aws

開発者ガイド

Amazon Kendra

Amazon Kendra: 開発者ガイド

Table of Contents

Amazon Kendra の概要

Amazon Kendra は、自然言語処理と高度な機械学習アルゴリズムを使用して、データから検索に関する質問に対する特定の回答を返すインテリジェントな検索サービスです。

従来のキーワードベースの検索とは異なり、Amazon Kendra はセマンティックおよびコンテキストの理解機能を使用して、ドキュメントが検索クエリに関連しているかどうかを判断します。質問に対する特定の回答が返され、ユーザーはヒトの専門家とのやりとりに近いエクスペリエンスを提供します。

> **Note**
>
> また、Amazon Kendra のセマンティック検索機能を使用すれば、別の検索サービスの結果を再度ランク付けできます。詳細については、「Amazon Kendra Intelligent Ranking」を参照してください。

Amazon Kendra では、複数のデータリポジトリをインデックスに接続し、ドキュメントを取り込んでクローリングすることにより、統一された検索エクスペリエンスを実現できます。ドキュメントのメタデータを利用して、ユーザーに機能豊富でカスタマイズされた検索環境を提供できるため、ユーザーはクエリに対する正しい回答を効率的に見つけることができます。

Amazon Kendraとは?

Amazon Kendra のクエリ

Amazon Kendra では、次のようなタイプのクエリが可能です。

Factoid 型の質問 - 誰が、何を、いつ、どこでの単純な質問。例:「シアトルに最も近いサービスセンターはどこですか?」 Factoid 型の質問には、事実ベースの回答を 1 単語または 1 語句で返すことができます。回答は、FAQ またはインデックスが作成されたドキュメントから取得されます。

説明的な質問 - 1 文、1 節、またはドキュメント全体が答えとなる質問。例:「Echo Plus をネットワークに接続するにはどうすればよいですか?」 または「低所得世帯向けの税制上の優遇措置を受けるにはどうすればよいですか?」

キーワードおよび自然言語に関する質問 - 意味がはっきりしない、複雑な会話内容を含む質問。例えば、キーノートのアドレス。Amazon Kendra では、複数の文脈上の意味を持つ「address」のような単語が見つかると、検索クエリの意味を正しく推測し、関連情報を返します。

Amazon Kendra の利点

Amazon Kendra は非常にスケーラブルで、パフォーマンス要求を満たすことができ、Amazon S3 や Amazon Lex などの他の AWS のサービスと緊密に統合され、エンタープライズレベルのセキュリティを提供します。Amazon Kendra を使用する利点のいくつかを以下に示します。

シンプルさ - Amazon Kendra は、検索するドキュメントを管理するためのコンソールと API を提供します。シンプルな検索 API を使用して、Amazon Kendra をウェブサイトやモバイルアプリケーションなどのクライアントアプリケーションに統合できます。

接続性 - Amazon Kendra は、Microsoft SharePoint などのサードパーティのデータリポジトリまたはデータソースに接続できます。データソースを使用して、ドキュメントのインデックス作成と検索を簡単に行うことができます。

正確性 - キーワード検索を使用する従来の検索サービスとは異なり、Amazon Kendra は質問のコンテキストを理解し、クエリに最も関連性の高い単語、スニペット、またはドキュメントを返します。Amazon Kendra は機械学習を使用して、検索結果を改善します。

セキュリティ - Amazon Kendra は、安全性の高いエンタープライズ検索エクスペリエンスを提供します。検索結果には、組織のセキュリティモデルが反映され、ドキュメントへのユーザーまたはグループのアクセスに基づいてフィルタリングできます。お客様は、ユーザーのアクセスを認証し、認可する責任を負うものとします。

Amazon Kendra Editions

Amazon Kendra Developer Edition と Enterprise Edition の 2 つのバージョンがあります。次の表は、これら 2 つの機能の概要と相違点をまとめたものです。

Amazon Kendra Developer Edition	Amazon Kendra Enterprise Edition
Amazon Kendra Developer Edition は、Amazon Kendra のすべての機能を低コストで提供します。	Amazon Kendra Enterprise Edition は、Amazon Kendra のすべての機能を備え、本番環境向けに設計されています。
最適なユースケース	最適なユースケース

Amazon Kendra Developer Edition

- Amazon Kendra がドキュメントにどのようにインデックスを作成するかを調べる
- 機能を試用する
- Amazon Kendra を使用するアプリケーションの開発

特徴

- 750 時間の使用が可能な無料利用枠
- 最大 5 つのインデックスで、それぞれ最大 5 つのデータソース
- 10,000 ドキュメントまたは 3 GB の抽出されたテキスト
- 1 日あたり約 4,000 クエリまたは 1 秒あたり 0.05 クエリ
- 1 つのアベイラビリティーゾーン (AZ) で実行。「アベイラビリティーゾーン」(AWS リージョンのデータセンター) を参照してください。

機能制限

- 本番環境のアプリケーションには使用できません。
- レイテンシーや可用性は保証されません。

Amazon Kendra Enterprise Edition

- エンタープライズドキュメントライブラリ全体のインデックス作成
- 本番環境のアプリケーションへのデプロイ

特徴

- 最大 5 つのインデックスで、それぞれ最大 50 つのデータソース
- 100,000 ドキュメントまたは 30 GB の抽出されたテキスト
- 1 日あたり約 8,000 クエリまたは 1 秒あたり 0.1 クエリ
- 3 つのアベイラビリティーゾーン (AZ) で実行。「アベイラビリティーゾーン」(AWS リージョンのデータセンター) を参照してください。

> **Note**
>
> Service Quotas コンソールを使用して IAM クォータの増加をリクエストできます。

機能制限

- なし

> **Note**
>
> Amazon Kendra でサポートされているリージョン、エンドポイント、サービスクォータのリストについては、「Amazon Kendra エンドポイントとクォータ」を参照してください。

Amazon Kendra の料金

最初の 30 日間は 750 時間まで使用できる Amazon Kendra Developer Edition が無料で開始できます。

試用期間が終了すると、プロビジョニングされた Amazon Kendra インデックスが空で、クエリが実行されていない場合でも、プロビジョニングされたすべての Amazon Kendra インデックスに対して料金が発生します。試用期間が終了すると、Amazon Kendra データソースを使用したドキュメントのスキャンおよび同期には追加料金がかかります。

課金および料金の詳細なリストについては、「Amazon Kendra の料金」を参照してください。

Amazon Kendra を初めてお使いになる方向けの情報

Amazon Kendra を初めて使用する方には、次のセクションを順に読むことをお勧めします。

1	2	3	4	5	6
Amazon Kendra の働き	開始	インデックスの作成	バッチアップロードを使用したドキュメントのインデックスへの直接追加	データソースコネクタの作成	インデックスの検索
Amazon Kendra コンポーネントを導入し、それらを使用して検索ソリューションを作成する方法について説明します。	アカウントを設定し、Amazon Kendra 検索 API をテストする方法について説明します。	Amazon Kendra を使用して検索インデックスを作成し、データソースを追加してドキュメントを同期する方法について説明します。	Amazon Kendra インデックスにドキュメントを直接追加する方法について説明します。	データリポジトリから Amazon Kendra インデックスにドキュメントを追加する方法について説明します。	Amazon Kendra 検索 API を使用してインデックスを検索する方法について説明します。

Amazon Kendra の働き

Amazon Kendra アプリケーションに検索機能を提供します。ドキュメントに直接インデックスを作成、またはサードパーティのドキュメントリポジトリからインデックスを作成して、ユーザーに関連情報をインテリジェントに提供します。 Amazon Kendra を使用して、さまざまなタイプのドキュメントの更新可能なインデックスを作成できます。サポートされているドキュメントタイプのリストについては、「ドキュメントの種類 Amazon Kendra」を参照してください。

Amazon Kendra 他のサービスと統合できます。たとえば、Amazon LexAmazon Kendra チャットボットに検索機能を追加して、ユーザーの質問に役立つ回答を提供することができます。Amazon Simple Storage ServiceAmazon Kendra バケットをドキュメントに接続してインデックスを作成するためのデータソースとして使用できます。また、AWS Identity and Access Management を使用してリソースへのアクセスポリシーやアクセス許可を設定できます。

Amazon Kendra には次のコンポーネントがあります。

- ドキュメントを格納して検索できるようにするインデックス。
- ドキュメントを保存して Amazon Kendra を接続するデータソース。データソースをインデックスと自動的に同期して、 Amazon Kendra インデックスがソースリポジトリで常に最新の状態に保たれるようにすることができます。
- ドキュメントをインデックスに直接追加するドキュメント追加 API。

コンソールまたは API Amazon Kendra から使用できます。インデックスを作成、編集、削除できます。インデックスを削除すると、そのデータソースコネクタがすべて削除され、からすべてのドキュメント情報が完全に削除されます。 Amazon Kendra

トピック

- [Index] (インデックス)
- ドキュメント
- データソース
- クエリ
- タグ

[Index] (インデックス)

インデックスにはドキュメントの内容が格納され、ドキュメントを検索できるように構造化されています。インデックスにドキュメントを追加する方法は、ドキュメントの保存方法によって異なります。

- Amazon S3 バケットや Microsoft SharePoint サイトなどのリポジトリにドキュメントを保存する場合は、データソースコネクタを使用してリポジトリからドキュメントにインデックスを付けます。
- ドキュメントをリポジトリに保存しない場合は、BatchPutDocumentAPI を使用してドキュメントに直接インデックスを付けます。
- Amazon Kendra (Amazon S3) バケットに保存する必要がある、よくある質問と回答は、バケットからアップロードします。

インデックスを作成するには、 Amazon Kendra コンソール、 AWS CLI、または AWS SDK を使用します。インデックス作成可能なドキュメントのタイプについては、「Document types」を参照してください。

Amazon Kendra 予約済みまたは共通のドキュメントフィールドを使用する

UpdateIndex API では、`DocumentMetadataConfigurationUpdates`予約済みインデックスフィールド名を使用して指定し、対応するドキュメント属性/フィールド名にマッピングすることで、 Amazon Kendra 予約済みフィールドまたは共通フィールドを作成できます。カスタムフィールドも作成できます。データソースコネクタを使用する場合、ほとんどの場合、データソースドキュメントのフィールドをインデックスフィールドにマップするフィールドマッピングが含まれています。Amazon Kendra コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。

`Search` オブジェクトを設定して、フィールドを表示可能、ファセット可能、検索可能、ソート可能のいずれかに設定できます。特定のフィールド値にマッピングされたブースト、新しさ、重要度の値に適用するフィールドのランク順序、ブースト期間、または期間を設定するように `Relevance` オブジェクトを設定できます。コンソールを使用する場合は、ナビゲーションメニューのファセットオプションを選択して、フィールドの検索設定をセットできます。関連性調整を設定するには、ナビゲーションメニューでインデックスを検索するオプションを選択し、クエリを入力し、サイドパネルのオプションを使用して検索の関連性を調整します。フィールドを作成すると、フィールドタイプを変更することはできません。

Amazon Kendra には、次のような予約済みまたは共通のドキュメントフィールドがあり、それらを使用できます。

- _authors - ドキュメントの内容を担当する 1 人以上の作成者のリスト。
- _category - ドキュメントを特定のグループに配置するカテゴリ。
- _created_at - ドキュメントが作成された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _data_source_id - ドキュメントを含むデータソースの識別子。
- _document_body - ドキュメントのコンテンツ。
- _document_id - ドキュメントの一意の識別子。
- _document_title - ドキュメントのタイトル。
- _excerpt_page_number - ドキュメントの抜粋が表示される PDF ファイルのページ番号。2020 年 9 月 8 日より前にインデックスが作成された場合、この属性を使用する前に、ドキュメントのインデックスを再作成する必要があります。
- _faq_id - これが質疑応答タイプのドキュメント (よくある質問) の場合、よくある質問の固有識別子です。
- _file_type - pdf や doc など、ドキュメントのファイルタイプ。
- _last_updated_at - ドキュメントが最後に更新された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _source_uri - ドキュメントが利用可能な URI。例えば、会社のウェブサイト上のドキュメントの URI などです。
- _version - ドキュメントの特定のバージョンの識別子。
- _view_count - ドキュメントが表示された回数。
- _language_code (文字列) - ドキュメントに適用される言語のコード。言語を指定しないと、デフォルトで英語になります。コードを含む、サポートされている言語の詳細については、英語以外の言語でドキュメントを追加するを参照してください。

カスタムフィールドの場合、予約フィールドまたは共通フィールドを作成する場合と同じように、UpdateIndex API で DocumentMetadataConfigurationUpdates を使用してこれらのフィールドを作成します。カスタムフィールドには適切なデータタイプを設定する必要があります。コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。一部のデー

タソースは、新しいフィールドやカスタムフィールドの追加をサポートしていません。フィールドを作成すると、フィールドタイプを変更することはできません。

カスタムフィールドには以下のタイプを設定できます。

- 日付
- 数
- 文字列
- 文字列リスト

BatchPutDocumentAPI を使用してインデックスにドキュメントを追加した場合は、`Attributes`ドキュメントのフィールド/属性を一覧表示し、オブジェクトを使用してフィールドを作成します。`DocumentAttribute`

Amazon S3 データソースからインデックスされたドキュメントの場合、フィールド情報を含む JSON メタデータファイルを使用してフィールドを作成します。

サポートされているデータベースをデータソースとして使用する場合は、フィールドマッピングオプションを使用してフィールドを設定できます。

インデックスの検索

インデックスの作成後は、ドキュメントの検索を開始できます。詳細については、「Searching indexes」を参照してください。

ドキュメント

このセクションでは、サポートするさまざまなドキュメント形式と、 Amazon Kendra ドキュメントのさまざまなフィールドや属性のインデックスを作成する方法について説明します。

トピック

- ドキュメントタイプまたは書式
- ドキュメントの属性またはフィールド

ドキュメントタイプまたは書式

Amazon Kendra PDF、HTML、 PowerPoint Word などの一般的なドキュメントタイプまたは形式をサポートします。インデックスには複数のドキュメント形式を含めることができます。

Amazon Kendra ドキュメント内のコンテンツを抽出して、ドキュメントを検索可能にします。抽出されたテキストとドキュメント内の表形式のコンテンツ (HTML テーブル) での検索が最適化されるように、ドキュメントが解析されます。つまり、ドキュメントを、検索で使用するフィールドまたは属性に構築します。最終更新日などのドキュメントメタメタデータは、検索に役立つフィールドになります。

