リポジトリの「[UseLegacyMcpJson フィールド](#)」を参照してください。

mcp.json ファイルは Q CLI でも使用できることに注意してください。

MCP ツールで詳細なコントロールを設定する方法については、「[組み込みツールリファレンス](#)」を参照してください。

MCP 設定 UI へのアクセス

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

AI クライアントと MCP サーバー間の通信には、STDIO と HTTP の 2 つの主要なトランスポートメカニズムがあります。

HTTP MCP サーバーの追加

HTTP 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いずれか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/default.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/default.json` に保存されます。

4. [名前] フィールドに、MCP サーバーの名前を入力します。
5. トランスポートプロトコルとして `http` を選択します。
6. URL フィールドに、初期化時に MCP サーバーが呼び出す URL を入力します。
7. [ヘッダー - オプション] で、HTTP リクエストヘッダーとして送信する必要があるキーと値のペアを入力できます。
8. 必要に応じて [タイムアウト] 値を入力します。
9. [保存] を選択します。

設定パネルがツールのアクセス許可パネルに置き換えられます。

10. 「[ツールのアクセス許可の確認と調整](#)」の手順に従います。

Note

MCP HTTP エンドポイントに認可が必要な場合、Amazon Q で自動的にブラウザページが開き、Amazon Q に MCP サーバーへのアクセスすることを許可することができます。

STDIO MCP サーバーの追加

STDIO 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いずれか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/default.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/default.json` に保存されます。

4. [名前] 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](#) をインストールする場合、名前は **AWS DocMCPServer** になります。

5. トランスポートプロトコルとして `stdio` を選択します。
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 `uvx` です。これは `uv tool run` のエイリアスで、エフェメラル Python 環境を作成します。

7. [引数] 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] に **`FASTMCP_LOG_LEVEL`**、[値] に **`ERROR`** と入力します。

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。

設定パネルが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 サーバーからツールが正常に機能することは想定できません。

[設定の修正] を選択して MCP 設定画面に戻り、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、対象の 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 [有効にする] 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. 無効にするユーザーを選択します。
3. [設定の編集] 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. [設定の編集] の横にある 3 つのドットを選択します。
4. [MCP サーバーを削除する] を選択します。
5. プロンプトが表示されたら、削除を確定します。

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、削除を確定します。

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のいずれかを設定できます。
 - 要求: 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: 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ません。

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る
- 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

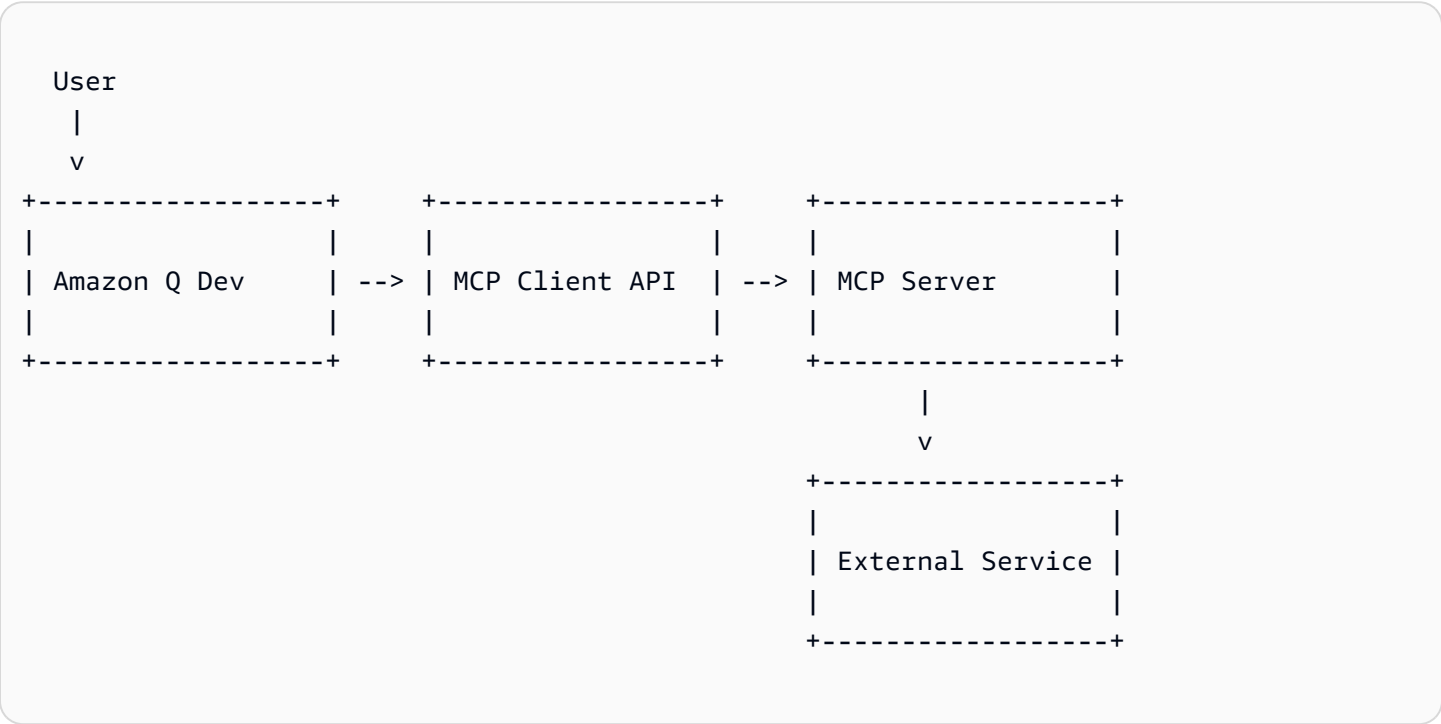
MCP アーキテクチャ

MCP は、以下においてクライアント/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 (API など) 経由で利用可能な外部システム

Example MCP コミュニケーションフロー



<caption>

MCP を介したユーザー、Amazon Q Developer CLI、外部サービス間のコミュニケーションフロー

</caption>

コア MCP の概念

ツール

ツールは、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
- API および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、入カスキーマ (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のチェーンを作成する
- 特定のワークフローをガイドする
- UI 要素として表示する (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。

- ファイルコンテンツ
- 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、以下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で実行されます
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 ログを監視する

Q Developer の MCP ガバナンス

サインイン方法として IAM アイデンティティセンターを使用するプロティアのお客様は、組織内のユーザーの MCP アクセスを制御できます。デフォルトでは、ユーザーは Q クライアントで任意の MCP サーバーを使用できます。管理者は、ユーザーによる MCP サーバーの使用を完全に無効にするか、ユーザーが使用できる MCP サーバーの詳細なリストを指定できます。

これらの制限は、MCP のオン/オフ切り替えと MCP レジストリを使用して制御します。MCP トグル属性とレジストリ属性は [Q Developer プロファイル](#) の一部であり、組織レベルまたはアカウントレベルで定義でき、アカウントレベルのプロファイルは組織レベルのプロファイルよりも優先されます。組織のデフォルトの MCP ポリシーを指定し、特定のアカウントで上書きできます。たとえば、組織の MCP は無効にしますが、特定のチーム (アカウント) の許可リストで有効にします。

 Note

トグルとレジストリ設定の両方がクライアント側に適用されます。エンドユーザーが回避できることに注意してください。

組織の MCP を無効にする

アカウントまたは組織の MCP を無効にするには:

1. Kiro コンソールを開きます。
2. [設定] を選択します。
3. Q Developer タブを選択します。
4. モデルコンテキストプロトコル (MCP) をオフに切り替えます。

組織の MCP 許可リストの指定

ユーザーがアクセスできる MCP サーバーを制御するには、許可されたサーバーで JSON ファイルを作成し、HTTPS 経由で提供し、Q Developer プロファイルに URL を追加します。このプロファイルを使用する Q Developer クライアントでは、ユーザーは許可リスト内の MCP サーバーにのみアクセスできます。

MCP レジストリ URL の指定

1. Kiro コンソールを開きます。
2. [設定] を選択します。
3. Q Developer タブを選択します。
4. Model Context Protocol (MCP) がオンになっていることを確認します。
5. MCP レジストリ URL フィールドで、編集 を選択します。
6. 許可リストに登録された MCP サーバーを含む MCP レジストリ JSON ファイルの URL を入力します。
7. [保存] を選択します。

MCP レジストリ URL は、[当社のデータ暗号化ポリシーに従って、転送中および保管中に暗号化](#)されます。

MCP レジストリファイル形式

レジストリ JSON ファイルの形式は、[MCP レジストリ標準 v0.1](#) のサーバースキーマ JSON のサブセットです。Q Developer でサポートされているサブセットの JSON スキーマ定義は、このドキュメントの最後にある[レジストリスキーマ](#)セクションで確認できます。

次の例は、リモート (HTTP) とローカル (stdio) の両方の MCP サーバードefinitionを含む MCP レジストリファイルを示しています。

```
{
  "servers": [
    {
      "server": {
        "name": "my-remote-server",
        "title": "My server",
        "description": "My server description",
        "version": "1.0.0",
        "remotes": [
          {
            "type": "streamable-http",
            "url": "https://acme.com/my-server",
            "headers": [
              {
                "name": "X-My-Header",
                "value": "SomeValue"
              }
            ]
          }
        ]
      }
    },
    {
      "server": {
        "name": "my-local-server",
        "title": "My server",
        "description": "My server description",
        "version": "1.0.0",
        "packages": [
          {
            "registryType": "npm",
            "registryBaseUrl": "https://npm.acme.com",
            "identifier": "@acme/my-server",
            "transport": {
```

```
        "type": "stdio"
    },
    "runtimeArguments": [
        {
            "type": "positional",
            "value": "-q"
        }
    ],
    "packageArguments": [
        {
            "type": "positional",
            "value": "start"
        }
    ],
    "environmentVariables": [
        {
            "name": "ENV_VAR",
            "value": "ENV_VAR_VALUE"
        }
    ]
}
]
}
}
]
}
}
```

次の表に、レジストリ JSON ファイルのプロパティを示します。特に明記されていない限り、すべてのプロパティは必須です。完全な JSON [スキーマについては、レジストリスキーマセクション](#)を参照してください。

ネストされた属性は、親からインデントされて表示されます。たとえば、「ヘッダー」は「リモート」の子属性で、「名前」と「値」は「ヘッダー」の子属性です。

属性	説明	オプション?	値の例
一般的な属性			

属性	説明	オプション?	値の例
名前	サーバー名。特定のレジストリファイル内で一意である必要があります。		「aws-ccapi-mcp」
title	人間が読めるサーバー名。	はい	「AWS CC API」
説明	サーバーの説明。		「自然言語による AWS infra の管理」
バージョン	サーバーのバージョン。セマンティックバージョニング (x.y.z) を強くお勧めします。		「1.0.2」
リモート (HTTP) サーバー属性			
リモート	リモートエンドポイントを指定するエントリが 1 つだけの配列。		-
型	「streamable-http」または「sse」のいずれかである必要があります。		「streamable-http」
url	MCP サーバーエンドポイント URL。		「https://mcp.figma.com/mcp」
ヘッダー	各リクエストに含める HTTP ヘッダーの配列。	はい	-
名前	HTTP ヘッダー名。		「認可」
値	HTTP ヘッダー値。		「ベアラー mF_9.B5f-4.1JqM」
ローカル (stdio) サーバー属性			

属性	説明	オプション?	値の例
パッケージ	MCP サーバー定義を含むエントリが 1 つだけの配列。		-
registryType	「npm」、「pypi」、または「oci」のいずれかである必要があります。 MCP サーバーパッケージをダウンロードして実行するには、次のパッケージランナーを使用します。 - レジストリタイプ「npm」の場合、「npx」ランナーが使用されます。 「pypi」の場合、「uvx」が使用されます 「oci」には「docker」が使用されます クライアントマシンには、適切なパッケージランナーがプリインストールされている必要があります。		「npm」
registryBaseUrl	パッケージレジストリ URL。	はい	「https://npm.acme.com」
識別子	サーバーパッケージ識別子。		「@acme/my-server」
トランスポート	プロパティ「type」が 1 つだけあるオブジェクト。		-
型	「stdio」である必要があります。		「stdio」

属性	説明	オプション?	値の例
runtimeArguments	ランタイム、つまり npx、uvx、または docker に提供される引数の配列。	はい	-
型	「位置」である必要があります。		「位置」
値	ランタイム引数値。		「-q」
packageArguments	MCP サーバーに提供される引数の配列。	はい	-
型	「位置」である必要があります。		「位置」
値	パッケージ引数値。		「開始」
environmentVariables	サーバーを起動する前に設定する env vars の配列。	はい	-
名前	環境変数名。		「LOG_LEVEL」
値	環境変数の値。		「INFO」