ドキュメントは行と列に編成できます。例えば、各ドキュメントは行で、タイトルや本文コンテンツなどの各ドキュメントフィールド/属性は列にします。例えば、データベースをデータソースとして使用する場合、データは行と列に構築または整理する必要があります。

ドキュメントをインデックスに追加するには、次の方法があります。

- BatchPutDocument API
- データソースコネクタ

FAQ ファイルを追加する場合は、CreateFaqAPI を使用してバケットに保存されているファイルを追加します。 Amazon S3 基本的な CSV 形式、ヘッダーにカスタムフィールド/属性を含む CSV 形式、カスタムフィールドを含む JSON 形式から選択できます。デフォルトの形式は基本的な CSV です。

以下では、サポートされている各ドキュメント形式と、 Amazon Kendra がドキュメントのインデックス作成時に各形式をどのように処理するかについて説明します。

ドキュメントの形式	処理方法	ドキュメントの処理方法	元の構造
ポータブルドキュメント形式 (PDF)	HTML	HTML に変換してから、コンテンツを抽出します。	構造化されない
HyperText マークアップ言語 (HTML)	HTML	HTML タグをフィルターで除外して、コンテンツを抽出します。コンテンツはメインの HTML 開始タグと終了タグ (<HTML>content</	半構造化

ドキュメントの形式	処理方法	ドキュメントの処理方法	元の構造
		HTML>) の間にある必要があります。	
Extensible Markup Language (XML)	XML	XML タグをフィルターで除外して、コンテンツを抽出します。	半構造化
拡張スタイルシート言語変換 (XSLT)	XSLT	タグをフィルターで除外して、コンテンツを抽出します。	半構造化
MarkDown (MD)	プレーンテキスト	MarkDown コンテンツは構文を含めて抽出されます。	半構造化
カンマ区切り値 (CSV)	CSV	各セルから抽出されたコンテンツで、1 つのファイルが 1 つのドキュメント結果として扱われます。	よくある質問ファイルの場合は構造化、それ以外は半構造化
Microsoft Excel (XLS および XLSX)	XLS および XLSX	各セルから抽出されたコンテンツで、1 つのファイルが 1 つのドキュメント結果として扱われます。	半構造化
JavaScript オブジェクト表記 (JSON)	プレーンテキスト	コンテンツは JSON 構文を含めた状態で抽出されます。	半構造化
リッチテキスト形式 (RTF)	RTF	RTF 構文はフィルターで除外され、内容が抽出されます。	半構造化

ドキュメントの形式	処理方法	ドキュメントの処理方法	元の構造
Microsoft PowerPoint (PPT)	PPT	PowerPoint 検索用にスライドから抽出されるのはテキストコンテンツだけです。イメージやその他のコンテンツは抽出されません。	構造化されない
Microsoft Word (DOCX)	DOCX	検索のために Word ページからテキストコンテンツのみが抽出されます。イメージやその他のコンテンツは抽出されません。	構造化されない
プレーンテキスト (TXT)	TXT	テキストドキュメント内のすべてのテキストが抽出されます。	構造化されない

ドキュメントの属性またはフィールド

ドキュメントには属性またはフィールドが関連付けられています。ドキュメントのフィールドは、ドキュメントのプロパティ、またはドキュメントの構造に含まれる属性です。たとえば、各ドキュメントにタイトル、本文、著者が含まれている場合があります。特定の文書にカスタムフィールドを追加することもできます。例えば、インデックスが税務文書を検索する場合、W-2、1099 などの税文書の種類にカスタムフィールドを指定できます。

クエリでドキュメント属性を使用するには、その前にインデックスフィールドにマッピングする必要があります。例えば、タイトルフィールドをフィールド `_document_title` にマッピングできます。詳細については、「Mapping fields」を参照してください。新しいフィールドを追加するには、フィールドをマッピングするインデックスフィールドを作成する必要があります。インデックスフィールドは、コンソールまたは UpdateIndexAPI を使用して作成します。

ドキュメントフィールドを使用して、レスポンスをフィルタリングし、ファセット検索結果を作成できます。例えば、特定のバージョンのドキュメントのみを返すように応答をフィルタリングしたり、検索条件に一致する 1099 タイプの税務文書のみを返すように検索をフィルタリングできます。詳細については、「Filtering and facet search」を参照してください。

ドキュメントフィールドを使用して、クエリレスポンスを手動で調整することもできます。たとえば、タイトルフィールドの重要度を高めて、 Amazon Kendra レスポンスで返すドキュメントを決定する際にフィールドに割り当てられる重みを増やすことができます。詳細については、「Tuning search relevance」を参照してください。

ドキュメントをインデックスに直接追加する場合は、BatchPutDocumentAPI の Document 入力パラメータでフィールドを指定します。DocumentAttributeカスタムフィールド値はオブジェクト配列で指定します。データソースを使用している場合、ドキュメントフィールドを追加するために使用する方法は、データソースによって異なります。詳細については、データソースフィールドのマッピングを参照してください。

Amazon Kendra 予約済みまたは共通のドキュメントフィールドを使用する

UpdateIndex API では、`DocumentMetadataConfigurationUpdates`予約済みインデックスフィールド名を使用して指定し、対応するドキュメント属性/フィールド名にマッピングすることで、 Amazon Kendra 予約済みフィールドまたは共通フィールドを作成できます。カスタムフィールドも作成できます。データソースコネクタを使用する場合、ほとんどの場合、データソースドキュメントのフィールドをインデックスフィールドにマップするフィールドマッピングが含まれています。Amazon Kendra コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。

`Search` オブジェクトを設定して、フィールドを表示可能、ファセット可能、検索可能、ソート可能のいずれかに設定できます。特定のフィールド値にマッピングされたブースト、新しさ、重要度の値に適用するフィールドのランク順序、ブースト期間、または期間を設定するように `Relevance` オブジェクトを設定できます。コンソールを使用する場合は、ナビゲーションメニューのファセットオプションを選択して、フィールドの検索設定をセットできます。関連性調整を設定するには、ナビゲーションメニューでインデックスを検索するオプションを選択し、クエリを入力し、サイドパネルのオプションを使用して検索の関連性を調整します。フィールドを作成すると、フィールドタイプを変更することはできません。

Amazon Kendra には、次のような予約済みまたは共通のドキュメントフィールドがあり、それらを使用できます。

- _authors - ドキュメントの内容を担当する 1 人以上の作成者のリスト。
- _category - ドキュメントを特定のグループに配置するカテゴリ。
- _created_at - ドキュメントが作成された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _data_source_id - ドキュメントを含むデータソースの識別子。
- _document_body - ドキュメントのコンテンツ。
- _document_id - ドキュメントの一意の識別子。
- _document_title - ドキュメントのタイトル。
- _excerpt_page_number - ドキュメントの抜粋が表示される PDF ファイルのページ番号。2020 年 9 月 8 日より前にインデックスが作成された場合、この属性を使用する前に、ドキュメントのインデックスを再作成する必要があります。
- _faq_id - これが質疑応答タイプのドキュメント (よくある質問) の場合、よくある質問の固有識別子です。
- _file_type - pdf や doc など、ドキュメントのファイルタイプ。
- _last_updated_at - ドキュメントが最後に更新された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _source_uri - ドキュメントが利用可能な URI。例えば、会社のウェブサイト上のドキュメントの URI などです。
- _version - ドキュメントの特定のバージョンの識別子。
- _view_count - ドキュメントが表示された回数。
- _language_code (文字列) - ドキュメントに適用される言語のコード。言語を指定しないと、デフォルトで英語になります。コードを含む、サポートされている言語の詳細については、英語以外の言語でドキュメントを追加するを参照してください。

カスタムフィールドの場合、予約フィールドまたは共通フィールドを作成する場合と同じように、UpdateIndex API で DocumentMetadataConfigurationUpdates を使用してこれらのフィールドを作成します。カスタムフィールドには適切なデータタイプを設定する必要があります。コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。一部のデータソースは、新しいフィールドやカスタムフィールドの追加をサポートしていません。フィールドを作成すると、フィールドタイプを変更することはできません。

カスタムフィールドには以下のタイプを設定できます。

- 日付
- 数
- 文字列
- 文字列リスト

BatchPutDocumentAPI を使用してインデックスにドキュメントを追加した場合は、`Attributes`ドキュメントのフィールド/属性を一覧表示し、オブジェクトを使用してフィールドを作成します。`DocumentAttribute`

Amazon S3 データソースからインデックスされたドキュメントの場合、フィールド情報を含む JSON メタデータファイルを使用してフィールドを作成します。

サポートされているデータベースをデータソースとして使用する場合は、フィールドマッピングオプションを使用してフィールドを設定できます。

データソース

データソースは、 Amazon Kendra ドキュメントやコンテンツに接続してインデックスを作成するデータリポジトリまたは場所です。たとえば、Microsoft Amazon Kendra に接続して、 SharePoint このソースに保存されているドキュメントをクロールしてインデックスを作成するように構成できます。クロールする URL Amazon Kendra を指定して Web ページのインデックスを作成することもできます。 Amazon Kendra データソースをインデックスと自動的に同期して、データソース内で追加、更新、または削除されたドキュメントがインデックスでも追加、更新、または削除されるようにすることができます。

サポートされているデータソースは以下の通りです。

- Adobe Experience Manager
- Alfresco
- Aurora (MySQL)
- Aurora (PostgreSQL)
- Amazon FSx (Windows)
- Amazon FSx (NetApp ONTAP)
- データベースデータソース

- Amazon RDS (Microsoft SQL Server)
- Amazon RDS (MySQL)
- Amazon RDS (オラクル)
- Amazon RDS (PostgreSQL)
- Amazon S3 バケット
- Amazon Kendra Web クローラー
- Amazon WorkDocs
- [Box] (ボックス)
- Confluence
- カスタムデータソース
- Dropbox
- Drupal
- GitHub
- Gmail
- Google Workspace ドライブ
- IBM DB2
- Jira
- Microsoft Exchange
- Microsoft OneDrive
- Microsoft SharePoint
- Microsoft Teams
- Microsoft SQL Server
- Microsoft Yammer
- MySQL
- Oracle Database
- PostgreSQL
- Quip
- Salesforce
- ServiceNow

- Slack
- Zendesk

でサポートされているドキュメントタイプまたは形式のリストについては、 Amazon Kendra 「ドキュメントタイプ」を参照してください。データソースからドキュメントにインデックスを付けるデータソースコネクタを作成する前に、まずインデックスを作成する必要があります。

> **Note**
>
> ドキュメントのインデックスを作成するには、データソースは必要ありません。バッチアップロードを使用すると、ドキュメントをインデックスに直接追加できます。詳細については、「Adding documents directly to an index」を参照してください。

Amazon Kendra コンソール、 AWS CLI、または SDK の使用方法については、「はじめに」を参照してください。

クエリ

回答を得るために、ユーザーはインデックスをクエリします。ユーザーはクエリで自然言語を使用できます。レスポンスは、最も良い答えを提供するインデックスにタイトル、テキストの抜粋、ドキュメントの場所などの情報を含みます。

Amazon Kendra ドキュメントの内容だけでなく、ドキュメントに関して提供されたすべての情報を使用して、ドキュメントがクエリに関連しているかどうかを判断します。たとえば、インデックスにドキュメントの最終更新日に関する情報が含まれている場合、 Amazon Kendra 最近更新されたドキュメントにはより高い関連性を割り当てるように指示できます。

クエリには、 Amazon Kendra そのフィルター条件を満たすドキュメントのみを返すように、応答をフィルターする方法の条件を含めることもできます。例えば、department というインデックスフィールドを作成した場合、部門フィールドが legal に設定されているドキュメントのみが返されるように応答をフィルタリングできます。詳細については、「Filtering search」を参照してください。

インデックス内の個々のフィールドの関連性をチューニングすることで、クエリの結果に影響を与えることができます。チューニングによって、結果に対するフィールドの重要性が変わります。例えば、new カテゴリを持つドキュメントの重要性を上げる場合、このカテゴリのドキュメントがレスポンスに含まれる可能性が高くなります。詳細については、「Tuning search relevance」を参照してください。

クエリの使用に関する詳細については、「Searching an index」を参照してください。

タグ

インデックス、データソース、よくある質問にタグまたはラベルを割り当てて、管理します。タグを使用して、 Amazon Kendra リソースをさまざまな方法で分類できます。例えば、目的、所有者、アプリケーション、または任意の組み合わせで分類します。タグはそれぞれ、1 つのキーと 1 つの値で構成されており、どちらもお客様側が定義します。

タグを使用すると、次のことができます。

- AWS リソースを特定して整理します。 AWS 多くのサービスがタグ付けをサポートしているため、異なるサービスのリソースに同じタグを割り当てて、リソースが関連していることを示すことができます。たとえば、インデックスと、 Amazon Lex そのインデックスを使用するボットに同じタグを付けることができます。
- コストの割り当て。 AWS Billing and Cost Management タグはダッシュボードで有効化します。AWS タグを使用してコストを分類し、毎月のコスト配分レポートを配信します。詳細については、「 AWS Billing and Cost Management について」の「コスト配分とタグ付け」を参照してください。
- リソースへのアクセス制御。 AWS Identity and Access Management (IAM) ポリシーのタグを使用して、 Amazon Kendra リソースへのアクセスを制御できます。 IAM これらのポリシーをロールまたはユーザーにアタッチして、タグベースのアクセス制御を有効にできます。詳細については、「Authorization based on tags」を参照してください。

、 AWS Command Line Interface (AWS CLI) AWS Management Console、または Amazon Kendra API を使用してタグを作成および管理できます。

リソースのタギング

Amazon Kendra コンソールを使用している場合は、リソースを作成するときにタグ付けすることも、後で追加することもできます。コンソールを使用して、タグを更新または削除することもできます。

AWS Command Line Interface (AWS CLI) または Amazon Kendra API を使用している場合は、以下の操作を使用してリソースのタグを管理します。

- CreateDataSource—データソースの作成時にタグを適用します。
- CreateFaq—FAQ を作成するときにタグを適用します。

- CreateIndex—インデックスを作成するときにタグを適用します。
- ListTagsForResource— リソースに関連付けられているタグを表示します。
- TagResource—リソースのタグを追加および変更します。
- UntagResource—リソースからタグを削除します。