MCP レジストリファイルの供給

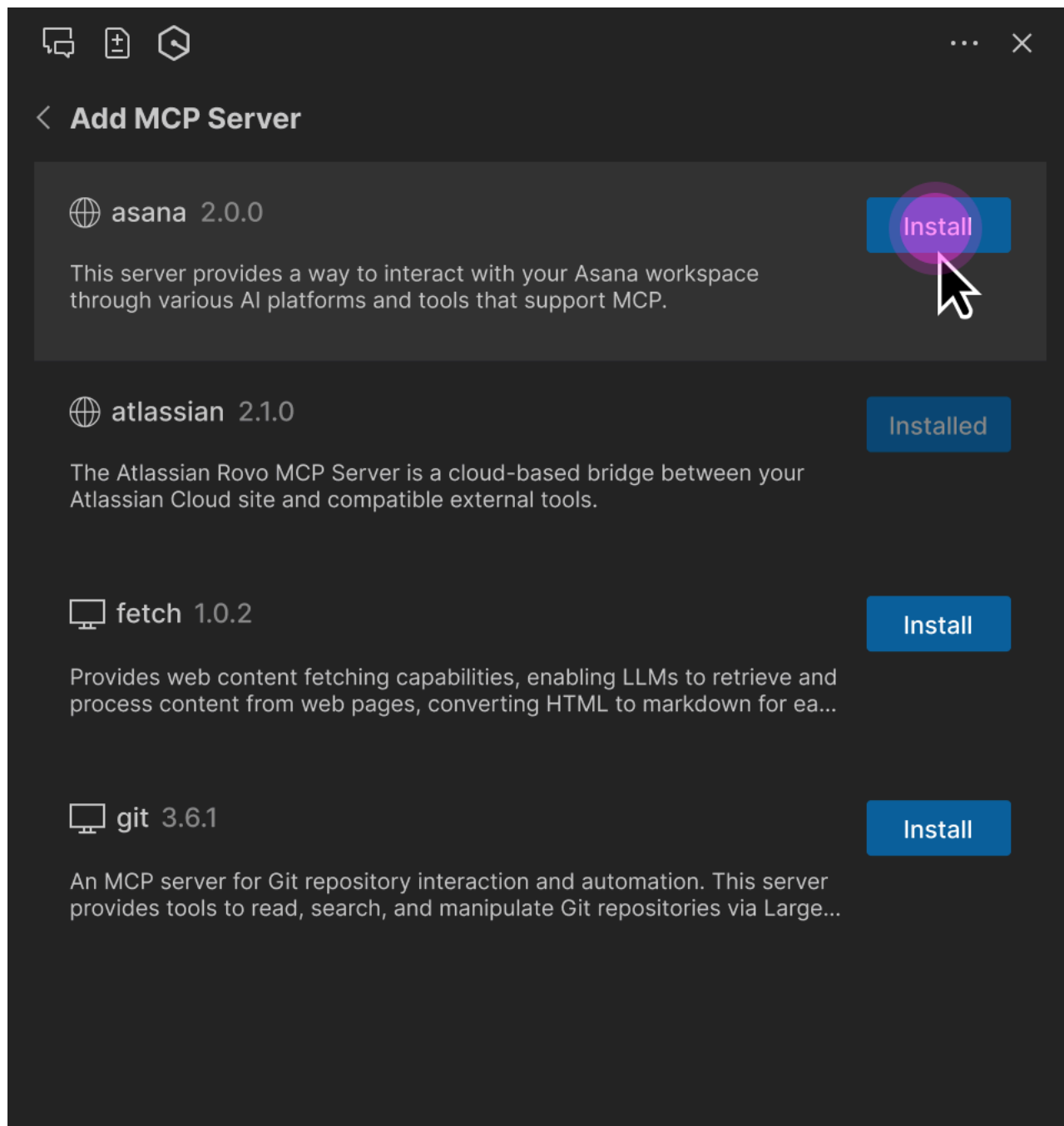
Amazon S3、Apache、nginx などのウェブサーバーを使用して、HTTPS 経由で MCP レジストリ JSON ファイルを提供します。URL は、ユーザーのコンピュータ上の Q Developer クライアントがアクセスできる必要がありますが、企業ネットワークに対してプライベートにすることができます。

HTTPS エンドポイントには、信頼できる認証機関によって署名された有効な SSL 証明書が必要です。自己署名証明書はサポートされていません。

Q Developer は、起動時と 24 時間ごとに MCP レジストリを取得します。定期的な同期中に、ローカルにインストールされた MCP サーバーがレジストリに存在しなくなった場合、Q Developer はそのサーバーを終了し、ユーザーがそれを追加しないようにします。ローカルにインストールされたサーバーのバージョンがレジストリ内のサーバーと異なる場合、Q Developer はレジストリで定義されたバージョンでサーバーを再起動します。

Q Developer プラグイン

ユーザーが Q Developer を起動すると、レジストリ URL がプロファイルで定義されているかどうかを確認します。その場合、その URL でレジストリ JSON を取得し、ユーザーがレジストリで定義された MCP サーバーのみを使用できるようにします。ユーザーが MCP サーバーを追加すると、Q Developer はレジストリのサーバーのリストを表示します。



レジストリ MCP サーバーパラメータ (URL、パッケージ識別子、runtimeArguments など) は読み取り専用です。ただし、ユーザーは次のことができます。

1. MCP ツールのアクセス許可を調整します (「実行をリクエストする」、「常に実行」、または「拒否」)。
2. MCP サーバースコープ (グローバルまたは Workspace) を選択します。
3. リクエストのタイムアウトを変更します。
4. ローカル MCP サーバーの追加の環境変数を指定します。
5. リモート MCP サーバーの追加の HTTP ヘッダーを指定します。

Note

ユーザー指定の環境変数または HTTP ヘッダーは、レジストリ定義を上書きします。これにより、ユーザーは認証キーやローカルフォルダパスなど、セットアップに固有の属性を指定できます。

MCP レジストリ JSON スキーマ

次の JSON スキーマは、Q Developer でサポートされている MCP レジストリファイル形式を定義します。このスキーマを使用して、作成したレジストリファイルを検証できます。

```
{
  "$schema": "https://json-schema.org/draft-07/schema",
  "properties": {
    "servers": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "server": {
            "$ref": "#/definitions/ServerDetail"
          }
        },
        "required": [
          "server"
        ]
      }
    }
  },
  "definitions": {
    "ServerDetail": {
```

```

"properties": {
  "name": {
    "description": "Server name. Must be unique within a given registry file.",
    "example": "weather-mcp",
    "maxLength": 200,
    "minLength": 3,
    "pattern": "^[a-zA-Z0-9._-]+$",
    "type": "string"
  },
  "title": {
    "description": "Optional human-readable title or display name for the MCP
server. MCP subregistries or clients MAY choose to use this for display purposes.",
    "example": "Weather API",
    "maxLength": 100,
    "minLength": 1,
    "type": "string"
  },
  "description": {
    "description": "Clear human-readable explanation of server functionality.
Should focus on capabilities, not implementation details.",
    "example": "MCP server providing weather data and forecasts via
OpenWeatherMap API",
    "maxLength": 100,
    "minLength": 1,
    "type": "string"
  },
  "version": {
    "description": "Version string for this server. SHOULD follow semantic
versioning (e.g., '1.0.2', '2.1.0-alpha'). Equivalent of Implementation.version in MCP
specification. Non-semantic versions are allowed but may not sort predictably. Version
ranges are rejected (e.g., '^1.2.3', '~1.2.3', '\\u003e=1.2.3', '1.x', '1.*').",
    "example": "1.0.2",
    "maxLength": 255,
    "type": "string"
  },
  "packages": {
    "items": {
      "$ref": "#/definitions/Package"
    },
    "type": "array"
  },
  "remotes": {
    "items": {
      "anyOf": [

```

```

        {
            "$ref": "#/definitions/StreamableHttpTransport"
        },
        {
            "$ref": "#/definitions/SseTransport"
        }
    ]
},
"type": "array"
}
},
"required": [
    "name",
    "description",
    "version"
],
"type": "object"
},
"Package": {
    "properties": {
        "registryType": {
            "description": "Registry type indicating how to download packages (e.g.,
'npm', 'pypi', 'oci')",
            "enum": [
                "npm",
                "pypi",
                "oci"
            ],
            "type": "string"
        },
        "registryBaseUrl": {
            "description": "Base URL of the package registry",
            "examples": [
                "https://registry.npmjs.org",
                "https://pypi.org",
                "https://docker.io"
            ],
            "format": "uri",
            "type": "string"
        },
        "identifier": {
            "description": "Package identifier - either a package name (for registries)
or URL (for direct downloads)",
            "examples": [

```

```

        "@modelcontextprotocol/server-brave-search",
        "https://github.com/example/releases/download/v1.0.0/package.mcpb"
    ],
    "type": "string"
},
"transport": {
    "anyOf": [
        {
            "$ref": "#/definitions/StdioTransport"
        },
        {
            "$ref": "#/definitions/StreamableHttpTransport"
        },
        {
            "$ref": "#/definitions/SseTransport"
        }
    ],
    "description": "Transport protocol configuration for the package"
},

"runtimeArguments": {
    "description": "A list of arguments to be passed to the package's runtime
command (such as docker or npx).",
    "items": {
        "$ref": "#/definitions/PositionalArgument"
    },
    "type": "array"
},
"packageArguments": {
    "description": "A list of arguments to be passed to the package's binary.",
    "items": {
        "$ref": "#/definitions/PositionalArgument"
    },
    "type": "array"
},
"environmentVariables": {
    "description": "A mapping of environment variables to be set when running the
package.",
    "items": {
        "$ref": "#/definitions/KeyValueInput"
    },
    "type": "array"
}
},

```

```

    "required": [
      "registryType",
      "identifier",
      "transport"
    ],
    "type": "object"
  },
  "StdioTransport": {
    "properties": {
      "type": {
        "description": "Transport type",
        "enum": [
          "stdio"
        ],
        "example": "stdio",
        "type": "string"
      }
    },
    "required": [
      "type"
    ],
    "type": "object"
  },
  "StreamableHttpTransport": {
    "properties": {
      "type": {
        "description": "Transport type",
        "enum": [
          "streamable-http"
        ],
        "example": "streamable-http",
        "type": "string"
      },
      "url": {
        "description": "URL template for the streamable-http transport. Variables in {curly_braces} reference argument valueHints, argument names, or environment variable names. After variable substitution, this should produce a valid URI.",
        "example": "https://api.example.com/mcp",
        "type": "string"
      }
    },
    "headers": {
      "description": "HTTP headers to include",
      "items": {
        "$ref": "#/definitions/KeyValueInput"
      }
    }
  }
}

```



```
    },
    "type": "array"
  }
},
"required": [
  "type",
  "url"
],
"type": "object"
},
"SseTransport": {
  "properties": {
    "type": {
      "description": "Transport type",
      "enum": [
        "sse"
      ],
      "example": "sse",
      "type": "string"
    },
    "url": {
      "description": "Server-Sent Events endpoint URL",
      "example": "https://mcp-fs.example.com/sse",
      "format": "uri",
      "type": "string"
    },
    "headers": {
      "description": "HTTP headers to include",
      "items": {
        "$ref": "#/definitions/KeyValueInput"
      },
      "type": "array"
    }
  },
  "required": [
    "type",
    "url"
  ],
  "type": "object"
},
"PositionalArgument": {
  "properties": {
    "type": {
      "enum": [
```

```
        "positional"
      ],
      "example": "positional",
      "type": "string"
    },
    "value": {
      "description": "The value for the input.",
      "type": "string"
    }
  },
  "required": [
    "type",
    "value"
  ],
  "type": "object"
},
"KeyValueInput": {
  "properties": {
    "name": {
      "description": "Name of the header or environment variable.",
      "example": "SOME_VARIABLE",
      "type": "string"
    },
    "value": {
      "description": "The value for the input.",
      "type": "string"
    }
  },
  "required": [
    "name"
  ],
  "type": "object"
}
},
"required": [
  "servers"
],
"type": "object"
}
```

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に統合することができ、専門的な 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。サポートされている統合には、GitLab Duo と GitHub が含まれ、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、ルーチンタスクを自動化し、コード品質を向上させることで、開発を合理化するのに役立ちます。

GitLab Duo with Amazon Q Developer

GitLab Duo with 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 ティア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 チャットも含まれており、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 with Amazon Q 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機能開発。
- 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。マージリクエストでの迅速なアクションを含む。
- マージリクエストの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。

使用を開始するには、「[Set up GitLab Duo with Amazon Q](#)」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とコードレビューが可能になります。Amazon Q Developer に GitHub 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反復処理し、共同開発ワークフローを作成します。

Amazon Q Developer は、GitHub で次の主な機能を提供します。

- 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。
- コマンドをスラッシュして、問題から機能開発を開始し、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、反復的な開発を行う。
- Amazon Q Developer に機能開発タスクを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: GitHub のインストール、機能の使用、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 を使用すると、GitLab または GitHub 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。これらのルールは、チームのコーディング標準とベストプラクティスを定義します (Python コードで型ヒントを要求する、Java コードで Javadoc コメントを要求するなど)。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開発者間で一貫性が確保され、開発者がプロジェクト内で Amazon Q Developer とやり取りする際に Amazon Q Developer のコンテキストに自動的に組み込まれます。これにより、生成されたすべての応答が確立されたガイドラインに従っていることが保証されます。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 with Amazon Q

[GitLab Duo Amazon Q では](#)、アイデアのコード実装の提案、品質と脆弱性に関するマージリクエストのレビュー、ユニットテストの提案など、人工知能 (AI) エクスペリエンスのスイートが提供されます。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 自己管理型サービスおよび Ultimate ティアサブスクリプション (GitLab Duo with Amazon Q サブスクリプションアドオン) で使用できます。詳細については、「GitLab documentation」の「[GitLab plans](#)」を参照してください。

GitLab Duo with Amazon Q 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「[GitLab Duo with Amazon Q の開始方法](#)」を参照してください。GitLab Duo with Amazon Q で使

用できるすべての機能については、「GitLab documentation」の「[Additional supported features](#)」を参照してください。

トピック

- [GitLab Duo の概念](#)
- [GitLab Duo with Amazon Q の開始方法](#)
- [GitLab Duo with Amazon Q に関する問題のトラブルシューティング](#)

GitLab Duo の概念

[GitLab Duo with Amazon Q](#) 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

- [GitLab Duo with Amazon Q のセットアップ](#)
- [AWS リソースとアクセス許可ポリシーを使用したオンボーディング](#)
- [GitLab のクイックアクション](#)

GitLab Duo with Amazon Q のセットアップ

GitLab Duo で Amazon Q の AI 機能を使用するには、前提条件を満たし、AWS リソースを作成する必要があります。詳細については、「GitLab documentation」の「[Set up GitLab Duo with Amazon Q](#)」を参照してください。

AWS リソースとアクセス許可ポリシーを使用したオンボーディング

GitLab Duo の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コントロール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、GitLab Duo with Amazon Q をセットアップする方法については、「GitLab documentation」の「[Set up GitLab Duo with Amazon Q](#)」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、GitLab Duo with Amazon Q 統合で機能を利用できるようになります。IAM ロールを作成する際は、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、「[GitLabDuoWithAmazonQPermissionsPolicy](#)」を参照してください。

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 でのポリシーとアクセス許可](#)」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/*"
  }
]
}

```

キーのライフサイクルと使用状況を完全にコントロールする場合は、オプションで、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。kms:ViaService 条件キーは、コンテンツの暗号化と復号化において、どのユーザーが CMK を使用できるかを制限します。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 のクイックアクション

クイックアクションが呼び出されると、GitLab の問題とマージリクエストでタスクを実行します。GitLab のクイックアクションを呼び出す方法については、「[GitLab documentation](#)」を参照してください。

マージリクエストの生成とイテレーション

- `/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、Amazon Q にレビュー準備完了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、「[Turn an idea into a merge request](#)」を参照してください。

コードレビュー

- `/q review` – GitLab Duo with Amazon Q を使用して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に対してコード修正を生成して提案する際に、潜在的な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、以前に閉じたリクエストを再度開くと開始されます。ただし、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自動コードレビューがトリガーされることはありません。`/q review` の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、「[Review a merge request](#)」を参照してください。

ウェブ UI と IDE のチャットセッション

- GitLab Duo チャットおよびコード提案は Amazon Q と連携して、CI/CD 構成、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、GitLab

Duo with Amazon Q チャット機能呼び出すことができます。詳細については、「[Ask GitLab Duo Chat](#)」を参照してください。

GitLab Duo with Amazon Q の開始方法

[GitLab Duo with Amazon Q](#) を使用すると、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。自己管理型 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 Duo with Amazon Q](#) の使用を開始することができます。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 リソースとアクセス許可ポリシーを使用したオンボーディング](#)」を参照してください。必要なリソースを作成し、GitLab Duo with Amazon Q をセットアップする方法については、「GitLab documentation」の「[Set up GitLab Duo with Amazon Q](#)」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でオンまたはオフにすることで、GitLab Duo with Amazon Q の可用性をいつでも設定できます。詳細については、「[Turn off GitLab Duo with Amazon Q](#)」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、品質と脆弱性に関するマージリクエストを確認し、ユニットテストを提案できます。また、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することもできます。

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 Duo with Amazon Q](#)」を参照してください。

GitLab Duo with Amazon Q に関する問題のトラブルシューティング

GitLab Duo with Amazon Q 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。これは、次のいずれかになります。

- 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 インスタンスに再度オンボードする必要があります。詳細については、「GitLab documentation」の「[Set up GitLab Duo with Amazon Q](#)」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: GitLab Duo with Amazon Q アプリケーションを変更した後、GitLab と Amazon Q 間の接続が同期しなくなった。

解決策: GitLab と Amazon Q が同期していない場合、Amazon Q が GitLab を呼び出す際に、認証情報が無効になったり、認証情報を更新できなかったり、GitLab からの不正な応答が発生したり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ない

問題: オンボーディング後に GitLab インスタンス URL を変更すると、GitLab と Amazon Q 間の接続が同期しなくなった。Amazon Q は GitLab インスタンスを正常に呼び出すことができない。

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 documentation」の「[Set up GitLab Duo with Amazon Q](#)」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 GitLab Duo with Amazon Q で必要な API に正しいアクセス許可を提供していない。

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 documentation」の「[Set up GitLab Duo with Amazon Q](#)」を参照してください。

Amazon Q Developer プロファイルが存在しない

問題: GitLab Duo with Amazon Q にオンボードしようとしているが、次の問題が発生している。Application could not be created by the AI Gateway: Error 400 - {"detail": "An error occurred (ValidationException) when calling the CreateOAuthAppConnection operation: ProfileDoesNotExist"}

解決策: 最初に 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 documentation」の「[Set up GitLab Duo with Amazon Q](#)」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で 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、新しいアイデアを実装し、品質の問題のコードを確認し、ユニットテストで脆弱性に対処できます。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のイテレーションを行います。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。

[GitHub](#) から Amazon Q Developer 統合にアクセスし、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 GitHub の使用を開始するには、「[クイックスタート: GitHub のインストール、機能の使用、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「[GitHub App をインストールするための要件](#)」および「[Organization のロール](#)」を参照してください。

Note

Amazon Q Developer と GitHub の統合は米国内でデータを処理します。詳細については、「[Cross-region processing in Amazon Q Developer](#)」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ユーザーのコンテンツを使用しません。AWS が今後これを有効にする場合は、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](#)
- [GitHub でのブラウザ拡張機能の使用](#)
- [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: GitHub のインストール、機能の使用、使用制限の引き上げ](#)
- [GitHub での Amazon Q Developer による機能開発とイテレーション](#)
- [Amazon Q Developer in GitHub を使用したコードレビュー](#)
- [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](#)
- [Amazon Q Developer for GitHub の問題のトラブルシューティング](#)

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、開始するために AWS アカウントを設定することなく、 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。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。インストールして承認すると、GitHub で Amazon Q Developer 機能を無料で使用できます。 [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、無料利用枠を増やすことができます。を増やすことができます。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: GitHub のインストール、機能の使用、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「[GitHub App をインストールするための要件](#)」および「[Organization のロール](#)」を参照してください。

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 IP アドレスを有効化する](#)」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、デプロイからトラブルシューティングまで、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 Developer エージェント – 問題を作成して機能開発ラベルを追加すると、Amazon Q Developer は新機能の実装とバグ修正を自動的行います。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、「[GitHub での Amazon Q Developer による機能開発とイテレーション](#)」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。これを確認してプルリクエストにコミットすることができます。コードレビューには、スレッド化された検出結果を含むコードレビューの概要が含まれます。Amazon Q Developer を操作するには、プルリクエストコメントで `/q` コマンドを使用して、コードレビューの結果について質問します。

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、「[Amazon Q Developer in GitHub を使用したコードレビュー](#)」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、または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

GitHub への Amazon Q Developer の統合は無料で利用できます。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。機能開発では 1 か月あたりの呼び出し数に制限があり、コードレビューでは 1 か月あたりの行数に制限があります。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、無料利用枠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 App をインストールするための要件](#)」および「[OAuth apps と Organization](#)」を参照してください。

GitHub でのブラウザ拡張機能の使用

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問題で機能開発用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 問題のコメントやプルリクエストでスラッシュコマンドを使用することで、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、新機能の実装とバグ修正を自動的に行います。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ます。コードレビューは、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とともに、修正案、スレッド化された検出結果を含むコードレビューの概要を提供します。プルリクエストコメントで `/q` を使用して、検出結果を操作します。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 help` - スラッシュコマンド、機能、カスタマイズの詳細に加えて、「Amazon Q Developer 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 GitHub に関する情報を提供します。

クイックスタート: GitHub のインストール、機能の使用、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、以下のタスクについて説明します。

1. GitHub Marketplace から Amazon Q Developer アプリをインストールして、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を開始することもできます。さらに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: GitHub に Amazon Q Developer をインストール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。GitHub で Amazon Q Developer を使用する最初のステップは、[GitHub](#) からアプリ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「[GitHub App をインストールするための要件](#)」および「[Organization のロール](#)」を参照してください。

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 アプリ](#) ページに移動します。
2. 必要に応じて、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。
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、[リポジトリのみを選択] を選択し、[リポジトリを選択] ドロップダウンを選択して、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、GitHub への Amazon Q Developer のインストールを完了し、リポジトリへのアクセスを承認します。

GitHub にアプリをインストールしてアクセスを許可すると、GitHub の Amazon Q Developer の概要ページにリダイレクトされます。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 IP アドレスを有効化する](#)」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: GitHub で Amazon Q Developer 機能を使用する

に 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許可したら、Amazon Q Developer エージェントを使用して、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、または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。

- [GitHub での Amazon Q Developer による機能開発とイテレーション](#)
- [Amazon Q Developer in GitHub を使用したコードレビュー](#)

ステップ 3: 無料使用枠を増やし、詳細を設定する

GitHub での Amazon Q Developer エージェントは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に、1 か月あたりの呼び出

しが制限されています。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。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 App をインストールするための要件](#)」および「[OAuth apps と Organization](#)」を参照してください。

GitHub での Amazon Q Developer による機能開発とイテレーション

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。

GitHub で Amazon Q Developer を使用すると、新機能の実装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、ユーザーはフィードバックを提供でき、Amazon Q Developer は推奨コードを繰り返し実行します。

Amazon Q Developer での機能開発の実行回数は、1 か月あたりの回数制限があります。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。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、または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、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。
2. GitHub 組織に移動し、Amazon Q Developer を使用して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「[Issue の作成](#)」を参照してください。
 - 新しい問題の場合、[タイトルの追加] テキスト入力フィールドで、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。

既存の問題の場合、問題のタイトルと説明を編集して、機能開発のために Amazon Q Developer にコンテキストを提供できます。詳細については、「GitHub ドキュメント」の「[issue 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 ラベルを適用するか、/q dev 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 への割り当て] ドロップダウンメニューを選択し、[Amazon Q Developer 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 Developer 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「/q dev」と入力します。
 2. [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「[issue 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 the proposed code changes, and the pull request is ready for review: [PR link]」) 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. [変更されたファイル] 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の問題がない場合、プルリクエストをマージできます。詳細については、「[pull request のマージ](#)」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のイテレーションを実行することもできます。

Amazon Q Developer 機能開発コードでイテレーションを実行するには

1. Amazon Q Developer によって作成されたプルリクエストを選択し、[変更されたファイル] タブを選択してコードの変更を表示します。
2. 必要に応じて、フィードバックを提供する特定のコード行について、+ を選択してフィードバックを含むコメントを追加します。

会話では、/q コマンドの後に自然言語 (/q implement my suggestionsや など) の手順を使用できます/q refactor this function for better performance。Amazon Q Developer は、フィードバックに基づいて行う変更を説明するコメントで応答します (たとえば、「フィードバックに基づいて次の変更を実装します: ...」)。実装が完了すると、Amazon Q Developer は、変更を確認する別のコメント (「提案された変更を実装しました」など) と、変更を表示できる生成されたコミットへのリンクを投稿します。

3. 会話で提供されたコミットリンクに従って、Amazon Q Developer によって行われた変更を確認します。必要に応じて、コマンドを使用して追加のフィードバックを提供し、/qさらに繰り返すを行うことができます。
4. 更新されたコードの変更の問題がない場合、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、「[pull request のマージ](#)」を参照してください。

Amazon Q Developer in GitHub を使用したコードレビュー

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 は、GitHub 内で自動コードレビューを有効化します。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。各レビューには、スレッド化された検出結果を含むコードレビューの概要が含まれます。Amazon Q Developer は、特定された問題の修正も生成します。これを確認してプルリクエストにコミットすることができます。プルリクエストコメントで `/q` コマンドを使用して、コードレビューの結果について質問したりやり取りしたりすることができます。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。

Note

Amazon Q Developer コンソールのコードレビュー機能設定は、プルリクエストが作成または再開されたときに実行される自動コードレビューを制御します。プルリクエストコメントで `/q review` スラッシュコマンドを使用してコードレビューを開始することは、この設定の影響を受けません。

`/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を含め、現在のステータス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 プルリクエスト内でのコードレビューの開始](#)」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以前に無効になっている場合は、GitHub 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。詳細については、「[Amazon Q Developer in GitHub の機能の編集](#)」を参照してください。

前提条件

Amazon Q Developer でコードレビューを開始する前に、ターゲット GitHub リポジトリに対する適切なアクセス許可が必要です。サポートされているリポジトリロールは、書き込み、メンテナンス、または管理者です。読み取りロールまたはトリアージロールを持つユーザー、およびロールを持たないメンバーは、Amazon Q Developer でコードレビューを開始できません。

GitHub トリアージロールを持つユーザーは、リポジトリ内のプルリクエストを引き続きレビューできます。ロールに関係なく、ユーザーはパブリックリポジトリでプルリクエストをレビューできます。

詳細については、「GitHub ドキュメント」の「[Organization のリポジトリロール](#)」および「[事前に定義済みの組織の役割について](#)」を参照してください。

GitHub プルリクエストのコードレビューの開始

新しいプルリクエストを開くか、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、コード品質、潜在的な問題、重要な検出結果に関するフィードバックを提供します。

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、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。
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「[pull request 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、潜在的な問題を検出します。Amazon Q Developer がレビューを完了すると、コードレビューの概要が表示されます。各検出結果は、プルリクエストにコミットできる推奨される修正とともに、概要の下にスレッドコメントとして表示されます。

4. 変更を実装し、プルリクエストのソースブランチに直接コミットを作成するようにエージェントに依頼します。これを行うには、`/q` で始まり、その後エージェントが変更を加えるためのリクエストを自然言語で投稿します。
5. (オプション) 特定の検出結果について質問します。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、質問 (「`/q explain the importance of this finding`」など) に続けて「`/q`」と入力します。
6. Amazon Q Developer によって提案されたコード変更を確認し、[提案をコミットする] を選択し、[変更をコミットする] を選択してプルリクエストを更新します。
7. 提案されたコード修正に問題がない場合は、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「[pull request のマージ](#)」を参照してください。

GitHub プルリクエスト内でのコードレビューの開始

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のイテレーションを実行できます。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。`/q review` スラッシュコマンドを使用したコードレビューの開始は、Amazon Q Developer コンソールのコードレビュー機能設定の影響を受けません。

プルリクエストでコードレビューの開始するには

1. 必要に応じて、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。

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、「[pull requests 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「/q review」と入力します。
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、コードレビューの概要が表示されます。各検出結果は、プルリクエストにコミットおよびアップデートできる提案される変更とともに、概要の下にスレッドコメントとして表示されます。

5. (オプション) 特定の検出結果について質問します。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、質問 (「/q explain the importance of this finding」など) に続けて「/q」と入力します。
6. 提案されたコード修正に問題がない場合は、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「[pull request のマージ](#)」を参照してください。

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については、1 か月あたりの呼び出し数が限られています。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、コードレビューと機能開発機能は、アプリのインストールGitHub時に 有効になります。登録時に、これらの機能のいずれか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 App をインストールするための要件](#)」および「[OAuth apps と Organization](#)」を参照してください。

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](#)に移動します。
2. ページの上部にある「Q Developer を有効にする」を選択し、プロンプトに従って Kiro と Amazon Q Developer を有効にします。

Kiro と Amazon Q Developer を以前に有効にした場合は、ステップ 3 に進みます。

3. ナビゲーションペインで、で Amazon Q Developer GitHubを選択します。
4. [インストールの登録] を選択し、[認証] を選択して GitHub に移動します。

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 7 に進みます。

5. 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。
6. [Amazon Q Developer を承認] を選択して、GitHub アカウントへのアクセスを付与します。認証後、Amazon Q Developer コンソールにリダイレクトされます。
7. [登録の詳細] で、GitHub の詳細を入力し、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] テキスト入力フィールドに、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) [組織名 - オプション] テキスト入力フィールドに、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。機能開発設定は変更できず、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。
8. Register を選択して、アプリのインストールを AWS アカウントに登録GitHubします。

アプリのインストールを正常に登録したら、登録の詳細を表示できます。コードレビュー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。登録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 GitHub に登録したら、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する。機能開発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検索する。
- 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する。

Amazon Q Developer in GitHub の機能の編集

Amazon Q Developer in GitHub で使用できる機能は、GitHub にアプリをインストールし、組織にアクセスするための認証を付与するときに、デフォルトで有効になります。GitHub 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。

Amazon Q Developer in GitHub の機能を有効化または無効化するには

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 [Amazon Q Developer in 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] を選択して変更を保存します。

[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 GitHub に反映されます。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、ラベルメニューで検索することなく、GitHub問題発生時に機能開発タスクを 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、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。このアクションは元に戻すことができません。

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 [Amazon Q Developer in 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:
 - [アクション] 列で、削除するインストールの [登録の削除] を選択します。
 - [名前] 列で、削除するインストールの登録名を選択し、[削除] を選択します。

[アクション] 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「**confirm**」と入力し、[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。

Amazon Q Developer for GitHub の問題のトラブルシューティング

Amazon Q Developer for GitHub 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。

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で Amazon Q Developer for GitHub 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、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。
2. GitHub 組織に移動し、許可するリポジトリに移動して、Amazon Q Developer for GitHub アプリを一覧表示します。
3. [設定] を選択し、ナビゲーションペインで [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、Amazon Q Developer for GitHub アプリを検索します。
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。

GitHub でのブランチ保護ルールの変更の詳細については、「[ブランチ保護ルールを作成する](#)」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問題に Amazon Q 開発エージェント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ない

問題: GitHub 問題を作成し、タスクを実行するために Amazon Q Developer を呼び出したが、技術的な問題に関する以下の一連のメッセージが表示された。

1. # I'm working on generating code for this issue. I'll update this issue with a comment and open a pull request when I'm done.
2. ## I'm having technical difficulties. I couldn't solve the issue. I'm going to try again. This might take some time.
3. ## I'm having technical difficulties. I couldn't solve the issue. I'm going to try again. This might take some time.
4. # I'm having technical difficulties. I couldn't solve the issue.

Consider reassigning this issue to a user. This will help you stay within the quotas for generative AI feature usage.

解決策: Amazon Q Developer が問題进行处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、「」を参照してください[GitHub での Amazon Q Developer による機能開発とイテレーション](#)。

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

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 GitLab または GitHub で構築できます。これらのルールは、チーム全体のコーディング標準とベストプラクティスを記述したものです。例えば、すべての Python コードが型ヒントを使用する必要があること、またはすべての Java コードで Javadoc コメントを使用する必要があることを示すルールを作成できます。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 はそれらをプロジェクト内で自動的にコンテキストとして使用し、ルールに従って機能開発用のコードを生成します。

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、次の手順を実行します。

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. プロジェクトのルートフォルダで、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. *project-root*/.amazonq/rules で、プロジェクトルールのファイルを作成します。これは、Markdown ファイルである必要があります。例えば、次のようになります。

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. 詳細なプロンプトをファイルに追加します。例えば、次のようになります。