タグの制限

Amazon Kendra リソースのタグには以下の制限が適用されます。

- タグの最大数 - 50
- キーの最大長 - 128 文字
- 最大値の長さ - 256 文字
- キーと値の有効な文字は、a ~ z、A ~ Z、スペース、特殊文字 (_ . : / = + - @) です。
- キーと値は大文字と小文字が区別されます
- `aws:` をキーのプレフィックスとして使用しないでください。 AWS 用に予約済みです。

Amazon Kendra のセットアップ

Amazon Kendra を使用する前に、Amazon Web Services (AWS) アカウントを持っている必要があります。 AWS アカウントを取得したら、Amazon Kendra コンソール、 (AWS CLI) 、 AWS Command Line Interface または AWS SDKs を使用して Amazon Kendra にアクセスできます。

このガイドには、 AWS CLI、Java、Python の例が含まれています。

トピック

にサインアップする AWS

Amazon Web Services (AWS) にサインアップすると AWS、Amazon Kendra を含む のすべてのサービスに アカウントが自動的にサインアップされます。料金は、使用するサービスの料金のみが請求されます。

AWS アカウントを既にお持ちの場合は、次のタスクに進んでください。 AWS アカウントをお持ちでない場合は、以下の手順に従ってアカウントを作成してください。

にサインアップするには AWS

1. https://aws.amazon.com を開き、 AWS アカウントの作成を選択します。
2. 画面上の指示に従ってアカウントの作成を完了します。12 桁の AWS アカウント番号を書き留めます。サインアップ手順の一環として、通話呼び出しを受け取り、電話のキーパッドを用いて PIN を入力することが求められます。
3. AWS Identity and Access Management (IAM) 管理者ユーザーを作成します。作成手順については、AWS Identity and Access Management IAM ユーザーガイドの最初の IAM ユーザーおよびグループの作成を参照してください。

リージョンとエンドポイント

エンドポイントは、ウェブサービスのエントリポイントとなるURLです。各エンドポイントは、特定の AWS リージョンに関連付けられます。Amazon Kendra コンソール、 AWS CLI、および Amazon Kendra SDKsの組み合わせを使用する場合は、特定のキャンペーンのすべての Amazon Kendra コンポーネント (インデックス、クエリなど) を同じリージョンで作成する必要があるため、デフォルトのリージョンに注意してください。Amazon Kendra でサポートされているリージョンやエンドポイントについては、リージョンとエンドポイントを参照してください。

のセットアップ AWS CLI

AWS コマンドラインインターフェイス (AWS CLI) は、Amazon Kendra を含む のサービスを管理する AWS ための統合デベロッパーツールです。このツールをインストールすることをお勧めします。

1. をインストールするには AWS CLI、AWS 「 コマンドラインインターフェイスユーザーガイド」のAWS 「 コマンドラインインターフェイスのインストール」の手順に従います。
2. を設定し、 を呼び出すようにプロファイル AWS CLI を設定するには AWS CLI、AWS 「 コマンドラインインターフェイスユーザーガイド」の「 の設定 AWS CLI」の手順に従います。
3. AWS CLI プロファイルが正しく設定されていることを確認するには、次のコマンドを実行します。

   ```
   aws configure --profile default
   ```

   プロファイルが正しく設定されている場合は、次のような出力が表示されます。

   ```
   AWS Access Key ID [****************52FQ]:
   AWS Secret Access Key [****************xgyZ]:
   Default region name [us-west-2]:
   Default output format [json]:
   ```

4. AWS CLI が Amazon Kendra で使用するように設定されていることを確認するには、次のコマンドを実行します。

   ```
   aws kendra help
   ```

   が正しく AWS CLI 設定されている場合、Amazon Kendra、Amazon Kendra ランタイム、および Amazon Kendra イベントでサポートされている AWS CLI コマンドのリストが表示されます。

AWS SDKsのセットアップ

使用する AWS SDKsをダウンロードしてインストールします。このガイドでは、Python の例を示しています。他の AWS SDKs「アマゾン ウェブ サービスのツール」を参照してください。

Python SDK 用のパッケージは Boto3 と呼ばれています。

以下の Python コマンドを実行する前に、まず、ご使用のオペレーティングシステム用の Python 3.6 以降をダウンロードして、インストールする必要があります。Python 3.5 以前のサポートは廃止されました。Python Scripts ディレクトリに pip が含まれていない場合は、get-pip.py をダウンロードして Scripts ディレクトリに保存できます。ターミナルプログラムを使用して Python ディレクトリを Path 変数または環境変数として設定することもできます。

```
# Install the latest Boto3 release via pip
pip install boto3

# You can install a specific version of Boto3 for compatibility reasons
# Install Boto3 version 1.0 specifically
pip install boto3==1.0.0

# Make sure Boto3 is no older than version 1.15.0
pip install boto3>=1.15.0

# Avoid versions of Boto3 newer than version 1.15.3
pip install boto3<=1.15.3
```

Boto3 を使用するには、IAM コンソール を使用して AWS アカウントの認証情報を設定する必要があります。

IAM の アクセスロール Amazon Kendra

インデックス、データソース、またはよくある質問を作成する場合、 には AWS リソースの作成に必要な Amazon Kendra リソースへのアクセス Amazon Kendra が必要です。 Amazon Kendra リソースを作成する前に、 AWS Identity and Access Management （ IAM) ポリシーを作成する必要があります。オペレーションを呼び出すときに、ポリシーをアタッチしたロールの Amazon リソースネーム (ARN) を指定します。例えば、 BatchPutDocument API を呼び出して Amazon S3 バケットからドキュメントを追加する場合は、バケットにアクセスできるポリシーを持つ Amazon Kendra ロールを に提供します。

Amazon Kendra コンソールで新しい IAM ロールを作成するか、使用する IAM 既存のロールを選択できます。コンソールには、ロール名に「kendra」か、「Kendra」という文字列を含むロールが表示されます。

次のトピックでは、必要なポリシーの詳細について説明します。 Amazon Kendra コンソールを使用して IAM ロールを作成すると、これらのポリシーが自動的に作成されます。

トピック

IAM インデックスの ロール

インデックスを作成するときは、 に書き込むアクセス許可を IAM ロールに付与する必要があります Amazon CloudWatch。また、 がロールを引き受け Amazon Kendra ることを許可する信頼ポリシーも指定する必要があります。次のポリシーを提供する必要があります。

IAM インデックスの ロール

が CloudWatch ログにアクセス Amazon Kendra することを許可するロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "cloudwatch:PutMetricData",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/Kendra"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "logs:DescribeLogGroups",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "arn:aws:logs:your-region:your-account-id:log-group:/aws/
kendra/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogStreams",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:your-region:your-account-id:log-group:/aws/
kendra/*:log-stream:*"
        }
    ]
}
```

による Amazon Kendra へのアクセスを許可するロールポリシー AWS Secrets Manager。をキーの場所 Secrets Manager としてユーザーコンテキストを使用している場合は、次のポリシーを使用できます。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"cloudwatch:PutMetricData",
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "cloudwatch:namespace":"AWS/Kendra"
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":"logs:DescribeLogGroups",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"logs:CreateLogGroup",
         "Resource":"arn:aws:logs:your-region:your-account-id:log-group:/aws/kendra/*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "logs:DescribeLogStreams",
            "logs:CreateLogStream",
            "logs:PutLogEvents"
         ],
         "Resource":"arn:aws:logs:your-region:your-account-id:log-group:/aws/kendra/
*:log-stream:*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "secretsmanager:GetSecretValue"
         ],
         "Resource":[
```

```
            "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "kms:Decrypt"
         ],
         "Resource":[
            "arn:aws:kms:your-region:your-account-id:key/key-id"
         ],
         "Condition":{
            "StringLike":{
               "kms:ViaService":[
                  "secretsmanager.your-region.amazonaws.com"
               ]
            }
         }
      }
   ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM API の BatchPutDocumentロール

> **Warning**
>
> Amazon Kendra は、 プリン Amazon Kendra シパルに S3 バケットとやり取りするためのアクセス許可を付与するバケットポリシーを使用しません。代わりに IAM ロールを使用します。任意のプリンシパルに誤ってアクセス許可を付与することによるデータセキュリティの問題を避けるため、 Amazon Kendra がバケットポリシーに信頼されたメンバーとして含まれていないことを確認してください。ただし、バケットポリシーを追加して、異なるアカウント間で Amazon S3 バケットを使用できます。詳細については、「複数のアカウント間で Amazon S3 を使用するポリシー」を参照してください。S3 データソースの IAM ロールについては、「IAM ロール」を参照してください。

BatchPutDocument API を使用して Amazon S3 バケット内のドキュメントのインデックスを作成する場合は、 Amazon Kendra IAM ロールに バケットへのアクセスを提供する必要があります。また、 がロールを引き受け Amazon Kendra ることを許可する信頼ポリシーも指定する必要があります。バケット内のドキュメントが暗号化されている場合は、 AWS KMS カスタマーマスターキー (CMK) を使用してドキュメントを復号するためのアクセス許可を提供する必要があります。

IAM API の BatchPutDocumentロール

Amazon Kendra バケットへのアクセス Amazon S3 を に許可するために必要なロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

信頼ポリシーにはaws:sourceAccount と aws:sourceArn を含めることをお勧めします。これにより、アクセス許可が制限され、 aws:sourceAccountおよび aws:sourceArnがsts:AssumeRoleアクションの IAM ロールポリシーで指定されているものと同じかどうかが安全に確認されます。これにより、権限のないエンティティが IAM ロールとそのアクセス許可にアクセスするのを防ぐことができます。詳細については、混乱した代理問題に関する AWS Identity and Access Management ガイドを参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "kendra.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your-account-id"
                },
                "StringLike": {
                    "aws:SourceArn": "arn:aws:kendra:your-region:your-account-id:index/
*"
                }
            }
```

```
        }
    ]
}
```

Amazon Kendra が カスタマーマスターキー (CMK) を使用して AWS KMS Amazon S3 バケット内のドキュメントを復号できるようにするオプションのロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

IAM データソースの ロール

CreateDataSource API を使用する場合は、リソースへのアクセス許可を持つ Amazon Kendra IAM ロールを付与する必要があります。必要な固有のアクセス許可は、データソースによって異なります。

IAM Adobe Experience Manager データソースの ロール

Adobe Experience Manager を使用する場合、以下のようなポリシーでロールを提供します。

- Adobe Experience Manager を認証するためのシー AWS Secrets Manager クレットへのアクセス許可。
- Adobe Experience Manager コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Adobe Experience Manager データソース Amazon Kendra は、 を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
```

```
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Alfresco データソースの ロール

Alfresco を使用する場合、以下のようなポリシーでロールを提供します。

- シー AWS Secrets Manager クレットにアクセスして Alfresco を認証するためのアクセス許可。
- Alfresco コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Alfresco データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAurora (MySQL) データソースの ロール

Aurora (MySQL) を使用する場合は、次のポリシーでロールを指定します。

- シー AWS Secrets Manager クレットにアクセスして を認証するアクセス許可 Aurora (MySQL)。
- Aurora (MySQL) コネクタに必要なパブリック APIs を呼び出すアクセス許可。 MySQL
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Aurora (MySQL) データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAurora (PostgreSQL) データソースの ロール

Aurora (PostgreSQL) を使用する場合は、次のポリシーでロールを指定します。

- シー AWS Secrets Manager クレットにアクセスして を認証するアクセス許可 Aurora (PostgreSQL)。
- Aurora (PostgreSQL) コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Aurora (PostgreSQL) データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAmazon FSx データソースの ロール