```
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. 変更をコミットし、レビューして、マージします。
7. (オプション) プロジェクトルールの Markdown ファイルをさらに追加します。

これで、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でコンテキストとして自動的に使用します。

Amazon Q Developer in chat applications でのチャット

Amazon Q Developer とのチャットは、Microsoft Teams および Slack チャットアプリケーションで行うことができます。Amazon Q は、設定されたチャネルで、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ついての質問に答えることができます。設定されたチャットアプリケーションでは、Amazon Q の以下のチャット機能を利用できます。

- [AWS に関するチャット](#)
- [リソースに関する Amazon Q Developer とのチャット](#)
- [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](#)
- [テレメトリと運用に関するチャット](#)
- [Reachability Analyzer での Amazon Q ネットワークのトラブルシューティング](#)

Amazon Q をチャットアプリケーションで使うときに利用できる機能の完全な一覧については、「Amazon Q Developer in chat applications 管理者ガイド」の「[What is Amazon Q Developer in chat applications?](#)」を参照してください。

Note

Amazon Q Developer in chat applications 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。

お使いのチャネルで Amazon Q チャットを有効にする

Amazon Q Developer が既に設定されている Microsoft Teams チャネルまたは Slack チャネルにチャット機能を追加する手順は次のとおりです。チャットアプリケーションで初めて Amazon Q Developer を設定してユーザーが Amazon Q とチャットできるようにするには、「Amazon Q Developer in chat applications 管理者ガイド」の「[Get started with Microsoft Teams](#)」と「[Get started with 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](#) マネージドポリシーを含め、次にポリシーをチャン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. AWS マネジメントコンソール にサインインして、[\[IAM コンソール\]](#) 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ンネルガードレールに追加します。
 - a. [Amazon Q Developer in chat applications コンソール](#)を開きます。
 - b. 設定済みのクライアントを選択します。
 - c. 設定済みのチャンネルを選択します。
 - 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。

チャンネルで 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、設定済みのチャンネルで Amazon Q にたず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?

Amazon Q Developer のセキュリティ

で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 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。
- クラウド内のセキュリティ – 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、() を使用して認証情報を保護し AWS アカウント、個々のユーザーを設定することをお勧めします AWS Identity and Access Management 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アクセス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、内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 検証済みの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。

お客様の E メールアドレスなどの極秘または機密情報を、[タグ](#)、または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含めないことを強くお勧めします。これは、API AWS マネジメントコンソール、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 でコンテンツがどのように使用されるかについて詳しくは、「[Amazon Q Developer のサービスの改善](#)」を参照してください。

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 およびコマンドラインでのデータ共有のオプトアウト](#)
- [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、質問とその回答、および追加のコンテキスト (コンソールのメタデータやコードなど) を保存して、質問やリクエストに対する回答を生成します。データの暗号化方法については、「[Amazon Q Developer でのデータ暗号化](#)」を参照してください。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 マネジメントコンソール」
- Amazon Q による AWS コンソールエラーの診断
- Eclipse、JetBrains IDE、Visual Studio Code、および Visual Studio での Amazon Q
- コマンドラインでの Amazon Q

Amazon Q Developer Pro ティア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。

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。それ以外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。以下の機能では保存方法が異なる点に注意してください。

Note

Amazon Q アーティファクトを使用する場合、ビジュアライゼーション関連のコンテンツは米国リージョンに保存されます。

[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 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「[CloudWatch 調査のセキュリティ](#)」のトピックを参照してください。

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、「[Cross-region inference](#)」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 アイデンティティセンターの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。KMS キーの作成の詳細については、「AWS Key Management Service Developer Guide」の「[Creating keys](#)」を参照してください。

カスタ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。この権限は、エンドユーザーが保管中のデータ暗号化に暗号化キーを使用できるようにするものです。権限の詳細については、「[Grants in 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会話を維持する場合は、そ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。

カスターマネージド KMS キーの使用

カスターマネージド KMS キーを作成した後、Amazon Q Developer の管理者は Amazon Q Developer コンソールでキーを指定し、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、「[Amazon Q Developer で暗号化方法の管理](#)」を参照してください。

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「[管理者に Amazon Q Developer コンソールの使用を許可する](#)」に含まれています。

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、「[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](#)」を参照してください。

Amazon Q Developer の使用中に KMS アクセス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アクセス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ツを使用するのか、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ユーザーのコンテンツを使用しません。AWS が今後これを有効にする場合は、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。

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 マネジメントコンソール 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 およびコマンドラインでのデータ共有のオプトアウト](#)」を参照してください。

IDE およびコマンドラインでのデータ共有のオプトアウト

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。

IDE で収集されるテレメトリ

IDE 内の Amazon Q で収集される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](#)」を参照してください。

Amazon Q を使用する IDE によって収集されるテレメトリデータの詳細については、以下の GitHub リポジトリのドキュメントを参照してください。

- [Amazon Q extension for VS Code](#)
- [Amazon Q plugin for JetBrains](#)
- [Amazon Q plugin for Eclipse](#)
- [AWS Amazon Q を使用した Visual Studio Toolkit](#)

Q CLI で収集されるテレメトリ

Q CLI で収集される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「[telemetry_definitions.json](#)」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るテレメトリ

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

自由テキストフィールドには、個人を特定できる情報 (PII) や、機密情報あるいは秘匿性の高い情報は追加しないでください。

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手順を確認するには、使用する IDE を選択してください。

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。

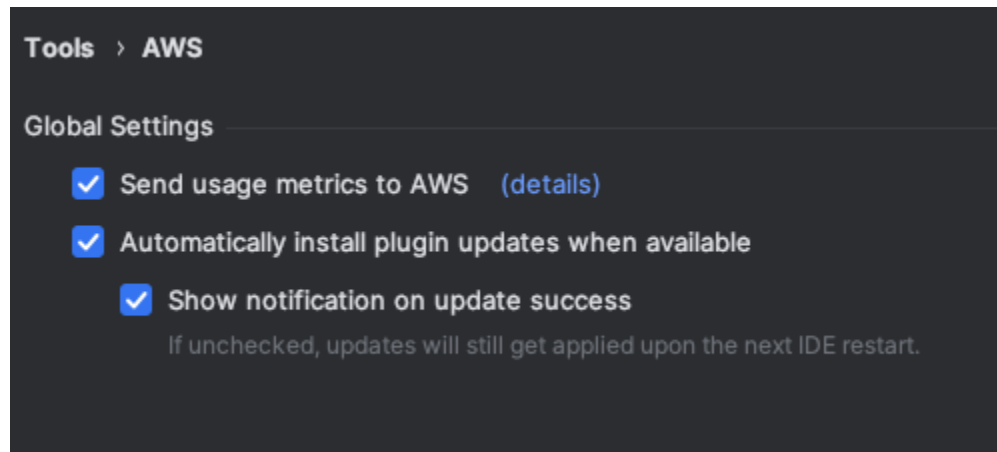
Note

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 IDE 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. Eclipse IDE で、[設定] を開きます。
2. 左のナビゲーションペインで、[Amazon Q] を選択します。
3. [AWSに使用状況メトリクスを送信] の横にあるチェックボックスをオフにします。
4. [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 を使用する場合は、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 ウィンドウの上部で、[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. `qct configure` を実行して要求された設定の詳細を入力するか、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、`N`を入力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. テレメトリ収集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たら「Y」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 ティアでは、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。

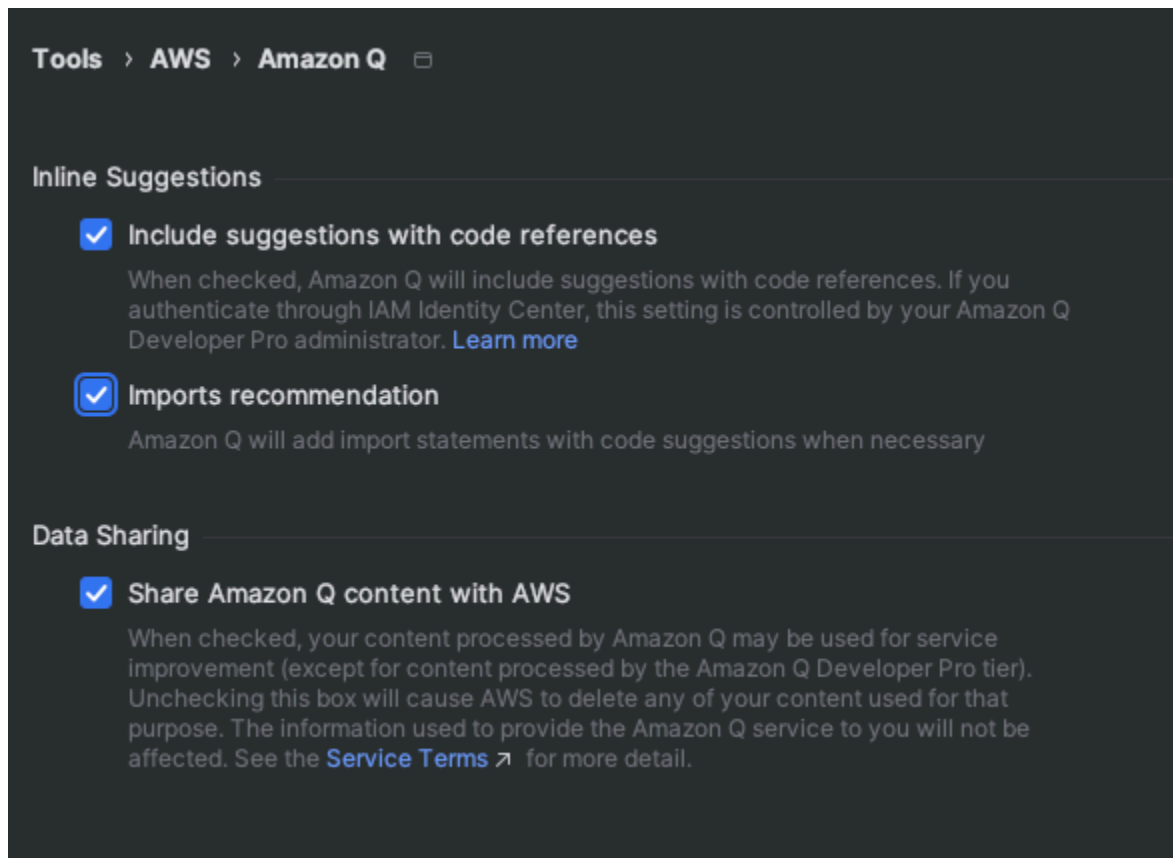
1. VS Code で [設定] を開きます。
2. VS Code ワークスペースを使用している場合は、[ワークスペース] サブタブに切り替えます。VS Code では、ワークスペース設定がユーザー設定より優先されます。
3. [検索] バーに「Amazon Q: Share Content」と入力します。
4. チェックボックスをオフにします。