を使用する場合は Amazon FSx、次のポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Amazon FSx ファイルシステムを認証するためのアクセス許可。
- ファイルシステムが存在する Amazon Virtual Private Cloud (VPC) Amazon FSx へのアクセス許可。
- Amazon FSx ファイルシステムの Active Directory のドメイン名を取得するためのアクセス許可。
- Amazon FSx コネクタに必要なパブリック API アクションの呼び出し許可。
- インデックスを更新する `BatchPutDocument` および `BatchDeleteDocument` API の呼び出し許可。

```
{
    "Version": "2012-10-17",
        "Statement": [
        {
          "Effect": "Allow",
          "Action": [
            "secretsmanager:GetSecretValue"
          ],
          "Resource": [
            "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:
{{secret-id}}"
          ]
        },
        {
          "Effect": "Allow",
          "Action": [
            "kms:Decrypt"
          ],
          "Resource": [
            "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/{{key-id}}"
          ],
          "Condition": {
            "StringLike": {
              "kms:ViaService": [
                "secretsmanager.{{your-region}}.amazonaws.com"
              ]
            }
          }
        },
        {
          "Effect": "Allow",
          "Action":[
            "ec2:CreateNetworkInterface",
            "ec2:DeleteNetworkInterface"
          ],
          "Resource": [
                "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/*",
                "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]"
          ]
        },
        {
          "Effect": "Allow",
          "Action": [
```

```
            "ec2:DescribeSubnets",
            "ec2:DescribeNetworkInterfaces"
          ],
          "Resource": "*"
        },
        {
          "Effect": "Allow",
          "Action": [
            "ec2:CreateNetworkInterfacePermission"
          ],
          "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-
interface/*",
          "Condition": {
            "StringEquals": {
              "ec2:AuthorizedService": "kendra.*.amazonaws.com"
            },
            "ArnEquals": {
              "ec2:Subnet": [
                "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]"
              ]
            }
          }
        },
        {
          "Sid": "AllowsKendraToGetDomainNameOfActiveDirectory",
          "Effect": "Allow",
          "Action": "ds:DescribeDirectories",
          "Resource": "*"
        },
        {
          "Sid": "AllowsKendraToCallRequiredFsxAPIs",
          "Effect": "Allow",
          "Action": [
              "fsx:DescribeFileSystems"
          ],
          "Resource": "*"
        },
        {
          "Sid": "iamPassRole",
          "Effect": "Allow",
          "Action": "iam:PassRole",
          "Resource": "*",
          "Condition": {
            "StringEquals": {
```

```
              "iam:PassedToService": [
                "kendra.*.amazonaws.com"
              ]
            }
          }
        },
        {
          "Effect": "Allow",
          "Action": [
            "kendra:BatchPutDocument",
            "kendra:BatchDeleteDocument"
          ],
          "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/
{{index-id}}"
        }
        ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM データベースデータソースの ロール

データベースをデータソースとして使用する場合は、 への接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- サイトのユーザー名とパスワードを含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「データソース」を参照してください。

- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 Secrets Manager。
- インデックスを更新するために BatchPutDocument および BatchDeleteDocument オペレーションを使用する許可。
- サイトとの通信に使用される SSL 証明書を含む Amazon S3 バケットへのアクセス許可。

> **Note**
>
> データベースデータソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kendra:BatchPutDocument",
                "kendra:BatchDeleteDocument"
            ],
            "Resource": [
```

```
                "arn:aws:kendra:your-region:your-account-id:index/index-id"
            "Condition": {
                "StringLike": {
                    "kms:ViaService": [
                        "kendra.your-region.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

データソースで使用できるポリシーには 2 つのオプションがあります。

との通信に使用される SSL 証明書を含む Amazon S3 バケットを暗号化している場合は、キー Amazon Kendra へのアクセスを許可するポリシーを指定します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

VPC を使用している場合は、必要なリソース Amazon Kendra へのアクセスを許可するポリシーを指定します。必要なポリシーについては「データソースおよび VPC の IAM ロール」を参照してください。

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAmazon RDS (Microsoft SQL Server) データソースの ロール

Amazon RDS (Microsoft SQL Server) データソースコネクタを使用する場合は、次のポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Amazon RDS (Microsoft SQL Server) データソースインスタンスを認証するためのアクセス許可。
- Amazon RDS (Microsoft SQL Server) データソースコネクタに必要なパブリック APIs を呼び出すアクセス許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon RDS (Microsoft SQL Server) データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
"arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
```

```
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM (MySQL) データソースの Amazon RDS ロール MySQL

Amazon RDS (MySQL) データソースコネクタを使用する場合は、次のポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Amazon RDS (MySQL) データソースインスタンスを認証するためのアクセス許可。
- Amazon RDS (MySQL) データソースコネクタに必要なパブリック APIs を呼び出すアクセス許可。 MySQL
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon RDS (MySQL) データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
```

```
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM (Oracle) データソースの Amazon RDS ロール

Amazon RDS Oracle データソースコネクタを使用する場合は、次のポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Amazon RDS (Oracle) データソースインスタンスを認証するためのアクセス許可。
- Amazon RDS (Oracle) データソースコネクタに必要なパブリック APIs を呼び出すアクセス許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon RDS Oracle データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
"arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
```

```
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAmazon RDS (PostgreSQL) データソースの ロール

Amazon RDS (PostgreSQL) データソースコネクタを使用する場合は、次のポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Amazon RDS (PostgreSQL) データソースインスタンスを認証するためのアクセス許可。
- Amazon RDS (PostgreSQL) データソースコネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon RDS (PostgreSQL) データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
"arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
```

```
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAmazon S3 データソースの ロール

> **Warning**
>
> Amazon Kendra は、 プリン Amazon Kendra シパルに S3 バケットとやり取りするためのアクセス許可を付与するバケットポリシーを使用しません。代わりに、 IAM ロールを使用します。任意のプリンシパルに誤ってアクセス許可を付与することによるデータセキュリティの問題を避けるため、 Amazon Kendra がバケットポリシーに信頼されたメンバーとして含まれていないことを確認してください。ただしバケットポリシーを追加すれば、異なるアカウント間で Amazon S3 バケットを使用できます。詳細については、「複数のアカウントで Amazon S3 を使用するためのポリシー (下にスクロール)」を参照してください。

Amazon S3 バケットをデータソースとして使用する場合は、バケットへのアクセス許可を持つロールを指定し、 `BatchPutDocument` および `BatchDeleteDocument`オペレーションを使用します。バケット内の Amazon S3 ドキュメントが暗号化されている場合は、 AWS KMS カスタマーマスターキー (CMK) を使用してドキュメントを復号するためのアクセス許可を提供する必要があります。

次のロールポリシーでは、 Amazon Kendra がロールを引き受けることを許可する必要があります。下にスクロールすると、ロールを引き受けるための信頼ポリシーが表示されます。

が Amazon S3 バケット Amazon Kendra をデータソースとして使用できるようにする必須ロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
         {
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name"
            ],
            "Effect": "Allow"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kendra:BatchPutDocument",
                "kendra:BatchDeleteDocument"
            ],
            "Resource": [
                "arn:aws:kendra:your-region:your-account-id:index/index-id"
            ]
        }
    ]
}
```

Amazon Kendra が カスタマーマスターキー (CMK) を使用して AWS KMS Amazon S3 バケット内のドキュメントを復号できるようにするオプションのロールポリシー。

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

```
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

アクセス AWS KMS 許可 Amazon Kendra をアクティブ化 AWS KMS または共有 Amazon VPCせずに、 の使用中に が バケットにアクセス Amazon S3 できるようにするオプションのロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::{{bucket-name}}/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::{{bucket-name}}"
      ],
      "Effect": "Allow"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
```

```
        "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]",
        "arn:aws:ec2:{{your-region}}:{{your-account-id}}:security-group/[[security-
group]]"
      ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:CreateNetworkInterface"
        ],
        "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/
*",
        "Condition": {
            "StringLike": {
                "aws:RequestTag/AWS_KENDRA": "kendra_{{your-account-id}}_{{index-
id}}_{data-source-id}}_*"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:CreateTags"
        ],
        "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/
*",
        "Condition": {
            "StringEquals": {
                "ec2:CreateAction": "CreateNetworkInterface"
            }
        }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeNetworkInterfaces"
      ],
```

```
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission"
      ],
      "Resource": "arn:aws:ec2:{{your-region}}:{{your-accoount-id}}:network-interface/
*",
      "Condition": {
        "StringEquals": {
          "ec2:AuthorizedService": "kendra.amazonaws.com"
        },
        "ArnEquals": {
          "ec2:Subnet": [
            "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
      ],
      "Resource": [
        "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}",
        "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:BatchPutDocument",
        "kendra:BatchDeleteDocument"
      ],
      "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}"
    }
  ]
```

```
}
```

Amazon Kendra 、、および アクセス AWS KMS 許可が有効になっている の使用中に Amazon VPC が バケットにアクセス Amazon S3 できるようにするオプションのロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::{{bucket-name}}/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::{{bucket-name}}"
      ],
      "Effect": "Allow"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/{{key-id}}"
      ],
      "Condition": {
        "StringLike": {
          "kms:ViaService": [
            "s3.{{your-region}}.amazonaws.com"
          ]
        }
      }
    },
    {
```

```
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]",
        "arn:aws:ec2:{{your-region}}:{{your-account-id}}:security-group/[[security-
group]]"
      ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:CreateNetworkInterface"
        ],
        "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/
*",
        "Condition": {
            "StringLike": {
                "aws:RequestTag/AWS_KENDRA": "kendra_{{your-account-id}}_{{index-
id}}_{data-source-id}}_*"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:CreateTags"
        ],
        "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/
*",
        "Condition": {
            "StringEquals": {
                "ec2:CreateAction": "CreateNetworkInterface"
            }
        }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    },
```

```
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeNetworkInterfaces"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission"
      ],
      "Resource": "arn:aws:ec2:{{your-region}}:{{your-account-id}}:network-interface/
*",
      "Condition": {
        "StringEquals": {
          "ec2:AuthorizedService": "kendra.amazonaws.com"
        },
        "ArnEquals": {
          "ec2:Subnet": [
            "arn:aws:ec2:{{your-region}}:{{your-account-id}}:subnet/[[subnet-ids]]"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
      ],
      "Resource": [
        "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}",
        "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:BatchPutDocument",
        "kendra:BatchDeleteDocument"
```

```
      ],
      "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}"
    }
  ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

複数のアカウントで Amazon S3 を使用するためのポリシー

Amazon S3 バケットが Amazon Kendra インデックスに使用するアカウントとは別のアカウントにある場合は、アカウント間でバケットを使用するポリシーを作成できます。

Amazon S3 バケットが Amazon Kendra インデックスとは異なるアカウントにある場合に、バケットをデータソースとして使用するロールポリシー。なお `s3:PutObject` および `s3:PutObjectAcl` はオプションであり、アクセス制御リストに設定ファイルを含めたい場合に使用してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::$bucket-in-other-account/*"
            ],
```

```
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::$bucket-in-other-account/*"
            ],
            "Effect": "Allow"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kendra:BatchPutDocument",
                "kendra:BatchDeleteDocument"
            ],
            "Resource": [
                "arn:aws:kendra:$your-region:$your-account-id:index/$index-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::$bucket-in-other-account/*"
        }
    ]
}
```

Amazon S3 データソースロールがアカウント間でバケットにアクセスすることを許可する Amazon S3 バケットポリシー。なお s3:PutObject および s3:PutObjectAcl はオプションであり、アクセス制御リストに設定ファイルを含めたい場合に使用してください。

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

```
                "AWS": "$kendra-s3-connector-role-arn"
            },
            "Action": [
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
                "arn:aws:s3:::$bucket-in-other-account/*"
            ]
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "$kendra-s3-connector-role-arn"
            },
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::$bucket-in-other-account"
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM ウェブクローラーデータソースの Amazon Kendra ロール

Web Crawler Amazon Kendra を使用する場合は、次のポリシーでロールを指定します。

- 基本認証によってバックアップされたウェブサイトまたはウェブプロキシサーバーに接続するための認証情報を含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容に関する詳細は、「Web クローラーデータソースの使用」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。
- Amazon S3 バケットを使用してシード URLs またはサイトマップのリストを保存する場合は、Amazon S3 バケットへのアクセス許可を含めます。

> **Note**
>
> Amazon Kendra ウェブクローラーデータソース Amazon Kendra は、 を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account:key/key-id"
    ],
    "Condition": {
      "StringLike": {
```

```
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

シード URLs またはサイトマップを Amazon S3 バケットに保存する場合は、このアクセス許可をロールに追加する必要があります。

```
,
{"Effect": "Allow",
     "Action": [
         "s3:GetObject"
      ],
      "Resource": [
         "arn:aws:s3:::bucket-name/*"
      ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
}
```

IAMAmazon WorkDocs データソースの ロール

を使用する場合 Amazon WorkDocs、次のポリシーでロールを指定します。

- Amazon WorkDocs サイトリポジトリに対応するディレクトリ ID (組織 ID) の検証許可。
- Amazon WorkDocs サイトディレクトリを含むアクティブディレクトリのドメイン名の取得許可。
- Amazon WorkDocs コネクタに必要なパブリック API アクションの呼び出し許可。
- インデックスを更新する `BatchPutDocument` および `BatchDeleteDocument` API の呼び出し許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowsKendraToGetDomainNameOfActiveDirectory",
      "Effect": "Allow",
      "Action": "ds:DescribeDirectories",
      "Resource": "*"
    },
    {
      "Sid": "AllowsKendraToCallRequiredWorkDocsAPIs",
      "Effect": "Allow",
      "Action": [
        "workdocs:GetDocumentPath",
        "workdocs:GetGroup",
        "workdocs:GetDocument",
        "workdocs:DownloadDocumentVersions",
        "workdocs:DescribeUsers",
        "workdocs:DescribeFolderContents",
        "workdocs:DescribeActivities",
        "workdocs:DescribeComments",
        "workdocs:GetFolder",
        "workdocs:DescribeResourcePermissions",
        "workdocs:GetFolderPath",
        "workdocs:DescribeInstances"
      ],
      "Resource": "*"
    },
    {
```

```
      "Sid": "iamPassRole",
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "kendra.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "AllowsKendraToCallBatchPutDeleteAPIs",
      "Effect": "Allow",
      "Action": [
        "kendra:BatchPutDocument",
        "kendra:BatchDeleteDocument"
      ],
      "Resource": [
        "arn:aws:kendra:your-region:account-id:index/$index-id"
      ]
    }
  ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Box データソースの ロール

Box を使用する場合、以下のようなポリシーでロールを提供します。

- シー AWS Secrets Manager クレットにアクセスして Slack を認証するためのアクセス許可。
- Box コネクタに必要なパブリック API アクションの呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Box データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
```

```
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-d}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Confluence データソースの ロール

IAM Confluence Connector v1.0 の ロール

Confluence サーバーをデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- Confluence への接続に必要な認証情報を含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「Confluence データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。

> **Note**
>
> Amazon Kendra を介して Confluence データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

VPC を使用している場合は、必要なリソース Amazon Kendra へのアクセスを許可するポリシーを指定します。必要なポリシーについては「データソースおよび VPC の IAM ロール」を参照してください。

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Confluence Connector v2.0 の ロール

Confluence コネクタ v2.0 データソース用には、以下のようなポリシーを提供します。

- Confluence の認証情報を含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「Confluence データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 AWS Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。

また、 がロールを引き受けることを許可する信頼ポリシー Amazon Kendra をアタッチする必要があります。

> **Note**
>
> Amazon Kendra を介して Confluence データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

が Confluence に接続 Amazon Kendra できるようにするロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": [
        "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:your-region:your-account-id:key/key-id"
      ],
      "Condition": {
        "StringLike": {
```

```
          "kms:ViaService": [
            "secretsmanager.your-region.amazonaws.com"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
      ],
      "Resource": [
        "arn:aws:kendra:your-region:your-account-id:index/index-id",
        "arn:aws:kendra:your-region:your-account-id:index/index-id/data-source/*"
      ]
    }
    {
      "Effect": "Allow",
      "Action": [
        "kendra:BatchPutDocument",
        "kendra:BatchDeleteDocument"
      ],
      "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
    }
  ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
}
```

IAM Dropbox データソースの ロール

Dropbox を使用する場合、以下のようなポリシーでロールを提供します。

- AWS Secrets Manager シークレットにアクセスして Dropbox を認証するためのアクセス許可。
- Dropbox コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMapp`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Dropbox データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
"Version": "2012-10-17",
  "Statement": [
  {"Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {"Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {"StringLike": {"kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
```

```
  },
  {"Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {"Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Drupal データソースの ロール

Drupal を使用する場合、以下のようなポリシーでロールを提供します。

- AWS Secrets Manager シークレットにアクセスして Drupal を認証するためのアクセス許可。
- Drupal コネクタに必要なパブリック API の呼び出し許可。

- BatchPutDocument、BatchDeleteDocument、PutPrincipalMapping、DeletePrincipalMappi
  および ListGroupsOlderThanOrderingId API を呼び出す許可。

> **Note**
>
> Drupal データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
```

```
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM GitHub データソースの ロール

を使用する場合は GitHub、次のポリシーでロールを指定します。

- シー AWS Secrets Manager クレットにアクセスして を認証するためのアクセス許可 GitHub。
- GitHub コネクタに必要なパブリック APIs を呼び出すアクセス許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> GitHub データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Gmail データソース用の ロール

Gmail を使用する場合、以下のようなポリシーでロールを提供します。

- AWS Secrets Manager シークレットにアクセスして Gmail を認証するためのアクセス許可。
- Gmail コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Gmail データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
"Version": "2012-10-17",
  "Statement": [
  {"Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {"Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {"StringLike": {"kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {"Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {"Effect": "Allow",
```

```
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Google Drive データソースの ロール

Google Workspace Drive データソースを使用する場合は、サイトへの接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- Google Drive サイトへの接続に必要なクライアントアカウントの E メール、管理者アカウントの E メール、およびプライベートキーを含むシー AWS Secrets Manager クレットを取得および復号するためのアクセス許可。シークレットの内容の詳細については、「Google Drive データソース」を参照してください。
- BatchPutDocument および BatchDeleteDocument APIsを使用するアクセス許可。

> **Note**
>
> Google Drive データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

次の IAM ポリシーは、必要なアクセス許可を提供します。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

```
{
```

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

IAM IBM DB2 データソースの ロール

Confluence サーバーをデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- シー AWS Secrets Manager クレットにアクセスして IBM DB2 データソースインスタンスを認証するためのアクセス許可。
- IBM DB2 データソースコネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> IBM DB2 データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
```

```
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"kendra.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

IAM Jira データソースの ロール

Jira を使用する場合、以下のようなポリシーでロールを提供します。

- AWS Secrets Manager シークレットにアクセスして Jira を認証するためのアクセス許可。
- Jira コネクタに必要なパブリック API アクションの呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Jira データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

が ロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
         "Principal":{
            "Service":"kendra.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

IAM Microsoft Exchange データソース用の ロール

Microsoft Exchange データソースを使用する場合は、サイトへの接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- Microsoft Exchange サイトへの接続に必要なアプリケーション ID とシー AWS Secrets Manager クレットキーを含むシークレットを取得および復号するためのアクセス許可。シークレットの内容に関する詳細は、「Microsoft Exchange データソース」を参照してください。
- BatchPutDocument および BatchDeleteDocument APIsを使用するアクセス許可。

> **Note**
>
> Microsoft Exchange データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

次の IAM ポリシーは、必要なアクセス許可を提供します。

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

```
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

インデックスを作成するユーザーのリストを Amazon S3 バケットに保存する場合は、S3 GetObjectオペレーションを使用するアクセス許可も提供する必要があります。次の IAM ポリシーで、必要な許可が提供されます。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Action": [
      "s3:GetObject"
```

```
    ],
    "Resource": [
      "arn:aws:s3:::bucket-name/*"
    ],
    "Effect": "Allow"
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/[[key-ids]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com",
          "s3.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"kendra.amazonaws.com"
         },
```

```
            "Action":"sts:AssumeRole"
        }
    ]
}
```

IAM Microsoft OneDrive データソース用の ロール

Microsoft OneDrive データソースを使用する場合は、サイトへの接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- OneDrive サイトへの接続に必要なアプリケーション ID とシー AWS Secrets Manager クレットキーを含むシークレットを取得および復号するためのアクセス許可。シークレットの内容の詳細については、「Microsoft OneDrive データソース」を参照してください。
- BatchPutDocument および BatchDeleteDocument APIsを使用するアクセス許可。

> **Note**
>
> Microsoft OneDrive データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

次の IAM ポリシーは、必要なアクセス許可を提供します。

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

```
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

インデックスを作成するユーザーのリストを Amazon S3 バケットに保存する場合は、S3 GetObjectオペレーションを使用するアクセス許可も提供する必要があります。次の IAM ポリシーで、必要な許可が提供されます。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Action": [
      "s3:GetObject"
    ],
    "Resource": [
      "arn:aws:s3:::bucket-name/*"
```

```
    ],
    "Effect": "Allow"
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/[[key-ids]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com",
          "s3.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
}
```

IAM Microsoft SharePoint データソース用の ロール

IAM Connector v1.0 の SharePoint ロール

Microsoft SharePoint コネクタ v1.0 データソースの場合は、次のポリシーでロールを指定します。

- SharePoint サイトのユーザー名とパスワードを含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「Microsoft SharePoint データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 AWS Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。
- SharePoint サイトとの通信に使用される SSL 証明書を含む Amazon S3 バケットへのアクセス許可。

また、 がロールを引き受けることを許可する信頼ポリシー Amazon Kendra をアタッチする必要があります。

> **Note**
>
> Amazon Kendra を介して Microsoft SharePoint データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kendra:BatchPutDocument",
                "kendra:BatchDeleteDocument"
            ],
            "Resource": [
                "arn:aws:kendra:your-region:your-account-id:index/index-id"
            ],
            "Condition": {
                "StringLike": {
                    "kms:ViaService": [
                        "kendra.your-region.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

SharePoint サイトとの通信に使用される SSL 証明書を含む Amazon S3 バケットを暗号化している場合は、キー Amazon Kendra へのアクセスを許可するポリシーを指定します。

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

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Connector v2.0 の SharePoint ロール

Microsoft SharePoint コネクタ v2.0 データソースの場合は、次のポリシーでロールを指定します。

- SharePoint サイトの認証情報を含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「Microsoft SharePoint データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 AWS Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。
- SharePoint サイトとの通信に使用される SSL 証明書を含む Amazon S3 バケットへのアクセス許可。

また、 がロールを引き受けることを許可する信頼ポリシー Amazon Kendra をアタッチする必要があります。

> **Note**
>
> Amazon Kendra を介して Microsoft SharePoint データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": [
        "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:your-region:your-account-id:key/key-id"
      ],
      "Condition": {
        "StringLike": {
          "kms:ViaService": [
            "secretsmanager.your-region.amazonaws.com"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
```

```
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
      ],
      "Resource": [
        "arn:aws:kendra:your-region:your-account-id:index/index-id",
        "arn:aws:kendra:your-region:your-account-id:index/index-id/data-source/*"
      ]
    },
    {
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::bucket-name/key-name"
      ],
      "Effect": "Allow"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kendra:BatchPutDocument",
        "kendra:BatchDeleteDocument"
      ],
      "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:your-region:your-account-id:subnet/subnet-ids",
        "arn:aws:ec2:your-region:your-account-id:security-group/security-group"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": "arn:aws:ec2:region:account_id:network-interface/*",
      "Condition": {
        "StringLike": {
          "aws:RequestTag/AWS_KENDRA": "kendra_your-account-id_index-id_*"
```

```
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:your-region:your-account-id:network-interface/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction": "CreateNetworkInterface"
        }
      }
    },

{
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission"
      ],
      "Resource": "arn:aws:ec2:your-region:your-account-id:network-interface/*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/AWS_KENDRA": "kendra_your-account-id_index-id_*"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeVpcs",
        "ec2:DescribeRegions",
        "ec2:DescribeNetworkInterfacePermissions",
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    }
  ]
}
```

SharePoint サイトとの通信に使用される SSL 証明書を含む Amazon S3 バケットを暗号化している場合は、キー Amazon Kendra へのアクセスを許可するポリシーを指定します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:youraccount-id:key/key-id"
            ]
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Microsoft SQL Server データソース用の ロール

Microsoft SQL Server をデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Microsoft SQL Server インスタンスを認証するためのアクセス許可。
- Microsoft SQL Server コネクタに必要なパブリック API の呼び出し許可。

- BatchPutDocument、BatchDeleteDocument、PutPrincipalMapping、DeletePrincipalMappi
  および ListGroupsOlderThanOrderingId API を呼び出す許可。

> **Note**
>
> Amazon Kendra を介して Microsoft SQL Server データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
```

```
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Microsoft Teams データソースの ロール

Microsoft Teams データソースを使用する場合は、サイトへの接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- Microsoft Teams への接続に必要なクライアント ID とクライアント AWS Secrets Manager シークレットを含むシークレットを取得および復号するためのアクセス許可。シークレットの内容に関する詳細は、「Microsoft Teams データソース」を参照してください。

> **Note**
>
> Microsoft Teams データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

次の IAM ポリシーは、必要なアクセス許可を提供します。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:client-id:secret:secret-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
```

```
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Microsoft Yammer データソース用の ロール

Microsoft Yammer データソースを使用する場合は、サイトへの接続に必要なアクセス許可を持つ Amazon Kendra ロールを に提供します。具体的には次のとおりです。

- Microsoft Yammer サイトへの接続に必要なアプリケーション ID とシー AWS Secrets Manager クレットキーを含むシークレットを取得および復号するためのアクセス許可。シークレットの内容に関する詳細は、「Microsoft Yammer データソース」を参照してください。
- BatchPutDocument および BatchDeleteDocument APIsを使用するアクセス許可。

> **Note**
>
> Amazon Kendra を介して Microsoft Yammer データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

次の IAM ポリシーは、必要なアクセス許可を提供します。

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

```
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

インデックスを作成するユーザーのリストを Amazon S3 バケットに保存する場合は、S3 GetObjectオペレーションを使用するアクセス許可も提供する必要があります。次の IAM ポリシーで、必要な許可が提供されます。

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

```
{
  "Effect": "Allow",
  "Action": [
    "secretsmanager:GetSecretValue"
  ],
  "Resource": [
    "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
  ]
},
{
  "Action": [
    "s3:GetObject"
  ],
  "Resource": [
    "arn:aws:s3:::bucket-name/*"
  ],
  "Effect": "Allow"
},
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt"
  ],
  "Resource": [
    "arn:aws:kms:your-region:your-account-id:key/[[key-ids]]"
  ],
  "Condition": {
    "StringLike": {
      "kms:ViaService": [
        "secretsmanager.your-region.amazonaws.com",
        "s3.your-region.amazonaws.com"
      ]
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "kendra:BatchPutDocument",
    "kendra:BatchDeleteDocument"
  ],
  "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
}]
```

```
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM MySQL データソースの ロール

My SQL をデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして My SQL データソースインスタンスを認証するためのアクセス許可。
- My SQL データソースコネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> MySQL データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Oracle データソース用の ロール

Oracle をデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして Oracle データソースインスタンスを認証するためのアクセス許可。
- Oracle データソースコネクタに必要なパブリック API アクションの呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Oracle データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
```

```
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

```
{
```

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

IAM PostgreSQL データソースの ロール

PostgreSQL をデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- AWS Secrets Manager シークレットにアクセスして PostgreSQL データソースインスタンスを認証するためのアクセス許可。
- PostgreSQL データソースコネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon Kendra を介して PostgreSQL データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{region}}:{{account_id}}:secret:[[secret_id]]"
    ]
  },
```

```
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{region}}:{{account_id}}:key/[[key_id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.*.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}",
 "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{region}}:{{account_id}}:index/{{index_id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
         "Effect":"Allow",
         "Principal":{
            "Service":"kendra.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

IAM Quip データソースの ロール

Quip を使用する場合、以下のようなポリシーでロールを提供します。

- シー AWS Secrets Manager クレットにアクセスして Quip を認証するためのアクセス許可。
- Quip コネクタに必要なパブリック API アクションの呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi` および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Quip データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{yoour-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{your-
index-id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{your-index-
id}}/data-source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"kendra.amazonaws.com"
```

```
        },
        "Action":"sts:AssumeRole"
      }
   ]
}
```

IAM Salesforce データソースの ロール

Slaesforce をデータソースとして使用する場合は、以下のようなポリシーでロールを指定します。

- Salesforce サイトのユーザー名とパスワードを含む AWS Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「Salesforce データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。

> **Note**
>
> Salesforce データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

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

```
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:account-id:index/index-id"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM ServiceNow データソースの ロール

をデータソース ServiceNow として使用する場合は、次のポリシーでロールを指定します。

- ServiceNow サイトのユーザー名とパスワードを含む Secrets Manager シークレットへのアクセス許可。シークレットの内容の詳細については、「ServiceNow データソース」を参照してください。
- AWS KMS カスタマーマスターキー (CMK) を使用して、 に保存されているユーザー名とパスワードシークレットを復号するアクセス許可 Secrets Manager。
- インデックスを更新するために `BatchPutDocument` および `BatchDeleteDocument` オペレーションを使用する許可。

> **Note**
>
> ServiceNow データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:your-region:your-account-id:secret:secret-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.your-region.amazonaws.com"
        ]
      }
```

```
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:your-region:your-account-id:index/index-id"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Slack データソースの ロール

Slack を使用する場合、次のポリシーでロールを提供します。

- シー AWS Secrets Manager クレットにアクセスして Slack を認証するためのアクセス許可。
- Slack コネクタに必要なパブリック API アクションの呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Amazon Kendra を介して Slack データソースを に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM Zendesk データソースの ロール

Zendesk を使用する場合、以下のようなポリシーでロールを提供します。

- Zendesk Suite を認証するためのシー AWS Secrets Manager クレットへのアクセス許可。
- Zendesk コネクタに必要なパブリック API の呼び出し許可。
- `BatchPutDocument`、`BatchDeleteDocument`、`PutPrincipalMapping`、`DeletePrincipalMappi`
  および `ListGroupsOlderThanOrderingId` API を呼び出す許可。

> **Note**
>
> Zendesk データソースは、 Amazon Kendra を介して に接続できます Amazon VPC。を使用している場合は Amazon VPC、アクセス許可 を追加する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": [
      "arn:aws:secretsmanager:{{your-region}}:{{your-account-id}}:secret:[[secret-id]]"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:{{your-region}}:{{your-account-id}}:key/[[key-id]]"
    ],
    "Condition": {
      "StringLike": {
        "kms:ViaService": [
          "secretsmanager.{{your-region}}.amazonaws.com"
        ]
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "kendra:PutPrincipalMapping",
        "kendra:DeletePrincipalMapping",
        "kendra:ListGroupsOlderThanOrderingId",
        "kendra:DescribePrincipalMapping"
    ],
```

```
    "Resource": ["arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-
id}}", "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}/data-
source/*"]
  },
  {
    "Effect": "Allow",
    "Action": [
      "kendra:BatchPutDocument",
      "kendra:BatchDeleteDocument"
    ],
    "Resource": "arn:aws:kendra:{{your-region}}:{{your-account-id}}:index/{{index-id}}"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