JetBrains

Amazon Q Developer Pro ティアでは、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 ティアでは、Amazon Q はコンテンツを収集しません。

Amazon Q Developer 無料利用枠で、Eclipse IDE 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. Eclipse IDE の最新バージョンを利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 AWS と共有] の横にあるチェックボックスをオフにします。
5. [適用] を選択して変更を保存します。

Visual Studio

Amazon Q Developer Pro ティアでは、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 とコンテンツを共有しません 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 AWSとコンテンツを共有しません。

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リージョンにトラフィックを分散し、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、以下が可能になります。

- 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 LLM に基づいた、新しい 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 の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は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暗号化されて送信されます。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 を使用する際のデータの保存先については、「[Amazon Q Developer でのデータ保護](#)」を参照してください。

Amazon Q Developer の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、リクエストが発生した地域に応じて、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-south-east-1)
	アジアパシフィック (シドニー) (ap-south-east-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一覧については、「[Amazon Q Developer が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 間で行う API コール AWS リージョン 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

Amazon Q Developer によって行われる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 コール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 コール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 コールを実行できないようにする IAM ポリシーについては、「[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](#)」を参照してください。

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

- [オーディエンス](#)
- [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](#)

オーディエンス

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

認証とは、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 アイデンティティセンター (IAM アイデンティティセンター) 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 AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、AWS マネジメントコンソール 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、では、アカウントのセキュリティを高めるために多要素認証 (MFA) を使用する AWS ことをお勧めします。詳細については、「AWS IAM アイデンティティセンター 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 アイデンティティセンター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 アイデンティティセンター 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簡単になります。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 っていますが、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 マネジメントコンソール 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 アイデンティティセンター 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 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 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。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 マネジメントコンソール、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、AWS WAF、Amazon VPC は 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 をアカウントの一部またはすべてに適用できます。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アクセス許可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 機能	Amazon Q のサポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	いいえ
ポリシー条件キー	いいえ

IAM 機能	Amazon Q のサポート
ACL	なし
ABAC (ポリシー内のタグ)	いいえ
一時的な認証情報	あり
プリンシパルアクセス権限	あり
サービスロール	いいえ
サービスリンクロール	はい

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については、「IAM ユーザーガイド」の[AWS のサービス「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、ユーザーグループ、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。アイデンティティベース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。JSON ポリシーで使用す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コントロールできます。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、[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、全体のアカウント、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。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。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。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。リソースレベルのアクセス許可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 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。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。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アクセス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、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可するように ABAC ポリシーを設計できます。

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

一時的な認証情報は、AWS リソースへの短期的なアクセスを提供し、フェデレーションまたはスウィッチロールの使用時に自動的に作成されます。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 ユーザーガイド」の「[IAM の一時的な認証情報](#)」および「[AWS のサービスと IAM との連携](#)」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

転送アクセスセッション (FAS) は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウ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 を使用します。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 マネジメントコンソール、、、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、組織で利用できる 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、削除できるかどうかを決定します。これらのアクションでは、AWS アカウントに費用が発生する場合があります。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。これらはで使えます 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 ポリシーでアクセス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。たとえば、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 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 アクセスアナライザーを使用して IAM ポリシーを検証し、安全で機能的な権限を確保する – IAM アクセスアナライザーは、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 – 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

アクセスを提供するには、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。

- 以下のユーザーとグループ AWS IAM アイデンティティセンター:

アクセス許可セットを作成します。「AWS IAM アイデンティティセンター ユーザーガイド」の「[アクセス許可セットを作成する](#)」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

次のポリシーは、Amazon Q Developer 管理者が 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するためのものです。

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、ユーザーに Amazon Q コンソールでのアクションの実行を許可します。Amazon Q コンソールは、Amazon Q と AWS IAM アイデンティティセンター 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。

Amazon Q Developer Pro タスクの一部には、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べきものがあります。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](#)」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」セクション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、「[管理者にプラグインの設定を許可する](#)」を参照してください。

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。

```
{
  "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": [
      "identitystore:DescribeUser"
    ],
    "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。

```
{  
  "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 管理者に Amazon Q Developer コンソールへのアクセスが必要となります。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 `codeconnections:ListRepositories` のアクセス許可に関するエラーを報告します。エラーを無視して、これらのアクセス許可を含むポリシーを作成します。アクセス許可は必須で、ポリシーはエラーがあっても機能します。

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、#####を自分の 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:ListOwners",
            "codeconnections:ListRepositories",
            "codeconnections:ListConnections",
            "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も必要です。

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"
        ]
      }
    }
  }
]
}

```

管理者があるプロバイダーからの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 で指定されたある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、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 Wiz など、設定を許可するプラグインプロバイダーの名前。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も必要です。

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCreateProviderPlugin",
      "Effect": "Allow",
      "Action": [
        "q:CreatePlugin",

```

```

        "q:GetPlugin",
        "q>DeletePlugin"
    ],
    "Resource": "arn:aws:qdeveloper:us-east-1:111122223333: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:us-east-1:111122223333:vpc/*"
    ],
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/CreatedBy": "AWSApplicationMigrationService"
      }
    }
  },
  {
    "Sid": "MGNNetworkMigrationAnalyzerEC2RequestSgTag",
    "Effect": "Allow",
    "Action": [
      "ec2:CreateSecurityGroup"
    ],
    "Resource": [
      "arn:aws:ec2:us-east-1:111122223333:security-group/*",
      "arn:aws:ec2:us-east-1:111122223333:security-group-rule/*"
    ],
    "Condition": {
      "StringEquals": {

```

```

        "aws:RequestTag/CreatedBy": "AWSApplicationMigrationService"
    }
}
},
{
    "Sid": "MGNNetworkMigrationAnalyzerEC2SecurityGroupTags",
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags"
    ],
    "Resource": [
        "arn:aws:ec2:us-east-1:111122223333:security-group/*",
        "arn:aws:ec2:us-east-1:111122223333:security-group-rule/*",
        "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
        "arn:aws:ec2:us-east-1:111122223333:network-insights-path/*",
        "arn:aws:ec2:us-east-1:111122223333: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:us-east-1:111122223333:subnet/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/CreatedBy": "AWSApplicationMigrationService"
        }
    }
}
}

```

```

    },
    {
      "Sid": "MGNNetworkMigrationAnalyzerENISG",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:us-east-1:111122223333: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、、、AWS Documentation サイトなどの AWS アプリケーションやウェブサイトで Amazon Q Developer AWS マネジメントコンソールの機能にアクセスすることを許可 AWS Console Mobile Application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 で Amazon Q](#) にアクセスするユーザーや、[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ーマネージドキーを設定している場合に 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": "*"
    }
  ]
}

```

で Amazon Q CLI の使用をユーザーに許可する AWS CloudShell

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の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](#)。このポリシーは、[コマンドラインでの Amazon Q](#) へのアクセスには影響しません。

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、Amazon Q とのリソースに関するチャットを許可し、Amazon Q がユーザーに代わってリソース情報を取得することを許可します。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": "*"
  },
  {
    "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推奨事項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、AWS Compute Optimizer、AWS Budgets、AWS 無料利用枠、AWS 料金表、Savings Plans のアクセス許可と予約の推奨事項が含まれています。

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": "AllowCostExplorerAccess",
      "Effect": "Allow",
      "Action": [
        "ce:GetCostAndUsage",
        "ce:GetCostAndUsageWithResources",
        "ce:GetCostForecast",
        "ce:GetUsageForecast",
        "ce:GetTags",
        "ce:GetCostCategories",
        "ce:GetDimensionValues",
        "ce:GetSavingsPlansUtilization",
        "ce:GetSavingsPlansCoverage",
        "ce:GetSavingsPlansUtilizationDetails",
        "ce:GetReservationUtilization",
        "ce:GetReservationCoverage",
        "ce:GetSavingsPlansPurchaseRecommendation",
        "ce:GetReservationPurchaseRecommendation",
        "ce:GetRightsizingRecommendation",
        "ce:GetAnomalies",
        "ce:GetCostAndUsageComparisons",
        "ce:GetCostComparisonDrivers"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowCostOptimizationHubAccess",
      "Effect": "Allow",
      "Action": [
        "cost-optimization-hub:GetRecommendation",
        "cost-optimization-hub:ListRecommendations",
        "cost-optimization-hub:ListRecommendationSummaries"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowComputeOptimizerAccess",
      "Effect": "Allow",
      "Action": [
        "compute-optimizer:GetAutoScalingGroupRecommendations",
        "compute-optimizer:GetEBSVolumeRecommendations",
        "compute-optimizer:GetEC2InstanceRecommendations",
        "compute-optimizer:GetECSServiceRecommendations",

```

```

        "compute-optimizer:GetRDSDatabaseRecommendations",
        "compute-optimizer:GetLambdaFunctionRecommendations",
        "compute-optimizer:GetIdleRecommendations",
        "compute-optimizer:GetLicenseRecommendations",
        "compute-optimizer:GetEffectiveRecommendationPreferences"
    ],
    "Resource": "*"
},
{
    "Sid": "AllowBudgetsAccess",
    "Effect": "Allow",
    "Action": [
        "budgets:ViewBudget"
    ],
    "Resource": "*"
},
{
    "Sid": "AllowFreeTierAccess",
    "Effect": "Allow",
    "Action": [
        "freetier:GetFreeTierUsage",
        "freetier:GetAccountPlanState",
        "freetier:ListAccountActivities",
        "freetier:GetAccountActivity"
    ],
    "Resource": "*"
},
{
    "Sid": "AllowPricingAccess",
    "Effect": "Allow",
    "Action": [
        "pricing:GetProducts",
        "pricing:GetAttributeValues",
        "pricing:DescribeServices"
    ],
    "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": "*"
    }
  ]
}

```

あるプロバイダーのプラグインとユーザーが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アカウントのアカウント AWS ID。

- **plugin-provider** – CloudZero、Datadog、または Wiz 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。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-1:111122223333: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した場合、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ません。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、または Wiz 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。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。

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:ListGroupsWithUser",
        "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際に、[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 を設定する際は、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、KMS キーの使用を、指定された AWS サービスからのリクエストに制限します。また条件キーを使用して、特定のサービスからリクエストが送信された場合に KMS キーの使用許可を拒否することもできます。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 Developer Guide」の「[kms:ViaService](#)」を参照してください。

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、承認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て、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 Developer Guide」の「[Encryption context](#)」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"
      ]
    }
  ]
}
```



```
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:ViaService": "q.us-east-1.amazonaws.com",
            "kms:EncryptionContext:aws-crypto-
ec:aws:qdeveloper:accountId": "111122223333"
        }
    }
},
{
    "Sid": "Sid1",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/rolename"
    },
    "Action": "kms:DescribeKey",
    "Resource": "*"
}
]
```

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 アイデンティティセンター 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 アイデンティティセンター で表される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え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表示するには、「AWS Managed Policy Reference」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表示するには、「AWS Managed Policy Reference」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 を使用するの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。

ユーザーの IAM エンティティに AWSServiceRoleForAmazonQDeveloper を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 Managed Policy Reference」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 を使用するの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。

ユーザーの IAM エンティティに AWSServiceRoleForUserSubscriptions を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。
- `kms` – プリンシパルが KMS キーにアクセスして Identity Center で承認できるようにします。

このポリシーのアクセス許可を確認するには、「AWS Managed Policy Reference」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、GitLab Duo with Amazon Q 統合の機能を利用するアクセス許可を付与します。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、以下を許可します。

- GitLab Duo の使用許可 - 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 の管理アクセス許可 - OAuth アプリ接続の作成と削除を有効にし、統合設定をコントロールします。
- GitLab Duo の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 Managed Policy Reference」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	Identity Center で KMS キーへのアクセスを許可するためのアクセス許可が追加されました。	2025 年 10 月 29 日
AmazonQFullAccess – ポリシーを更新	Identity Center で KMS キーへのアクセスを許可するためのアクセス許可が追加されました。	2025 年 10 月 29 日
AWSServiceRoleForUserSubscriptions – ポリシーを更新	Identity Center で KMS キーへのアクセスを許可するためのアクセス許可が追加されました。	2025 年 10 月 29 日
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
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
GitLabDuoWithAmazonQPermissions	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アクセス許可が追加されました。	2025 年 4 月 30 日

変更	説明	日付
ionsPolicy – ポリシーを更新		
GitLabDuoWithAmazonQPermissionsPolicy – 新しいポリシー	GitLab が Amazon Q Developer に接続して GitLab Duo with Amazon Q 統合機能を使用できるようになりました。	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なりました。	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
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

変更	説明	日付
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 アイデンティティセンター、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」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。

[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 というロールのアクセス許可ポリシーでは、Amazon Q Developer は、指定されたリソースで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 マネジメントコンソール、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ある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	いいえ
米国西部 (北カリフォルニア)	us-west-1	いいえ
米国西部 (オレゴン)	us-west-2	いいえ
アフリカ (ケープタウン)	af-south-1	いいえ

リージョン名	リージョン識別子	Amazon Q Developer でのサポート
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	いいえ
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	いいえ
アジアパシフィック (シンガポール)	ap-southeast-1	いいえ
アジアパシフィック (シドニー)	ap-southeast-2	いいえ
アジアパシフィック (東京)	ap-northeast-1	いいえ
カナダ (中部)	ca-central-1	いいえ
欧州 (フランクフルト)	eu-central-1	いいえ
欧州 (アイルランド)	eu-west-1	いいえ
欧州 (ロンドン)	eu-west-2	いいえ
欧州 (ミラノ)	eu-south-1	いいえ
欧州 (パリ)	eu-west-3	いいえ
欧州 (ストックホルム)	eu-north-1	いいえ
中東 (バーレーン)	me-south-1	いいえ
中東 (UAE)	me-central-1	いいえ
南米 (サンパウロ)	sa-east-1	いいえ
AWS GovCloud (米国東部)	us-gov-east-1	いいえ
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、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。

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](#) というロールのアクセス許可ポリシーでは、ユーザーサブスクリプションは、指定されたリソースで次のアクションを完了することができます。

- 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 マネジメントコンソール、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ある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	はい
米国西部 (オレゴン)	us-west-2	はい

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	はい
米国東部 (オハイオ)	us-east-2	はい
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	はい
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	はい

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対象範囲内」](#)の「コンプライアンス」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「コンプライアンスプログラム」](#)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。を使用する際のコンプライアンス責任の詳細については AWS のサービス、[AWS「セキュリティドキュメント」](#)を参照してください。

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ー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、フォールトトレランス、および拡張性が優れています。

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 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環境を AWS 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。

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 が想定どおりに動作するように、次の URL と Amazon リソースネーム (ARN) へのトラフィックを許可リストに登録します。

許可リストの一般的な URL

次の URL で、以下の置き換えを行います。

- *idc-directory-id-or-alias* を IAM アイデンティティセンターインスタンスのディレクトリ ID またはエイリアスで置き換える。IAM アイデンティティセンターの詳細については、「AWS IAM アイデンティティセンター 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 アイデンティティセンターのリージョン](#)」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 の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「[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。

[URL]	目的
<i>idc-directory-id-or-alias</i> .awsapps.com	認証
oidc. <i>sso-region</i> .amazonaws.com	認証
*.sso. <i>sso-region</i> .amazonaws.com	認証
*.sso-portal. <i>sso-region</i> .amazonaws.com	認証
*.aws.dev	認証

[URL]	目的
*.awsstatic.com	認証
*.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. <i>profile-region</i> .amazonaws.com	Amazon Q Developer 機能
https://idetoolkits-hostedfiles.amazonaws.com/*	IDE の Amazon Q Developer、構成
https://idetoolkits.amazonaws.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

Amazon S3 バケット URL および ARN の許可リストへの追加

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。これらのバケットを許可リストに登録するには、Amazon S3 へのアクセス制御の方法に応じて、バケット URL またはバケット ARN を使用します。

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 の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。

Amazon S3 バケットの URL および 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	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
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	
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	Amazon Q コード変換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 バケットの URL および ARN	目的
欧州 (フランクフルト): - https://amazonq-code-transformation-eu-central-1-a0a89cc2b94.s3.amazonaws.com/ - arn:aws:s3:::amazonq-code-transformation-eu-central-1-a0a89cc2b94	

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内部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

SOCKS プロキシとプロキシ自動構成 (PAC) ファイルはサポートされていません。次の手順を使用して、HTTP または HTTPS プロキシを手動で設定する必要があります。

Eclipse

- Eclipse で、次のように [設定] を開きます。
 - Windows または Ubuntu の場合:
 - Eclipse メニューバーで、[ウィンドウ] を選択して、[設定] を選択します。
 - macOS の場合:
 - メニューバーで Eclipse を選択し、macOS のバージョンに応じて [設定] または [プリファレンス] を選択します。
- 検索バーで、「**Amazon Q**」と入力して Amazon Q を開きます。
- [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、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 のメインメニューで、[ツール]、[オプション] の順に選択します。
2. [オプション] メニューで、[AWS Toolkit] を展開して、[プロキシ] を選択します。
3. [Proxy] メニューで、[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 で、**CMD + ,** (Mac) または **Ctrl + ,** (Windows/Linux) を押して、VS Code の [設定] を開きます。
2. [設定] 検索バーで「**Http: Proxy**」と入力して検索します。
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「**HTTP: No Proxy**」と入力して検索します。
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る。
- 企業プロキシがシステムの信頼ストアにない証明書を使用している。
- Amazon Q が企業証明書を自動的に検出できない。

Eclipse

- Eclipse の Amazon Q 設定の [プロキシ設定] 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、「[Getting your corporate certificate](#)」を参照してください。

JetBrains

JetBrains で、IntelliJ IDEA ドキュメントの「[Trusted root certificates](#)」の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、「[Getting your corporate certificate](#)」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、「[企業証明書の取得](#)」を参照してください。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 で、**CMD + ,** (Mac) または **Ctrl + ,** (Windows/Linux) を押して、VS Code の [設定] を開きます。
2. [設定] 検索バーで「**Amazon Q > Proxy: Certificate Authority**」と入力して検索します。
3. 企業証明書ファイルのパスを入力します。ファイル拡張子は .pem または .crt になります。

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、「[企業証明書の取得](#)」を参照してください。

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 `https://` が含まれていることを確認します。
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。
 - Eclipse で Amazon Q にサインインし、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] の順に選択します。
 - Eclipse メニューから、[ウィンドウ]、[ビューを表示]、[エラーログ] の順に選択します。

Note

次のエラーメッセージが表示された場合は、以下のように対応します。

- `unable to verify the first certificate` の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- `self signed certificate` の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- `ECONNREFUSED` の場合、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 `https://` が含まれていることを確認します。
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「[Locating IDE log files](#)」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は、以下のように対応します。

- `unable to verify the first certificate` の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- `self signed certificate` の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- `ECONNREFUSED` の場合、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 `https://` が含まれていることを確認します。
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. Toolkit AWS 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。
 - Toolkit logs AWS 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、「[Troubleshooting Extensions with the Activity Log](#)」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は、以下のように対応します。

- unable to verify the first certificate の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- self signed certificate の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- ECONNREFUSED の場合、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 https:// が含まれていることを確認します。
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は、以下のように対応します。

- unable to verify the first certificate の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- self signed certificate の場合、「[ステップ 2: SSL 証明書処理を設定する](#)」の手順に従って証明書を手動で指定します。
- ECONNREFUSED の場合、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 チームに以下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、認証方法などのプロキシサーバーの詳細。

または、以下の手順で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](#) で利用可能な機能に対して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。

以下の Amazon Q 用の VPC エンドポイントを作成するには、次のサービス名を使用します。

- `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 Region 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、「[を使用した Amazon Q Developer API コールのログ記録 AWS CloudTrail](#)」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 サブスクリバ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、「[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。
- 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 でのユーザープロンプトのログ記録](#)」を参照してください。

を使用した Amazon Q Developer API コールのログ記録 AWS CloudTrail

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 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 によりログ記録され、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
- リクエストの送信に使用された一時的なセキュリティ認証情報に、ロールとフェデレーテッドユーザーのどちらが使用されたか
- 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 element](#)」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、任意の出典からの単一のリクエストを表し、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に関する情報が含まれます。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": "AIDA6ON6E4XEGIEEXAMPLE",
    "arn": "arn:aws:iam::555555555555:user/Mary",
    "accountId": "555555555555",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDA6ON6E4XEGIEEXAMPLE",
        "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": "AIDA60N6E4XEGIEEXAMPLE",
    "arn": "arn:aws:iam::555555555555:user/Paulo",
    "accountId": "555555555555",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDA60N6E4XEGIEEXAMPLE",
        "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": "2024-04-10T14:06:08Z",
  "eventSource": "aws.s3",
  "eventName": "ListBuckets",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.168.1.1",
  "userAgent": "awscli/2.15.12",
  "requestParameters": {
    "bucket": "my-bucket"
  },
  "responseElements": {
    "buckets": [
      {
        "name": "my-bucket",
        "creationDate": "2024-04-10T14:06:08Z",
        "owner": {
          "id": "AIDA60N6E4XEGIEEXAMPLE",
          "type": "Role",
          "arn": "arn:aws:iam::555555555555:user/Paulo",
          "accountId": "555555555555",
          "userName": "Paulo"
        }
      }
    ]
  },
  "requestID": "12345678-1234-1234-1234-123456789012",
  "eventID": "12345678-1234-1234-1234-123456789012",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": false,
  "recipientAccountId": "555555555555",
  "eventCategory": "Data"
}
```

```
    },
    "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の確に把握できます。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	呼び出し	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把握できます。
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特定するには、SendTelemetryEvent と呼ばれるイベントを探し、そのイベントに含まれる 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 で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: 完了がブロックが行か。
- 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的識別子。
- 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、関連するコードスニ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ニペットのコピー、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 (コードスニ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`: 受け入れられたスニ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を [Amazon CloudWatch Logs](#) に保存できます。

Amazon Q Developer の管理者に、次のアクセス許可セットを使用してこれらのログを表示する権限を付与できます。

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ロギング \[V2\]](#)」を参照してください。

JSON

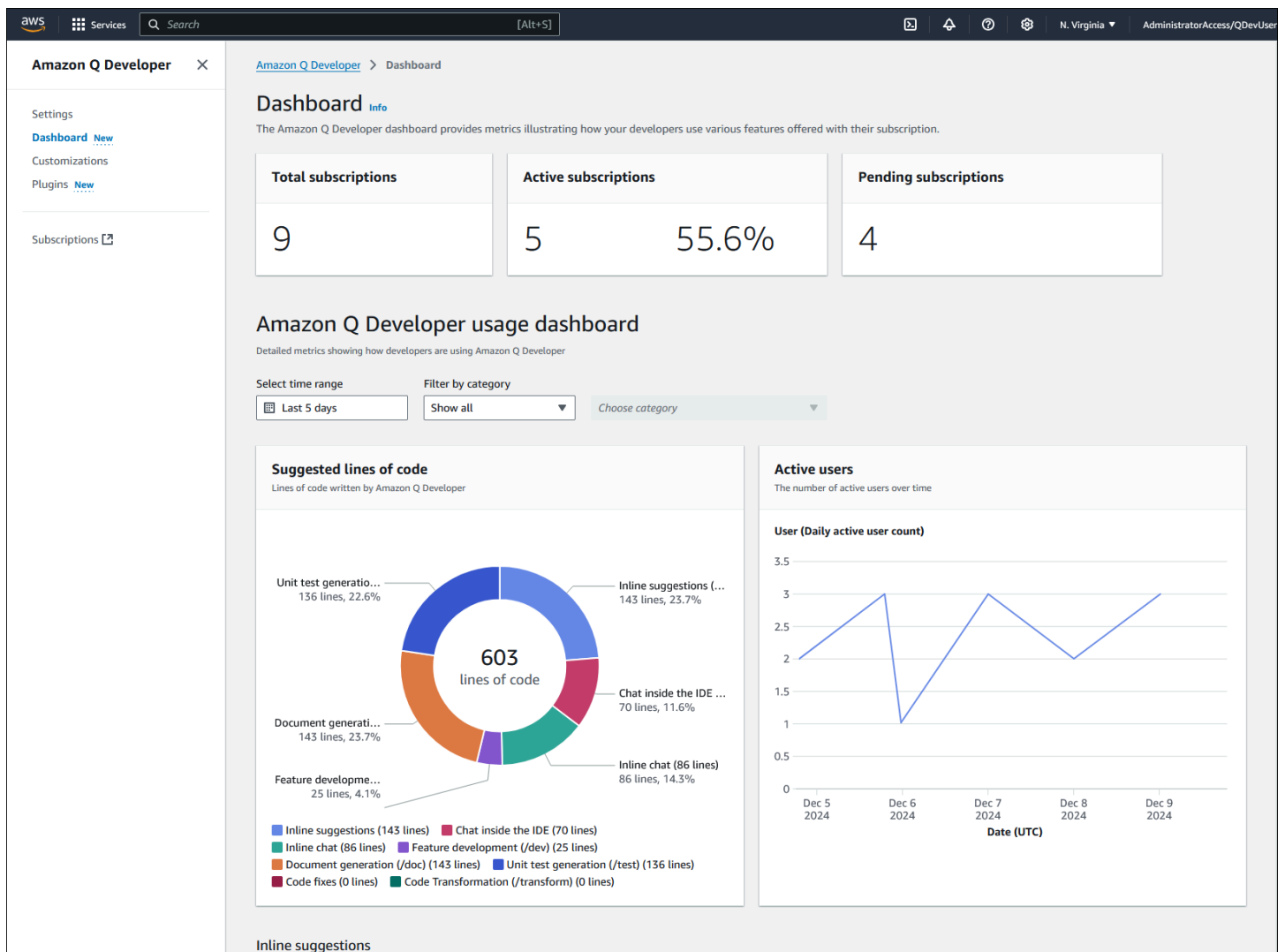
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowLogDeliveryActions",
      "Effect": "Allow",
      "Action": [
        "logs:PutDeliverySource",
        "logs:GetDeliverySource",
        "logs>DeleteDeliverySource",
        "logs:DescribeDeliverySources",
        "logs:PutDeliveryDestination",
        "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 ダッシュボードは、貴社の Pro ティアのサブスクライバーがサービスをどのように使用しているかについての有用なデータをまとめたものです。



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

および

- 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 マネジメントコンソール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) ペ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、「[ダッシュボードのトラブルシューティング](#)」を参照してください。
- Amazon Q Developer の使用状況のユーザーごとの内訳を含むユーザーメトリクスを日次レポートに送信する場合は、「[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。
- 特定のメトリクスの詳細については、「[Amazon Q Developer ダッシュボードの使用状況メトリクスの説明](#)」を参照するか、ダッシュボードページの上にあるヘルプリンク



を選択します。

Amazon Q Developer ダッシュボードの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。このサブスクリプションに対して請求が行われます。アクティブサブスクリプションの詳細については、「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。
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。このサブスクリプションに対しては請求は行われません。保留中のサブスクリプションの詳細については、「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。
受け入れたコード行	Amazon Q 機能 (機能開発、ドキュメント生成、ユニットテスト生成など) 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。
アクティブユーザー	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た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での提案と承認された提案の合計数を表示します。承認された提案の割合は、ユーザーが承認した提案の数を 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、承認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し続けたり、実行を開始したりしたために拒否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での提案の合計数と承認された提案の数を表示します。承認された提案の割合は、ユーザーが承認した提案の数を 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 ウィンドウでの Amazon Q からの応答の合計数を表示します。
機能開発 – 承認率	機能開発 機能の承諾率を表示します。承認率は、ユーザーが承認したコード行数を Amazon Q が提案したコード行の合計で割って計算されます。
ドキュメントの生成	ドキュメント 生成 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) の総数を表示します。承認率は、ユーザーが承認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。
ユニットテストの生成	ユニットテスト 生成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	コード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合計数を表示します。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。
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。

メトリクス名	説明
コード変換	変換機能によって実行されたコード 変換 の合計数と、処理されたコードの行数を表示します。

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 アイデンティティセンター、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 アイデンティティセンターのアカウントインスタンスで Amazon Q Developer をセットアップした場合は、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 日以前に生成されたメトリクスを表示するには、`cloudwatch:GetMetricData` および `cloudwatch:ListMetrics` も必要になります。
- 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] のトグル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なります。