仮想プライベートクラウド (VPC) IAM ロール

Virtual Private Cloud (VPC) を使用してデータソースに接続する場合は、次の追加のアクセス許可を提供する必要があります。

VPC IAM ロール

```
{
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
```

```
      "Resource": [
        "arn:aws:ec2:{{region}}:{{account_id}}:subnet/[[subnet_ids]]",
        "arn:aws:ec2:{{region}}:{{account_id}}:security-group/[[security_group]]"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": "arn:aws:ec2:{{region}}:{{account_id}}:network-interface/*",
      "Condition": {
        "StringLike": {
          "aws:RequestTag/AWS_KENDRA": "kendra_{{account_id}}_{{index_id}}_*"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:{{region}}:{{account_id}}:network-interface/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction": "CreateNetworkInterface"
        }
      }
    },

{
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission"
      ],
      "Resource": "arn:aws:ec2:{{region}}:{{account_id}}:network-interface/*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/AWS_KENDRA": "kendra_{{account_id}}_{{index_id}}_*"
        }
      }
    },
    {
      "Effect": "Allow",
```

```
      "Action": [
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeVpcs",
        "ec2:DescribeRegions",
        "ec2:DescribeNetworkInterfacePermissions",
        "ec2:DescribeSubnets"
      ],
      "Resource": "*"
    }
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM よくある質問の ロール (FAQs)

CreateFaq API を使用して質問と回答をインデックスにロードする場合、ソースファイルを含む Amazon S3 バケットへのアクセス権を Amazon Kendra IAM ロールに提供する必要があります。ソースファイルが暗号化されている場合は、 AWS KMS カスタマーマスターキー (CMK) を使用してファイルを復号するためのアクセス許可を提供する必要があります。

IAM FAQsの ロール

Amazon Kendra バケットへのアクセス Amazon S3 を に許可するために必要なロールポリシー。

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

Amazon Kendra が カスタマーマスターキー (CMK) を使用して AWS KMS Amazon S3 バケット内のファイルを復号できるようにするオプションのロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ],
            "Condition": {
                "StringLike": {
                    "kms:ViaService": [
                        "kendra.your-region.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

```
{
```

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

IAM クエリ提案用の ロール

Amazon S3 ファイルをクエリ提案ブロックリストとして使用する場合は、 Amazon S3 ファイルと Amazon S3 バケットへのアクセス許可を持つロールを指定します。バケット内のブロックリストテキストファイル (Amazon S3 ファイル) Amazon S3 が暗号化されている場合は、 AWS KMS カスタマーマスターキー (CMK) を使用してドキュメントを復号するためのアクセス許可を提供する必要があります。

IAM クエリ提案用の ロール

Amazon Kendra が Amazon S3 ファイルをクエリ提案ブロックリストとして使用することを許可する必須ロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {"Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

Amazon Kendra が カスタマーマスターキー (CMK) を使用して AWS KMS Amazon S3 バケット内のドキュメントを復号できるようにするオプションのロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {"Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAM ユーザーとグループのプリンシパルマッピング用の ロール

PutPrincipalMapping API を使用してユーザーをグループにマッピングし、検索結果をユーザーコンテキストでフィルタリングする場合は、グループに属するユーザーまたはサブグループのリストを指定する必要があります。リストが 1 つのグループのユーザーまたはサブグループが 1000 を超える場合は、リストおよび Amazon S3 バケットの Amazon S3 ファイルへのアクセス許可を持つロールを指定する必要があります。 Amazon S3 バケット内のリストのテキストファイル (Amazon S3 ファイル) が暗号化されている場合は、 AWS KMS カスタマーマスターキー (CMK) を使用してドキュメントを復号するためのアクセス許可を提供する必要があります。

IAM プリンシパルマッピングの ロール

Amazon Kendra グループに属するユーザーおよびサブグループのリストとして が Amazon S3 ファイルを使用できるようにする必須ロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {"Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

Amazon Kendra が カスタマーマスターキー (CMK) を使用して AWS KMS Amazon S3 バケット内のドキュメントを復号できるようにするオプションのロールポリシー。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {"Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:your-region:your-account-id:key/key-id"
            ]
        }
    ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

```
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"kendra.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

信頼ポリシーにはaws:sourceAccount と aws:sourceArn を含めることをお勧めします。これにより、アクセス許可が制限され、 aws:sourceAccountおよび aws:sourceArnがsts:AssumeRoleアクションの IAM ロールポリシーで指定されているものと同じかどうかが安全に確認されます。これにより、権限のないエンティティが IAM ロールとそのアクセス許可にアクセスするのを防ぐことができます。詳細については、混乱した代理問題に関する AWS Identity and Access Management ガイドを参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "kendra.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your-account-id"
                },
                "StringLike": {
                    "aws:SourceArn": "arn:aws:kendra:your-region:your-account-id:index-
id/*"
                }
            }
        }
    ]
}
```

IAM の ロール AWS IAM Identity Center

UserGroupResolutionConfiguration オブジェクトを使用して AWS IAM Identity Center ID ソースからグループとユーザーのアクセスレベルを取得する場合は、 へのアクセス許可を持つロールを指定する必要があります IAM Identity Center。

IAM の ロール AWS IAM Identity Center

Amazon Kendra による へのアクセスを許可する必須のロールポリシー IAM Identity Center。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso-directory:SearchUsers",
                "sso-directory:ListGroupsForUser",
                "sso-directory:DescribeGroups",
                "sso:ListDirectoryAssociations"
            ],
            "Resource": [
                 "*"
            ]
        },
        {
          "Sid": "iamPassRole",
          "Effect": "Allow",
          "Action": "iam:PassRole",
          "Resource": "*",
          "Condition": {
            "StringEquals": {
              "iam:PassedToService": [
                "kendra.amazonaws.com"
              ]
            }
          }
        }
     ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

IAMAmazon Kendra エクスペリエンスのための ロール

CreateExperience または UpdateExperience APIs を使用して検索アプリケーションを作成または更新する場合は、必要なオペレーションと IAM Identity Center へのアクセス許可を持つロールを指定する必要があります。

IAMAmazon Kendra 検索エクスペリエンスのための ロール

ユーザーおよびグループの情報を保存する Query オペレーション、 QuerySuggestions オペレーション、 SubmitFeedbackオペレーション、および IAM Identity Center Amazon Kendra へのアクセスを に許可するために必要なロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowsKendraSearchAppToCallKendraApi",
      "Effect": "Allow",
      "Action": [
        "kendra:GetQuerySuggestions",
        "kendra:Query",
        "kendra:DescribeIndex",
        "kendra:ListFaqs",
        "kendra:DescribeDataSource",
        "kendra:ListDataSources",
        "kendra:DescribeFaq",
        "kendra:SubmitFeedback"
      ],
```

```
      "Resource": [
        "arn:aws:kendra:your-region:your-account-id:index/index-id"
      ]
    },
    {
      "Sid": "AllowKendraSearchAppToDescribeDataSourcesAndFaq",
      "Effect": "Allow",
      "Action": [
        "kendra:DescribeDataSource",
        "kendra:DescribeFaq"
      ],
      "Resource": [
        "arn:aws:kendra:your-region:your-account-id:index/index-id/data-source/data-
source-id",
        "arn:aws:kendra:your-region:your-account-id:index/index-id/faq/faq-id"
      ]
    },
    {
      "Sid": "AllowKendraSearchAppToCallSSODescribeUsersAndGroups",
      "Effect": "Allow",
      "Action": [
        "sso-directory:ListGroupsForUser",
        "sso-directory:SearchGroups",
        "sso-directory:SearchUsers",
        "sso-directory:DescribeUser",
        "sso-directory:DescribeGroup",
        "sso-directory:DescribeGroups",
        "sso-directory:DescribeUsers",
        "sso:ListDirectoryAssociations"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringLike": {
          "kms:ViaService": [
            "kendra.your-region.amazonaws.com"
          ]
        }
      }
    }
  ]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

信頼ポリシーにはaws:sourceAccount と aws:sourceArn を含めることをお勧めします。これにより、アクセス許可が制限され、 aws:sourceAccountおよび aws:sourceArnがsts:AssumeRoleアクションの IAM ロールポリシーで指定されているものと同じかどうかが安全に確認されます。これにより、権限のないエンティティが IAM ロールとそのアクセス許可にアクセスするのを防ぐことができます。詳細については、混乱した代理問題に関する AWS Identity and Access Management ガイドを参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "kendra.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your-account-id"
                },
                "StringLike": {
                    "aws:SourceArn": "arn:aws:kendra:your-region:your-account-id:index-
id/*"
                }
            }
```

```
        }
    ]
}
```

IAM Custom Document Enrichment の ロール

CustomDocumentEnrichmentConfiguration オブジェクトを使用してドキュメントのメタデータとコンテンツの高度な変更を適用する場合は、 PreExtractionHookConfigurationおよび/または を実行するために必要なアクセス許可を持つロールを指定する必要がありますPostExtractionHookConfiguration。PreExtractionHookConfiguration および/または PostExtractionHookConfiguration の Lambda 関数を設定して、取り込みプロセス中にドキュメントのメタデータとコンテンツの高度な変更を適用します。 Amazon S3 バケットのサーバー側の暗号化を有効にする場合は、 AWS KMS カスタマーマスターキー (CMK) を使用して Amazon S3 バケットに保存されているオブジェクトを暗号化および復号するためのアクセス許可を提供する必要があります。

IAM Custom Document Enrichment の ロール

がバケット Amazon Kendra の暗号化PostExtractionHookConfigurationを使用してPreExtractionHookConfigurationおよび Amazon S3 を実行できるようにする必須ロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "s3:GetObject",
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::bucket-name/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::bucket-name"
```

```
    ],
    "Effect": "Allow"
  },
  {
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt",
      "kms:GenerateDataKey"
    ],
    "Resource": [
      "arn:aws:kms:your-region:your-account-id:key/key-id"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "lambda:InvokeFunction"
    ],
    "Resource": "arn:aws:lambda:your-region:your-account-id:function:lambda-function"
  }]
}
```

Amazon S3 バケット Amazon Kendra の暗号化PostExtractionHookConfigurationを使用せずに PreExtractionHookConfigurationおよび を実行できるようにするオプションのロールポリシー。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "s3:GetObject",
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::bucket-name/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:ListBucket"
    ],
    "Resource": [
```

```
      "arn:aws:s3:::bucket-name"
    ],
    "Effect": "Allow"
  },
  {
    "Effect": "Allow",
    "Action": [
      "lambda:InvokeFunction"
    ],
    "Resource": "arn:aws:lambda:your-region:your-account-id:function:lambda-function"
  }]
}
```

がロールを引き受け Amazon Kendra ることを許可する信頼ポリシー。

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

信頼ポリシーにはaws:sourceAccount と aws:sourceArn を含めることをお勧めします。これにより、アクセス許可が制限され、 aws:sourceAccountおよび aws:sourceArnが sts:AssumeRoleアクションの IAM ロールポリシーで指定されているものと同じかどうかが安全に確認されます。これにより、権限のないエンティティが IAM ロールとそのアクセス許可にアクセスするのを防ぐことができます。詳細については、混乱した代理問題に関する AWS Identity and Access Management ガイドを参照してください。

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

```
                "Service": [
                    "kendra.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your-account-id"
                },
                "StringLike": {
                    "aws:SourceArn": "arn:aws:kendra:your-region:your-account-id:index-
id/*"
                }
            }
        }
    ]
}
```

Amazon Kendra のデプロイ

Amazon Kendra 検索をウェブサイトにデプロイするときには、React で使用できるソースコードを指定して、アプリケーションを始めましょう。ソースコードは、改定版 MIT ライセンスの下で無償で提供されます。それをそのまま使うことも、自分のニーズに合わせて変更することもできます。提供されている React アプリは、作業の開始に役立つサンプルです。本番環境に対応したアプリではありません。

コードなしで検索アプリケーションをデプロイし、アクセス制御付きの検索ページにエンドポイント URL を生成する方法については、「Amazon Kendra Experience Builder」を参照してください。

以下のコード例は、既存の React ウェブアプリケーションに Amazon Kendra 検索を追加します。

- https://kendrasamples.s3.amazonaws.com/kendrasamples-react-app.zip - デベロッパーが既存の React ウェブアプリケーションに機能検索エクスペリエンスを組み込むために使用できるサンプルファイル。

サンプルは、Amazon Kendra コンソールの検索ページに基づいてモデル化されています。これらには、検索結果を検索および表示するための同じ機能があります。例全体を使用することも、独自の使用のために機能を 1 つだけ選択することもできます。

Amazon Kendra コンソールで検索ページの 3 つのコンポーネントを表示するには、コードアイコン (</>) を右メニューから選択します。各セクションにポインタを置くと、コンポーネントの簡単な説明が表示され、コンポーネントのソースの URL が表示されます。

トピック

概要

既存の React アプリケーションにサンプルコードを追加して、検索を有効にします。サンプルコードには、新しい React 開発環境を設定する手順が記載された Readme ファイルが含まれています。サンプルコード内のサンプルデータは、検索のデモンストレーションに使用できます。サンプル内の検索ファイルとコンポーネントは、次のように構成されています。

- メイン検索ページ (`Search.tsx`) - これは、すべてのコンポーネントを含むメインページです。ここでは、アプリケーションを Amazon Kendra API と統合します。
- 検索バー - これはユーザーが検索語を入力し、検索機能を呼び出すコンポーネントです。
- 結果 - これは Amazon Kendra の結果を表示するコンポーネントです。提案された回答、よくある質問の結果、推奨ドキュメントの 3 つのコンポーネントがあります。
- ファセット - これは検索結果にファセットを表示するコンポーネントで、ファセットを選択して検索を狭めることができます。
- ページ割り - これは Amazon Kendra からの応答をページ分割するコンポーネントです。

前提条件

開始するには、以下が必要です。

- Node.js と npm がインストール済み。Node.js バージョン 19 以前が必要です。
- Python 3 または Python 2 がダウンロードおよびインストール済み。
- Amazon Kendra への API コールを実行する SDK for Java または AWS SDK for JavaScript。
- 既存の React ウェブアプリケーション。サンプルコードには、必須フレームワーク/ライブラリの使用を含む、新しい React 開発環境を設定する手順が記載された Readme ファイルが含まれています。React ウェブアプリの作成に関する React ドキュメントのクイックスタート手順に従うこともできます。
- 開発環境で設定されている必要なライブラリと依存関係。サンプルコードには、必要なライブラリとパッケージの依存関係について記載された Readme ファイルが含まれています。`sass` は必須であり、`node-sass` は廃止済みであることに注意してください。以前に `node-sass` をインストールしていた場合は、これをアンインストールしてから `sass` をインストールしてください。