- 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、「[ユーザーアクティビティレポートのメトリクス](#)」で説明されているメトリクスが表示されます。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、それぞれ 1,000 人のユーザーを含む複数の CSV ファイルにデータを分割し、`part_1`、`part_2` などのサフィックスを付与します。

Note

ユーザーアクティビティレポートを有効にすると、Amazon Q が使用状況データを AWS に送信できるかについて管理者がどのように設定したかにかかわらず、Amazon Q はテレメト

リを収集します。この設定は、ユーザーの組織ではなく AWS corporation 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 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 アイデンティティセンターの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す。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、「[Amazon Q Developer プロファイルとは](#)」および「[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。
- 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) [プロンプトログ](#)に使用するバケットとは異なる場所にある。
- 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が含まれている。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": "111122223333"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:codewhisperer:us-east-1:111122223333:*"
        }
      }
    }
  ]
}

```

(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、「[管理者に Amazon Q Developer コンソールの使用を許可する](#)」で定義されたアクセス許可が必要です。

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. [ユーザーごとに詳細なメトリクスを収集する] を切り替えます。
5. [S3 ロケーション] で、CSV レポートを保持する 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 によって生成される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、「[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。

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たコード行。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) を通じて生成され、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。

メトリクス名	説明
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 の応答が含まれます。
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の修正の数。このメトリクスは、コード レビュー機能を通じて生成されたコード 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たコード行。このメトリクスは、コード レビュー機能 を通じて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、コード レビュー機能を通じて生成されたコード 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、コード レビュー機能を通じて生成されたコード 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、コード レビュー機能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、コード レビュー機能を使用して検出されたコード の問題に適用されます。
CodeReview_SucceededEventCount	検出され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、コード レビュー機能を使用して見つかったコード の問題に適用されます。

メトリクス名	説明
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たコード行。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 によって提案されたコード行。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。
DocGeneration_AcceptedFileUpdates	Amazon Q によって提案され、ユーザーが受け入れた行の更新の数。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、ユーザーが受け入れたドキュメント追加の行。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_AcceptedLineUpdates	Amazon Q によって提案され、ユーザーが受け入れたドキュメント更新の行。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。

メトリクス名	説明
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、ユーザーが拒否し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、ユーザーが拒否し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、ユーザーが拒否したドキュメント追加の行。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、ユーザーが拒否したドキュメント更新の行。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) で生成されたコードの追加が含まれます。
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) で提案されたコードの削除が含まれます。

メトリクス名	説明
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	ユーザーによって破棄された インラインチャット (Amazon Q チャット ではない) 提案の数。 「破棄」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、ユーザーが破棄したコード追加の行。「破棄」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) で生成されたコードの追加が含まれます。
InlineChat_DismissedLineDeletions	Amazon Q によって提案され、ユーザーが破棄したコード削除の行。「破棄」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) で提案されたコードの削除が含まれます。
InlineChat_EventCount	ユーザーがエンゲージした 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。

メトリクス名	説明
InlineChat_RejectedLineAdditions	Amazon Q によって提案され、ユーザーが拒否したコード追加の行。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) で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、ユーザーが拒否したコード削除の行。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) で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たコード行。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	ユーザーによって受け入れられた インライン提案 の数。
Inline_SuggestionsCount	ユーザーに表示された 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たコード行。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 で Amazon Q にエンゲージした回数。

メトリクス名	説明
TestGeneration_GeneratedLines	Amazon Q によって提案されたコード行。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 にエンゲージした数。ユーザーが コマンドラインでコードを変換した数 を除く。
Transformation_LinesGenerated	Amazon Q によって提案されたコード行。このメトリクスは、 コマンドラインで変換されたコード を除き、 /transform コマンド で生成されたコードに適用されます。
Transformation_LinesIngested	変換用に 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 を介して提供されるコードに適用されます。 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 でのユーザープロンプトのログ記録

管理者は、ユーザーが統合開発環境 (IDE) で 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。

開発者がインライン提案を使用する際、Amazon Q は承認された提案と明示的に拒否された提案を記録します。開発者が Amazon Q とチャットする際、Amazon Q は開発者のプロンプトと Amazon Q の回答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、次のパスにユーザーが作成した Amazon S3 バケット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、「[Amazon Q Developer のプロンプトログの例](#)」を参照してください。

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

- ユーザーがスタンドアロンアカウントでサブスクライブしていることを確認します。[AWS Organizations](#) を使用している場合は、管理アカウントでサブスクライブしていることを確認します。現在、Q Developer は、AWS Organizations のメンバーアカウントでサブスクライブしているユーザーのプロンプトログ記録をサポートしていません。
- 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す。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、「[Amazon Q Developer プロファイルとは](#)」および「[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。
 - 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。
 - 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": "111122223333"
          },
          "ArnLike": {
            "aws:SourceArn": "arn:aws:codewhisperer:us-east-1:111122223333:*"
          }
        }
      }
    ]
  }
}

```

(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。

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、「[管理者に Amazon Q Developer コンソールの使用を許可する](#)」で定義されたアクセス許可が必要です。

 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でサブスクライブしているユーザーではプロンプトログ記録がサポートされていないため、メンバーアカウント管理者はプロンプトログ記録を有効にできません。

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。

各例に続いて、ログファイルのフィールドを説明した表を示します。

プロンプトログの詳細については、「[Amazon Q Developer でのユーザープロンプトのログ記録](#)」を参照してください。

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る際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,
'XXXXXXX', {\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"
      }
    }
  ]
}
```

次の表は、インライン提案のログファイルのフィールドを示しています。

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。

フィールド名	説明
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ろ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。
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。タイムスタンプでは協定世界時 (UTC) を使用します。
generateCompletionsEventResponse	Amazon Q によって生成されたインライン提案を説明します。
completions	Amazon Q によって提供され、ユーザーが受け入れ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際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\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\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\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\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\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\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          /// <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          // Add a canned ACL.\n          CannedACL
= S3CannedACL.LogDeliveryWrite,\n          };\n          var response =
await client.PutBucketAsync(request);\n          return response.HttpStatusCode
== System.Net.HttpStatusCode.OK;\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          return response.AccessControlList;\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            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            };  
acl.AddGrant(fullControlGrant.Grantee,  
S3Permission.FULL_CONTROL);\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            // Create a new ACL.\n            var newAcl = new S3AccessControlList\n            {\n                Grants = new  
List<S3Grant> { grantUsingEmail, grantLogDeliveryGroup },\n                Owner =  
owner,\n            };\n            // Set the new ACL. We're throwing away the  
response here.\n            _ = await client.PutACLSync(new PutACLRequest\n            {\n                BucketName = bucketName,\n                Key = keyName,\n                AccessControlList = newAcl,\n            });\n        }\n    }\n}\n\n},\n\n\"requestId\": \"dad38fc0-815c-45f7-970a-db916cb7f131\"\n\n}\n\n]\n}
```

フィールド名	説明
records	プロンプトと応答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にユーザーが入力した プロンプトを説明します。
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、「 Amazon Q Developer とのインラインでのチャット 」を参照してください。
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。/ の場合
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。タイムスタンプでは協定世界時 (UTC) を使用します。
generateAssistantResponseEventResponse	Amazon Q によって生成された応答を説明します。
assistantResponse	Amazon Q がユーザーのプロンプトに提供した応答を示します。

フィールド名	説明
followupPrompts	応答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	応答に関連付けられたメタデータを説明します。
conversationId	応答の会話 ID を示します。会話 ID は、チャットセッションのメッセージをグループ化します。
utteranceId	応答の発話 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であるプロンプトと別のプロンプトを区別するラベルです。
codeReferenceEvents	応答に含まれるコードリファレンスへのリンクを説明します。
supplementaryWebLinksEvent	応答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応答の 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際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	プロンプトと応答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 プロンプトを説明します。
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、「 Amazon Q Developer とのインラインでのチャット 」を参照してください。
conversationId	応答の会話 ID を示します。会話 ID は、チャットセッションの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。/ の場合
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。

フィールド名	説明
timeStamp	ユーザーがプロンプトを入力した日時を示します。タイムスタンプでは協定世界時 (UTC) を使用します。
startTaskAssistCodeGenerationEventResponse	Amazon Q によって生成された応答について説明します。現在、 /dev コマンドへの応答の記録はサポートされていないため、フィールドには応答は含まれません。
assistantResponse	Amazon Q がユーザーのプロンプトに提供した応答を示します。
requestId	イベントを一意に識別する応答の ID。

Amazon Q Developer がサポートされているリージョン

Note

ユーザーが行ったリクエストに対して、Amazon Q Developer が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必要がある場合、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。AWS リージョンを有効または無効にする方法の詳細については、「AWS アカウント管理 リファレンスガイド」の「[アカウントで利用できる AWS リージョンの指定](#)」を参照してください。

データは、Amazon Q Developer を使用してい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、「[クロスリージョン処理](#)」を参照してください。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](#)に関する説明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、以下の AWS リージョンにおいて、AWS マネジメントコンソール、AWS Console Mobile Application、AWS ウェブサイト、AWS Documentation ウェブサイト、および統合チャットアプリケーションで利用できます。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](#)の説明を参照してください。

Note

AWS マネジメントコンソール内の Amazon Q Developer サービスは、「[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](#)」で説明されているリージョンでのみ使用できます。管理者として Amazon Q Developer の設定を管理するには、Amazon Q Developer サービスに移動し、リージョンセクター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、チャットやその他の Amazon Q コンソールの機能を使用できません。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](#)に関する説明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の解決に役立ちます。各セクションには、問題の説明、考えられる原因、ステップバイステップの解決策、検証手順が明確に示されており、すぐに業務を続けることができます。

特定の問題の詳細を確認する前に、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 AWS 安定していることを確認します。
- 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](#) で最新であることを確認する
- 問題が解決しない場合は、IDE を再起動するか、「/quit」と入力して 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:

- [ログへのアクセスの概要](#)
- [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](#)

ログへのアクセスの概要

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 IDE 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の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 ID、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 (VS Code または JetBrains) で Amazon Q チャットパネルを開きます。
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. 表示される機密情報に関する警告を確認します。
4. ログファイルの場所は、レビュー用の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 拡張ログのレビューでは、以下を確認します。

- 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確認します。
- 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。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/
Server	%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 の初期化とスタートアップオペレーション
- AWS SDK 呼び出し (Cognito ID、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 - 以下を含むチャットアプリケーション固有のログ。

- チャットアプリケーションのエラーとステータス処理の問題
- MCP (モデルコンテキストプロトコル) サーバー管理および接続エラー
- 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 - モデルコンテキストプロトコルサーバーログ (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

スコープと詳細の違い:

- 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、には送信されません AWS。すべてのログファイルは、詳細フラグがなくても、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、このアプローチに従います。

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します。
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ます。

macOS/Linux の場合:

```
less -r $XDG_RUNTIME_DIR/qlog/qchat.log  
less -r $XDG_RUNTIME_DIR/qlog/chat.log
```

macOS の場合 (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。

エラー

通常のコ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はここから始めます。

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オペレーションメッセージ

DEBUG

より詳細な調査に役立つ詳細な技術情報

Amazon Q CLI ログを調べる際は、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 - チャット処理とステータス管理エラー
- ERROR chat_cli::cli::agent - 移行および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
- ERROR fig_telemetry - システムテレメトリおよび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とログエントリの関連
- 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 ID - 特定の API コールとその結果の追跡
- 接続ステータス - ネットワーク接続と認証ステータスのモニタリング

i Tip

grep、awk、または検索機能を備えた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や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

ログで発生する可能性がある一般的な問題とその一般的な解決策は次のとおりです。

MCP サーバー接続エラー

ログパターン (qchat.log 内): 「Background listening thread for client [server-name]: RecvError(Closed)」または「All senders dropped for transport layer」

解決策: MCP サーバード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 内): 「An error occurred processing the current state err=Interrupted { tool_uses: None }」

解決策: これは、チャット操作がユーザーによってキャンセルされ (Ctrl+C など)、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 内): 「Failed to send cw telemetry event 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でない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 内): 「Migration did not happen for the following reason: Aborting migration」

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 内): AWS SDK オペレーションでの認証関連エラー

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): 「Connection timeout」、「Network unreachable」、または HTTP 接続の失敗

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 内): 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する 再認証が必要になる場合があります

ログ分析に関するヘルプの取得

ログ分析や問題の解決について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それを再現する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 マネジメントコンソール、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](#)
-
- [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
サブスクリプションのアカウント制限を更新しました	1 つの AWS リージョン組織あたり最大 20 個のアカウントでユーザーをサブスクライブできるようになりました。10 個から増加しました。	2026 年 2 月 18 日
GitLab の非推奨コード変換	Java モダナイゼーションのコード変換機能は、 GitLab DuoAmazon Q Developer で用に廃止されました。/q transform クイックアクションと CI/CD パイプラインのカスタマイズドキュメントは削除されました。	2026 年 1 月 16 日
非推奨の GitLab クイックアクションを削除しました	GitLab /q dev (revise) クイックアクションドキュメントの /q test コマンドを使用して、箇条書きとユニットテスト生成セクション全体を削除しました。 GitLab	2026 年 1 月 13 日
GitHub 機能開発ワークフローを更新	従来の GitHub レビュープロセスではなく、/q コマンドで会話ベースのフィードバックを使用するように GitHub 機能開発ワークフローを更新しました。ユーザーは、プルリクエストの会話で自然言語コマンドを通じてフィードバック	2025 年 12 月 29 日

を提供できるようになりました。

[GitHub アプリのインストール登録手順を更新しました](#)

[Amazon Q Developer コンソールトピックの使用制限の引き上げと詳細の設定](#)を更新し、Kiro と Amazon Q Developer の共有コンソールを操作する手順を追加しました。

2025 年 12 月 18 日

[の非推奨コード変換 GitHub](#)

Java モダナイゼーションのコード変換機能は、[Amazon Q Developer for GitHub](#)では廃止されました。

2025 年 12 月 15 日

[Q CLI が Kiro CLI になる](#)

Q CLI が Kiro CLI になりました。

2025 年 11 月 17 日

[マネージドポリシーの更新: AmazonQFullAccess、AmazonQDeveloperAccess、AWSServiceRoleForUserSubscriptions](#)

[AmazonQFullAccess](#) ポリシー、[AmazonQDeveloperAccess](#) ポリシー、および [AWSServiceRoleForUserSubscriptions](#) に追加のアクセス許可が追加されました。

2025 年 10 月 29 日

[IDE エージェントドキュメントを削除しました](#)

[エージェントチャット機能](#)は、IDE の /dev、/doc、/test、/review エージェントを置き換えました。[IDE のコードレビュー](#)の手順が更新されました。

2025 年 10 月 21 日

[プレースホルダーのタイトル](#)

プレースホルダーコンテンツ。

2025 年 10 月 21 日

IDE メモリバンクのドキュメントを追加

Amazon Q チャット用のメモリバンクを生成するためのドキュメントを追加しました。

2025 年 10 月 21 日

Q CLI 機能のドキュメントを追加

AI 支援、プロンプト管理、メッセージへの応答、プロジェクトルールを使用してエージェントを作成するためのドキュメントを追加しました。 <https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/command-line-prompts.html> <https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/command-line-reply-command.html> <https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/command-line-project-rules.html>

2025 年 10 月 3 日

拡張コンテキスト管理ドキュメント

エージェントリソース、セッションコンテキスト、ナレッジベースの比較表、決定フローチャート、ベストプラクティスを使用して、適切なコンテキストアプローチを選択するための包括的なガイドンスを追加しました。

2025 年 9 月 23 日

[Q CLI v1.16.0 機能ドキュメントを追加](#)

優先度ベースの選択、[レガシー MCP 設定のサポート](#)、新しいチャットコマンド (/tangent tail、 /changelog)、エージェント[関連の設定を含むエージェントのデフォルト動作](#)に関するドキュメントを追加しました。

2025 年 9 月 18 日

[コンテキストフックの廃止](#)

コンテキストフックは、[エージェントフックの代わりに廃止されます](#)。既存の設定はエージェントファイルに自動的に移行されます。

2025 年 9 月 17 日

[リモート MCP サーバーのサポートと組み込みツールのドキュメントを追加](#)

OAuth 認証と包括的な組み込みツール設定を備えた[リモート MCP サーバーの](#)ドキュメントを追加しました。

2025 年 9 月 17 日

[許可リストテーブルから Amazon S3 バケットを削除](#)

IDE でのソフトウェア開発やユニットテスト生成のために[Amazon S3 バケットを許可リスト](#)に登録する必要がなくなりました。

2025 年 9 月 9 日

[変換用コマンドラインツールの新しいバージョン](#)

[変換用コマンドラインツールの](#)最新バージョンには、バグ修正が含まれています。

2025 年 9 月 9 日

[Amazon Q Developer for GitHub のレビューエージェン
トのコンテンツを更新](#)

[Amazon Q Developer for GitHub](#) のコードレビューには、スレッド化された検出結果を含むコードレビューの概要が含まれます。プルリクエストコメントで Amazon Q Developer と検出結果についてやり取りできます。

2025 年 9 月 2 日

[Amazon Q Developer for GitHub のレビューセクション
に前提条件を追加](#)

[Amazon Q Developer for GitHub](#) のコードレビューは、GitHub の書き込み、メンテナンス、または管理ロールでのみ開始できます。

2025 年 9 月 2 日

[MCP の無効化](#)

[組織の MCP を無効化](#) することができます。

2025 年 8 月 21 日

[プロキシトピックを更新](#)

プロキシ自動構成 (PAC) ファイルがサポートされていないことを示すため、「[Amazon Q での企業プロキシの設定](#)」トピックを更新しました。

2025 年 8 月 19 日

[デプロイオプショントピック
を更新](#)

[デプロイオプシオン](#)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 [qct history 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](#)

「[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、新しい手順の追加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 Visual Studio Code の最近の Java 変換ジョブからアーティファクトを表示、管理、取得できます。

2025 年 8 月 6 日

[GitLab セルフマネージドインスタンスを更新](#)

[GitLab Duo with Amazon Q](#) の設定では、[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トピックを追加しました](#)

トラブルシューティングトピック、「[サブスクライブしているユーザーを表示できない](#)」を追加しました。

2025 年 8 月 1 日

[VSC でのエージェントチャットによるコードレビューの手順を追加](#)

Visual Studio Code でエージェントチャットを使用して、[コードレビューの実行](#)および[コードの問題に対処](#)の手順を追加しました。

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、および IDE のチャットでモデルを選択する手順を含む、「[モデルの選択](#)」トピックを追加しました。

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](#)

「[Amazon Q Developer のキーボードショートカット](#)」トピックを追加しました。このトピックでは、さまざまな IDE やプラットフォームで利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します。

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利用できるツール、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

[チャット履歴の圧縮トピックを追加](#)

「[Amazon Q Developer でのチャット履歴の圧縮](#)」トピックを追加しました。このトピックでは、IDE で Amazon Q Developer とチャットする際にコンテキストウィンドウの制限を管理する方法について説明します。

2025 年 7 月 30 日

[プロファイルの削除トピックを追加](#)

「[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手順を追加](#)

変換に Amazon Q コマンドラインツールを使用する際に、[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ー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ーの依存関係処理する新しい GitHub ワークフロー](#)ページを追加しました。

2025 年 7 月 16 日

[ユーザーアクティビティのメトリクストピックを更新](#)

「[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](#)機能に関連する新しい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](#)

「[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画像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画像](#)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に画像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](#)

[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 AWS アカウントにリンクできないことを示すメモを追加しました。

2025 年 7 月 7 日

[プロファイル共有セクションを更新](#)

「[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と誤って記載されていました。この文は削除され、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

[Amazon Q Developer for GitHub のコンテンツにスラッシュコマンドの詳細を追加](#)

[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、Amazon Q Developer for GitHub コンテンツへのリンクを提供するヘルプを呼び出すこともできます。

2025 年 6 月 30 日

[IDE の QCT CLI の詳細を更新](#)

[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、および LLM サマリー出力に関する詳細を追加しました。QCT CLI は <https://q.eu-central-1.amazonaws.com/> もサポートするようになりました。

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を目的として、[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](#)

[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ための KMS ポリシー](#)に、"`kms:Decrypt`"、"`kms:DescribeKey`"、"`kms:Encrypt`"、"`kms:GenerateDataKey`" を追加しました。

2025 年 6 月 25 日

[プロキシ設定手順を追加](#)

「[Amazon Q for Visual Studio Code での企業プロキシの設定](#)」セクションを追加しました。

2025 年 6 月 25 日

[ユーザーアクティビティレポートを更新](#)

[ユーザーアクティビティレポート](#)は UTC の午前 0 時に生成され、各ファイルに 1,000 ユーザー分のアクティビティ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](#)

Amazon Q の Visual Studio Code プラグインのバージョン 1.72.0 以降では、バケットの許可リストは不要であることを示すため「[Amazon S3 バケット URL および ARN の許可リストへの追加](#)」セクションを更新しました。

2025 年 6 月 24 日

[Amazon Q Developer の運用調査のリブランディング](#)

Amazon Q Developer の運用調査から CloudWatch 調査へのリブランディングを反映するため、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 チャットへのルールボタンの追加](#)

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を追加](#)

VS Code に[コンテキストのピン留め](#)機能を追加し、手動ピン留めとシステム追加コンテキスト管理のサポートにより、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できるようになりました。

2025 年 6 月 18 日

[IDE 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](#)の詳細と、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新しい変換要件を追加しました。複数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の Amazon Q とのチャット](#)

コンソールおよびチャットアプリケーションで、[Amazon Q のネットワークセキュリティについてチャット](#)することができます。

2025 年 6 月 17 日

[新しいデータストレージピックを追加](#)

「[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 \(Builder ID\) ユーザーのアップグレード](#)

個人アカウント (Builder ID) を持つユーザーは、[Pro ティアにアップグレード](#)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 」の使用不可のサブスクリプションステータスの説明を更新しました。	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	コマンドラインで Amazon Q が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
チャットコンテキストのサポートの JetBrains、Visual Studio、Eclipse IDE への拡張	Amazon Q は、JetBrains、Visual Studio、Eclipse IDE 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 IDE 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 IDE で使えます。	2025 年 6 月 5 日

[更新された GitHub 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。新しいコメントで /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

[チャットで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](#)

「[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 URL 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 コマンドライン以外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](#)

「[管理者にカスタマイズの作成を許可する](#)」トピックに、カスタマイズポリシーを作成する際に発生する可能性のあるアクセス許可エラーについての説明を追加しました。

2025 年 5 月 16 日

[トラブルシューティングタスクを更新](#)

「[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追加されました。

2025 年 5 月 15 日

[GitLab Duo および GitHub の新しいコードレビュー動作を更新](#)

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

[VPC エンドポイントを更新](#)

「[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

[チャットに画像サポートを追加](#)

Amazon Q は、画像モードで fs_read ツールを使用して、チャットセッションで画像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 コマンドおよび /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 AmazonQFullAccess ポリシーと AmazonQDeveloperAccess ポリシーへのアクセス許可が追加されました。	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、IAM アイデンティティセンターと 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追加されました。	2025 年 5 月 13 日
変換ウェブサイト削除	Amazon Q Developer 変換ウェブサイトが廃止され、それに伴い、関連ドキュメントが削除されました。	2025 年 5 月 12 日
用語の更新	このガイド全体で、ID 対応コンソールセッションという用語を ID 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	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

Amazon Q Developer for 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
マネージドポリシーを更新	アクセス許可が AmazonQFuIIAccess 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 内 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 AmazonQFuIIAccess 、 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 サポートの追加	CLI での MCP のサポートが追加されました。	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 ティア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	IDE で Amazon Q とチャットするときに 会話履歴 が保存されるようになりました。	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

[GitLab Duo with Amazon Q のオンボーディングとポリシーを更新](#)

[GitLab Duo with 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

[ダッシュボードアクセス許可を更新](#)

Amazon Q Developer ダッシュボードの表示に必要な[アクセス許可の一覧](#)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、および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](#)の内容を見直し改善しました。

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「[ツールのアクセス許可のドキュメント](#)」を改善しました。

2025 年 4 月 12 日

サブスクリプションエクスペリエンスを更新	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
Eclipse でのインラインチャットの利用	Eclipse の Amazon Q で インラインチャット を使用することができます。	2025 年 4 月 10 日
Amazon Q Developer プロファイルの欧州 (フランクフルト) での使用開始	Amazon Q Developer にサブスクライブする際、 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 機能を CLI に追加	/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。	2025 年 4 月 9 日
GitLab Duo with Amazon Q の更新	GitLab Duo with Amazon Q 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。/fix 機能は削除されました。	2025 年 4 月 8 日
変換に関する E メール通知	変換関連の更新に関する E メール通知 を受け取ることができます。	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 の各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 の各トピックが修正されました。	2025 年 3 月 25 日
ポリシー例の更新	「 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 」のポリシー例に sso:CreateInstance アクセス許可に関する記載が追加されました。	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回答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](#)

「[管理者に Amazon Q コンソールの使用を許可する](#)」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、AWS CLIを介した IAM による認証のサポートが含まれています。

2025 年 2 月 28 日

[Pro ティアへのアップグレード](#)

「[Amazon Q Developer 無料利用枠](#)」トピックに Pro ティアへのアップグレード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	「 Amazon Q Developer ダッシュボードの使用状況メトリクスの説明 」トピックが追加されました。	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、AWS リージョン 間で呼び出し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	サブスクリプションに関する 2 つのトピック、「 Amazon Q Developer サブスクリプションのステータス 」および「 Amazon Q Developer サブスクリプションの集約の表示 」が追加されました。	2025 年 2 月 19 日
Amazon Q Developer in chat applications に関する章	「Amazon Q Developer in chat applications」の章では、Amazon Q Developer in chat applications について説明します。 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 での 変換されたコード の保持時間が 24 時間から 30 日間に変更されました。	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 機能拡張	Amazon Q は、 /dev コマンドに回答して 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	カスタマイズで任意の数のリポジトリを含めることができることを「 カスタマイズの作成 」トピックに追加しました。	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	ユーザーアクティビティレポート に 新しいメトリクス が追加されました。	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	「 Amazon Q にリソースのトラブルシューティングを依頼する 」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を「 組織インスタンスを使用して Amazon Q Developer Pro ティアにユーザーをサブスクライブする 」セクションに追加しました。	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 Duo with Amazon Q	概念、開始手順、トラブルシューティングなど、 GitLab Duo with Amazon Q に関する情報。	2024 年 12 月 3 日
IDE でのドキュメントの生成	Amazon Q は、サポートされている IDE で コードの README を生成 できます。	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 (旧セキュリティスキャン) では、サポートされている IDE 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 から変換の変更を受け取ることを選択できます。	2024 年 11 月 27 日

Eclipse の Amazon Q	Amazon Q プラグイン は、Eclipse のプレビューで利用できます。	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 機能が一般提供されました。	2024 年 11 月 26 日
埋め込み SQL コードの変換	IDE で 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 コマンドラインでの Linux のサポート	Amazon Q コマンドライン は、Linux 環境をサポートするようになりました。Ubuntu 22 および 24 をサポートします。それ以外の場合は GNOME v42+ か、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 である環境で動作します。	2024 年 11 月 21 日

ユーザーのサブスクライブ	「 Amazon Q Developer Pro ティアへのアクセスのセットアップ 」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 機能が一般提供されるようになりました。また、.md、.mdx、.rst、および .txt 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。	2024 年 11 月 20 日
IAM アイデンティティセンターリージョンのサポート	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 に関する情報が含むセクションが追加されました。	2024 年 11 月 18 日
新しい言語の追加	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、およびカスタマーマネージドキー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 のクォータ	「 Pro ティアのクォータ 」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
新しいマネージドポリシー: AmazonQDeveloperAccess	AmazonQDeveloperAccess 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
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 セクションの明確化	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

新しい管理ポリシー: 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 管理者に権限を付与するポリシーが更新され、sso:ListProfiles が追加されました。	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

[新しい管理ポリシー: `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の Amazon Q Developer で設定された Slack および Microsoft Teams チャンネルに Amazon Q を追加する 方法が追加されました。	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。