例をセットアップする

Amazon Kendra 検索を React アプリケーションに追加するための完全な手順は、コードサンプルに含まれている Readme にあります。

kendrasamples-react-app.zip の使用開始方法

1. Node.js と npm のダウンロードとインストールを含め、前提条件 が完了していることを確認してください。
2. kendrasamples-react-app.zip をダウンロードして解凍します。
3. ターミナルを開いて、`aws-kendra-example-react-app/src/services/` に進みます。`local-dev-credentials.json` を開いて認証情報を提供します。このファイルをパブリックリポジトリに追加しないでください。
4. `aws-kendra-example-react-app` に進み、依存関係を `package.json` にインストールしてください。`npm install` を実行します。
5. ローカルサーバーでアプリのデモ版を起動します。`npm start` を実行します。キーボードで `Cmd/Ctrl + C` を入力すると、ローカルサーバーを停止できます。
6. ポートまたはホスト (IP アドレスなど) を変更するには、`package.json` に移動してホストとポートを `"start": "HOST=[host] PORT=[port] react-scripts start"` に更新します。Windows を使用している場合: `"start": "set HOST=[host] && set PORT=[port] && react-scripts start"`。
7. ウェブサイトのドメインを登録済みの場合は、アプリ名の後の `package.json` でこれを指定できます。例えば、`"homepage": "https://mywebsite.com"`。新しい依存関係を更新するためには `npm install` をもう一度実行してから、`npm start` を実行する必要があります。
8. アプリを構築するには、`npm build` を実行します。ビルドディレクトリの内容をホスティングプロバイダーにアップロードします。

> **Warning**
>
> React アプリは本番環境に対応していません。これは Amazon Kendra 検索のためにアプリをデプロイする例です。

メイン検索ページ

メイン検索ページ (Search.tsx) には、すべてのサンプル検索コンポーネントが含まれています。これには、出力用の検索バーコンポーネント、Query API からのレスポンスを表示する結果コンポーネント、およびレスポンスをページングするためのページ割りコンポーネントが含まれています。

検索コンポーネント

検索コンポーネントには、クエリテキストを入力するためのテキストボックスがあります。onSearch 機能は、メイン機能を Search.tsx で呼び出すフックで、Amazon Kendra Query API コールを行います。

結果コンポーネント

結果コンポーネントには、Query API からのレスポンスが表示されます。結果は 3 つの別個のエリアに表示されます。

- 提案された回答 - これらは、Query API により返された上位結果です。提案された回答は最大 3 つまで含まれます。レスポンスには、結果タイプ ANSWER があります。
- よくある質問の回答 - これらはレスポンスが返すよくある質問の結果です。よくある質問はインデックスに別々に追加されます。レスポンスには、タイプ QUESTION_ANSWER があります。詳細については、質問と回答を参照してください。
- 推奨ドキュメント - Amazon Kendra がレスポンスで返す追加ドキュメントです。Query API からのレスポンスには、DOCUMENT タイプがあります。

結果コンポーネントは、強調表示、タイトル、リンクなどの機能のコンポーネントを共有します。結果コンポーネントが機能するには、共有コンポーネントが存在する必要があります。

ファセットコンポーネント

ファセットコンポーネントには、検索結果で使用可能なファセットが一覧表示されます。各ファセットは、製作者などの特定のディメンションに沿ってレスポンスを分類します。リストからファセットを選択して、検索を特定のファセットに絞り込むことができます。

ファセットを選択すると、コンポーネントによって、ファセットに一致するドキュメントの検索を制限する属性フィルターを使用して Query が呼び出されます。

ページ割りコンポーネント

ページ割りコンポーネントを使用すると、複数のページでの Query API からの検索結果の表示が可能になります。Query API を PageSize および PageNumber パラメータで呼び出し、結果の特定のページを取得します。

コードなしで検索エクスペリエンスを構築する

フロントエンドコードを必要とせずに Amazon Kendra 検索アプリケーションを構築してデプロイできます。Amazon KendraExperience Builder は、数回のクリックで完全に機能する検索アプリケーションの構築とデプロイを支援し、すぐに検索を開始できます。検索ページをカスタムデザインし、ユーザーのニーズに合わせてエクスペリエンスを調整できます。Amazon Kendra は、検索ページの一意で完全にホストされたエンドポイント URL を生成し、ドキュメントやよくある質問の検索を開始します。検索エクスペリエンスの概念実証をすばやく構築し、他のユーザーと共有できます。

ビルダーで使用可能な検索エクスペリエンステンプレートを使用して、検索をカスタマイズします。他のユーザーを招待して検索エクスペリエンスを構築したり、チューニング目的で検索結果を評価したりできます。ユーザーが検索を開始する準備ができたら、セキュリティで保護されたエンドポイント URL を共有するだけです。

検索 Experience Builder の仕組み

検索エクスペリエンスを構築するための全体的なプロセスを次に示します。

1. 検索エクスペリエンスを作成するには、名前、説明を指定し、検索エクスペリエンスに使用するデータソースを選択します。
2. AWS IAM Identity Center でユーザーとグループのリストを設定し、検索エクスペリエンスへのアクセス権を割り当てます。自分をエクスペリエンスのオーナーとして含めます。詳細については、「the section called “検索ページへのアクセスを提供する”」を参照してください。
3. Amazon Kendra Experience Builder を開き、検索ページをデザインおよび調整します。独自の編集アクセス権または表示検索アクセス権を割り当てた他のユーザーと、検索エクスペリエンスのエンドポイント URL を共有できます。

CreateExperience API を呼び出して、検索エクスペリエンスを作成および設定します。コンソールを使用する場合は、インデックスを選択し、ナビゲーションメニューで [エクスペリエンス] を選択してエクスペリエンスを設定します。

検索エクスペリエンスを設計してチューニングする

検索エクスペリエンスを作成して設定したら、エンドポイント URL を使用して検索エクスペリエンスを開き、編集アクセス権を持つ所有者としての検索のカスタマイズを開始します。検索ボックスにクエリを入力し、サイドパネルの編集オプションを使用して検索をカスタマイズし、ページに適用する方法を確認します。公開する準備ができたら、[Publish] (公開) をクリックします。[Switch to live view] (ライブビューに切り替える) をクリックして、検索ページの最新公開バージョンを表示し、[Switch to build mode] (構築モードに切り替える) をクリックして、検索ページを編集またはカスタマイズすることもできます。

以下は、検索エクスペリエンスをカスタマイズする方法です。

フィルター

ファセット検索を追加するか、ドキュメント属性でフィルタリングします。これには、カスタム属性が含まれます。独自の設定済みのメタデータフィールドを使用して、フィルターを追加できます。例えば、各都市のカテゴリでファセット検索を行うには、すべての都市カテゴリを含む `_category` カスタムドキュメント属性を使用します。

提案する回答

機械学習で生成された回答をユーザーのクエリに追加します。例:「このコースはどれほど難しいですか?」 Amazon Kendra はコースの難しさを指しているすべてのドキュメントで最も関連性の高いテキストを取得し、最も関連性の高い回答を提案できます。

よくある質問

よくある質問への回答を得るには、よくある質問ドキュメントを追加します。例:「このコースを完了するのに何時間かかりますか?」 Amazon Kendra はこの質問に対する回答を含むよくある質問ドキュメントを使用して、正しい答えを出すことができます。

Sort

検索結果のソートを追加して、ユーザーが関連性、作成時刻、最終更新時刻、その他のソート基準で結果を整理できるようにします。

ドキュメント

検索ページでのドキュメントまたは検索結果の表示方法を設定します。ページに表示する結果の数を構成したり、ページ番号などのページ割りを追加したり、ユーザーフィードバックボタンを有効にしたり、検索結果にドキュメントメタデータフィールドを表示する方法を調整したりできます。

言語

言語を選択して、選択した言語で検索結果またはドキュメントをフィルタリングします。

検索ボックス

検索ボックスのサイズとプレースホルダテキストを設定し、クエリ提案を有効にします。

関連性チューニング

ドキュメントメタデータフィールドにブーストを追加して、ユーザーがドキュメントを検索するときにこれらのフィールドに重みを付けます。1 から始まり、10 に徐々に増加する重みを追加できます。テキスト、日付、および数値フィールドの種類をブーストできます。例えば、`_last_updated_at` および `_created_at` の他のフィールドよりも重い、または重要な重みまたは重要度を付けるには、これらのフィールドの重要度に応じて 1 ~ 10 の重みを指定します。検索アプリケーションまたはエクスペリエンスごとに異なる関連性チューニング設定を適用できます。

検索ページへのアクセスを提供する

検索エクスペリエンスには IAM Identity Center からアクセスします。検索エクスペリエンスを設定するとき、Identity Center ディレクトリに一覧表示されている他のユーザーが Amazon Kendra 検索ページへアクセスすることを許可します。それらのユーザーは、IAM Identity Center で自身の資格情報を使用してサインインし検索ページにアクセスするように導く、E メールを受け取ります。IAM Identity Center は組織レベルまたは AWS Organizations のアカウント所有者レベルでセットアップする必要があります。IAM Identity Center の詳細については、「Getting started with IAM Identity Center」を参照してください。

検索エクスペリエンスの IAM Identity Center でユーザーアイデンティティを有効にして、API またはコンソールを使用し、閲覧者または所有者アクセス許可を割り当てます。

- 閲覧者: クエリを発行し、検索に関連する提案された回答を受け取り、フィードバックを Amazon Kendra に提供して、検索を改善し続けることができます。
- 所有者: 検索ページのデザインをカスタマイズし、検索をチューニングし、検索アプリケーションを閲覧者として使用できます。コンソールで閲覧者へのアクセスを無効にすることは、現在サポートされていません。

他のユーザーに検索エクスペリエンスへのアクセス権を割り当てるには、まず、ExperienceConfiguration オブジェクトを使用して、Amazon Kendra エクスペリエ

ンスを持つ IAM Identity Center のユーザーアイデンティティを有効にします。ユーザー名や E メールアドレスなど、ユーザーの識別子を含むフィールド名を指定します。次に、AssociateEntitiesToExperience API を使用して、ユーザーのリストに検索エクスペリエンスへのアクセス権を付与し、AssociatePersonasToEntities API を使用して、アクセス許可を閲覧者または所有者に定義します。EntityConfiguration オブジェクトを使用して各ユーザーまたはグループを指定し、EntityPersonaConfiguraton オブジェクトを使用してそのユーザーまたはグループが閲覧者か所有者かを指定します。

コンソールを使用して他のユーザーに検索エクスペリエンスへのアクセス権を割り当てるには、まずエクスペリエンスを作成し、自分のアイデンティティと自分が所有者であることを確認する必要があります。その後、他のユーザーまたはグループを閲覧者または所有者として割り当てることができます。コンソールで、インデックスを選択し、ナビゲーションメニューの [Experiences] (エクスペリエンス) を選択します。エクスペリエンスを作成したら、リストからエクスペリエンスを選択できます。[Access management] (アクセス管理) に移動し、ユーザーまたはグループを閲覧者または所有者として割り当てます。

検索エクスペリエンスの設定

次に、検索エクスペリエンスを構成または作成する例を示します。

Console

Amazon Kendra 検索エクスペリエンスの作成方法

1. 左側のナビゲーションペインの [Indexes] (インデックス) で、[Experiences] (エクスペリエンス)、[Create experiences] (エクスペリエンスの作成) の順に選択します。
2. [Configure experience] (エクスペリエンスの設定) ページで、エクスペリエンスの名前と説明を入力し、コンテンツソースを選択し、エクスペリエンスの IAM ロールを選択します。IAM ロールの詳細については、「IAM roles for Amazon Kendra experiences」を参照してください。
3. [ディレクトリからアイデンティティを確認する] ページで、E メールなどのユーザー ID を選択します。Identity Center ディレクトリがない場合、フルネームと E メールを入力するだけで、Identity Center ディレクトリを作成できます。これには、エクスペリエンスのユーザーとしてのユーザーも含まれ、所有者のアクセス権が自動的に割り当てられます。
4. [Review to open Experience Builder] (確認してエクスペリエンスビルダーを開く) ページで、設定の詳細を確認し、[Create experience and open Experience Builder] (エクスペリエンスを作成し、エクスペリエンスビルダーを開く) をクリックして、検索ページの編集を開始します。

CLI

Amazon Kendra エクスペリエンスの作成方法

```
aws kendra create-experience \
 --name experience-name \
 --description "experience description" \
 --index-id index-id \
 --role-arn arn:aws:iam::account-id:role/role-name \
 --configuration '{"ExperienceConfiguration":[{"ContentSourceConfiguration":
{"DataSourceIds":["data-source-1","data-source-2"]},
 "UserIdentityConfiguration":"identity attribute name"}]}'

aws kendra describe-experience \
 --endpoints experience-endpoint-URL(s)
```

Python

Amazon Kendra エクスペリエンスの作成方法

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Create an experience.")

# Provide a name for the experience
name = "experience-name"
# Provide an optional description for the experience
description = "experience description"
# Provide the index ID for the experience
index_id = "index-id"
# Provide the IAM role ARN required for Amazon Kendra experiences
role_arn = "arn:aws:iam::${account-id}:role/${role-name}"
# Configure the experience
configuration = {"ExperienceConfiguration":
        [{
            "ContentSourceConfiguration":{"DataSourceIds":["data-source-1","data-
source-2"]},
            "UserIdentityConfiguration":"identity attribute name"
        }]
```

```
    }

try:
    experience_response = kendra.create_experience(
        Name = name,
        Description = description,
        IndexId = index_id,
        RoleArn = role_arn,
        Configuration = configuration
    )

    pprint.pprint(experience_response)

    experience_endpoints = experience_response["Endpoints"]

    print("Wait for Amazon Kendra to create the experience.")

    while True:
        # Get the details of the experience, such as the status
        experience_description = kendra.describe_experience(
            Endpoints = experience_endpoints
        )
        status = experience_description["Status"]
        print(" Creating experience. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

Amazon Kendra を作成するには

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateExperienceRequest;
import software.amazon.awssdk.services.kendra.model.CreateExperienceResponse;
import software.amazon.awssdk.services.kendra.model.DescribeExperienceRequest;
```

```
import software.amazon.awssdk.services.kendra.model.DescribeExperienceResponse;
import software.amazon.awssdk.services.kendra.model.ExperienceStatus;


public class CreateExperienceExample {

    public static void main(String[] args) throws InterruptedException {
        System.out.println("Create an experience");

        String experienceName = "experience-name";
        String experienceDescription = "experience description";
        String indexId = "index-id";
        String experienceRoleArn = "arn:aws:iam::account-id:role/role-name";

        KendraClient kendra = KendraClient.builder().build();

        CreateExperienceRequest createExperienceRequest = CreateExperienceRequest
            .builder()
            .name(experienceName)
            .description(experienceDescription)
            .roleArn(experienceRoleArn)
            .configuration(
                ExperienceConfiguration
                    .builder()
                    .contentSourceConfiguration(
                        ContentSourceConfiguration(
                            .builder()
                            .dataSourceIds("data-source-1","data-source-2")
                            .build()
                        )
                    )
                    .userIdentityConfiguration(
                        UserIdentityConfiguration(
                            .builder()
                            .identityAttributeName("identity-attribute-name")
                            .build()
                        )
                    ).build()
            ).build();

        CreateExperienceResponse createExperienceResponse =
 kendra.createExperience(createExperienceRequest);
        System.out.println(String.format("Experience response %s",
 createExperienceResponse));
```

```
        String experienceEndpoints = createExperienceResponse.endpoints();

        System.out.println(String.format("Wait for Kendra to create the
 experience.", experienceEndpoints));
        while (true) {
            DescribeExperienceRequest describeExperienceRequest =
 DescribeExperienceRequest.builder().endpoints(experienceEndpoints).build();
            DescribeExperienceResponse describeEpxerienceResponse =
 kendra.describeExperience(describeExperienceRequest);
            ExperienceStatus status = describeExperienceResponse.status();
            TimeUnit.SECONDS.sleep(60);
            if (status != ExperienceStatus.CREATING) {
                break;
            }
        }

        System.out.println("Experience creation is complete.");
    }
}
```

容量の調整

Amazon Kendra インデックス用のリソースをキャパシティユニットで提供します。各キャパシティユニットは、インデックスの追加リソースを提供します。ドキュメントストレージとクエリには、個別のキャパシティユニットがあります。 Amazon Kendra キャパシティユニットはエンタープライズエディションのインデックスにのみ追加できます。デベロッパーエディションのインデックスに容量を追加することはできません。

ドキュメントストレージキャパシティユニットは、インデックス用に次の追加のストレージを提供します。

- 100,000 ドキュメントか 30 GB のストレージ。

クエリキャパシティユニットは、インデックス用に次の追加のクエリを提供します。

- 1 秒あたり 0.1 クエリまたは 1 日あたり 8,000 クエリ。

各インデックスには、1 キャパシティーユニットに等しい基本キャパシティーが付属しています (30 GB のストレージ、毎秒 0.1 クエリ)。追加キャパシティユニットごとに追加料金が発生します。詳細については、「Amazon Kendra 料金表」を参照してください。

インデックス用のストレージおよびクエリリソースには、最大 100 の追加キャパシティーユニットを追加できます。さらにユニットが必要な場合は、サポートにお問い合わせください。

キャパシティーユニットを 1 日あたり最大 5 回調整して、使用要件に合わせることができます。インデックスに保存されているドキュメントの数を下回るドキュメントのストレージキャパシティを減らすことはできません。例えば、150,000 のドキュメントを保存する場合、ストレージキャパシティを 1 追加ユニット未満に減らすことはできません。

インデックスが使用しているリソースをコンソールで表示するには、インデックスの名前を選択してインデックスの設定やその他の情報を開くか、DescribeIndexAPI を使用できます。

Amazon Kendra また、インデックスの容量を超えると例外が返されます。すべてのドキュメントの抽出合計サイズがインデックスの制限を超えた場合、`ServiceQuotaExceededException` を受け取ります。ドキュメント数がインデックスの制限を超えた場合、各ドキュメントごとに、`InvalidRequest` を受け取ります。1 秒あたりのクエリ数が制限を超えた場合、`ThrottlingException` を受け取ります。制限の詳細については、「Amazon Kendraのクォータ」を参照してください。

蓄積されたクエリは最大 24 時間持続します。

容量の表示

インデックスの名前を選択して詳細にアクセスすると、 Amazon Kendra インデックスが使用しているリソースをコンソールで確認できます。コンソールには使用量のグラフが表示されるため、インデックスが使用するストレージおよびクエリキャパシティーを判断できます。この情報を使用して、キャパシティの追加計画を立てることができます。

ドキュメントストレージとクエリの使用を表示するには (コンソール)

1. AWS Management Console にサインインし、https://console.aws.amazon.com/kendra/home Amazon Kendra のコンソールを開きます。
2. インデックスのリストから、アクセスするインデックスを選択します。
3. 設定セクションまでスクロールすると、現在のドキュメントストレージとクエリの合計キャパシティーが表示されます。

API を使用して容量を表示するには、 Amazon Kendra DescribeIndexAPI `CapacityUnits` のパラメータを使用します。

キャパシティの追加および削除

インデックスの容量を追加する必要がある場合は、コンソールまたは Amazon Kendra API を使用して追加できます。

ストレージまたはクエリキャパシティーを追加または削除するには (コンソール)

1. AWS Management Console にサインインし、https://console.aws.amazon.com/kendra/home Amazon Kendra のコンソールを開きます。
2. インデックスのリストから、アクセスするインデックスを選択します。
3. [編集] を選択するか、[アクション] ドロップダウンから [編集] を選択します。
4. [次へ] を選択して、プロビジョニングの詳細ページに移動します。
5. ドキュメントストレージやクエリキャパシティーユニットを追加または削除します。
6. 続けて [次へ] を選択してレビューページに移動し、[更新] を選択して変更を保存します。

インデックスのキャパシティーを更新した後、変更が適用されるまで数分かかることがあります。

API を使用して容量を追加または削除するには、 Amazon Kendra UpdateIndexAPI `CapacityUnits` のパラメータを使用します。

Amazon Kendra インテリジェントランキングキャパシティ

キャパシティーユニットは、再スコア実行プランの 1 秒あたりの、次の再スコアリクエストを提供します。再スコア実行プランは Rescore API のプロビジョニングに使用されるリソースです。

- 1 秒あたり 0.01 のリクエストです。

各再スコア実行プランには、1 キャパシティーユニット (0.01 リクエスト/秒) に等しい基本キャパシティーがあります。追加キャパシティユニットごとに追加料金が発生します。詳細については、「Amazon Kendra 料金表」を参照してください。

最大 1000 個の追加キャパシティーユニットを再スコア実行プランに追加できます。さらにユニットが必要な場合は、サポートにお問い合わせください。

クエリ提案キャパシティ

クエリ候補を使用する場合、基本クエリ容量は 1 秒あたり 2.5 GetQuerySuggestionsコールです。`GetQuerySuggestions` キャパシティは、インデックスのプロビジョニングされたクエリキャパシティの 5 倍、または 1 秒あたり 2.5 コールのベースキャパシティのどちらか高い方です。例えば、インデックスのベースキャパシティは 1 秒あたり 0.1 クエリであり、`GetQuerySuggestions` キャパシティのベースは 1 秒あたり 2.5 コールです。インデックスの 1 秒あたりの合計 0.2 クエリに 1 秒あたり 0.1 クエリを追加すると、`GetQuerySuggestions` キャパシティは 1 秒あたり 2.5 コール (1 秒あたり 0.2 クエリの場合の 5 倍以上) になります。

Amazon Kendra エクスペリエンス容量

検索エクスペリエンスのキャパシティ

Amazon Kendra `Query` Amazon Kendra エクスペリエンスで はQuerySuggestions、SubmitFeedback 1 秒あたり 15 リクエスト、クエリバーストでは 1 秒あたり 40 リクエストでスロットリングを開始します。150 を超えるクエリキャパシティーユニットを持つインデックスの場合、これらの制限は引き続き適用されます。

例えば、インデックスのクエリキャパシティユニットは 150 であるため、検索エクスペリエンスアプリケーションでは 1 秒あたり 15 のリクエストを処理できます。ただし、クエリキャパシティユ

ニットを 200 にスケーリングしても、検索エクスペリエンスアプリケーションで処理されるのは 1 秒あたり 15 リクエストのみです。クエリキャパシティユニットを 100 にスケーリングすると、検索エクスペリエンスアプリケーションで処理されるのは 1 秒あたり 10 リクエストのみです。

適応型クエリバースト

Amazon Kendra には 1 クエリ容量ユニットの基本容量がプロビジョニングされています。1 日あたり最大 8,000 クエリを使用し、最小スループットは 1 秒あたり 0.1 クエリ (クエリキャパシティユニットあたり) です。蓄積されたクエリは最長 24 時間持続し、トラフィックの急増にも対応できます。許容されるバーストの量は、その時点でのクラスターの負荷によって異なるため、変化します。ピーク時の負荷レベルに対応できる十分なクエリキャパシティーユニットをプロビジョニングします。

プロビジョニングされたスループットを上回る予期しないトラフィックの急増を処理するための適応型アプローチとして、 Amazon Kendra組み込みのアダプティブクエリバーストがあります。適応型クエリバーストは、 Amazon Kendraの Enterprise Edition でのみ使用できます。

アダプティブクエリバーストは、未使用のクエリ容量を適用して予期しないトラフィックを処理できるようにする組み込み機能です。 Amazon Kendra インデックスにプロビジョニングしたクエリの最大数まで、未使用のクエリをプロビジョニングされたクエリの 1 秒あたりのレートで累積します。 Amazon Kendra これらの累積クエリは、割り当てられたキャパシティを超える予期しないトラフィックに使用されます。適応型クエリバーストの最適なパフォーマンスは、合計インデックスサイズ、クエリの複雑さ、累積された未使用のクエリ、インデックスの全体的なロードなど、いくつかの要因によって異なります。バーストキャパシティを正確に測定するには、独自のロードテストを実行することをお勧めします。

開始

このセクションでは、データソースを作成しドキュメントを Amazon Kendra インデックスに追加する方法について説明します。手順は、AWS コンソール、AWS CLI、AWS SDK for Python (Boto3) を使用する Python プログラム、AWS SDK for Java を使用する Java プログラムに関するものです。

トピック

前提条件

以下のステップは、入門ガイド演習の前提条件です。この手順では、アカウントを設定、ユーザーに代わって呼び出しを行うアクセス許可を Amazon Kendra に付与する IAM ロールを作成、Amazon S3 バケットからの文書にインデックスを作成する方法について説明します。例として S3 バケットを使用しますが、Amazon Kendra がサポートする他のデータソースを使用することもできます。「Data sources」を参照してください。

AWS アカウントへのサインアップ

AWS アカウントがない場合は、以下のステップを実行して作成します。

AWS アカウントにサインアップするには

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話のキーパッドを使用して検証コードを入力するように求められます。

AWS アカウントにサインアップすると、AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービスとリソースへのアクセス権があります。セキュリティのベストプラクティスとして、管理ユーザーに管理アクセスを割り当て、ルートユーザーアクセスが必要なタスクを実行する場合にのみ、ルートユーザーを使用してください。

サインアップ処理が完了すると、AWS からユーザーに確認メールが送信されます。https://aws.amazon.com/ の [アカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

管理ユーザーの作成

AWS アカウント にサインアップした後、AWS アカウントのルートユーザー を安全に保護し、AWS IAM Identity Center を有効にし、管理ユーザーを作成することで、日常的なタスクにルートユーザーを使用しないようにします。

AWS アカウントのルートユーザーをセキュリティで保護する

1. [ルートユーザー] を選択し、AWS アカウント のメールアドレスを入力して、アカウント所有者として AWS Management Console にサインインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、「AWS サインイン User Guide」の「Signing in as the root user」を参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、「IAM ユーザーガイド」の「AWS アカウントのルートユーザーの仮想 MFA デバイスを有効にする (コンソール)」を参照してください。

管理ユーザーを作成する

1. IAM Identity Center を有効にする

   説明については、「AWS IAM Identity Center ユーザーガイド」の「Enabling AWS IAM Identity Center」を参照してください。
2. IAM Identity Center で、管理ユーザーに管理者アクセスを付与します。

IAM アイデンティティセンターディレクトリ をアイデンティティソースとして使用するチュートリアルについては、「AWS IAM Identity Center ユーザーガイド」の「Configure user access with the default IAM アイデンティティセンターディレクトリ」を参照してください。

管理ユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

  IAM アイデンティティセンターのユーザーを使用してサインインする方法については、「AWS サインイン User Guide」の「Signing in to the AWS access portal」を参照してください。

- Amazon Kendra をテストするためにドキュメントを含む S3 バケットを使用している場合は、Amazon Kendra を使用しているのと同じリージョンに S3 バケットを作成します。手順については、Amazon Simple Storage Service コンソールユーザーガイドの S3 バケットの作成と設定を参照してください。

  ドキュメントを S3 バケットにアップロードします。手順については、Amazon Simple Storage Service ユーザーガイドのオブジェクトのアップロード、ダウンロードおよび管理を参照してください。

  別のデータソースを使用している場合は、データソースに接続するためのアクティブなサイトと認証情報が必要です。

コンソールを使用して開始する場合は、Amazon Kendra コンソールの使用をスタートする で開始します。

Amazon Kendra リソース: AWS CLI、SDK、コンソール

CLI、SDK、またはコンソールを使用する場合は、特定の権限が必要です。

CLI、SDK、またはコンソールに Amazon Kendra を使用するには、ユーザーに代わってリソースを作成および管理できる権限を Amazon Kendra に許可する必要があります。ユースケースによっては、これらの権限には、Amazon Kendra API 自体へ、カスタム CMK を使用してデータを暗号化する場合は AWS KMS keys へ、AWS IAM Identity Center と統合または検索エクスペリエンスを作成する場合は Identity Center ディレクトリへのアクセスが含まれます。さまざまなユースケースのための権限の一覧については、「IAM roles」を参照してください。

まず、以下のアクセス許可を IAM ユーザーにアタッチする必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1644430853544",
      "Action": [
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1644430878150",
      "Action": "kendra:*",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1644430973706",
      "Action": [
        "sso:AssociateProfile",
        "sso:CreateManagedApplicationInstance",
        "sso:DeleteManagedApplicationInstance",
        "sso:DisassociateProfile",
        "sso:GetManagedApplicationInstance",
        "sso:GetProfile",
        "sso:ListDirectoryAssociations",
        "sso:ListProfileAssociations",
        "sso:ListProfiles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1644430999558",
      "Action": [
        "sso-directory:DescribeGroup",
        "sso-directory:DescribeGroups",
        "sso-directory:DescribeUser",
        "sso-directory:DescribeUsers"
      ],
```

```
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "Stmt1644431025960",
      "Action": [
        "identitystore:DescribeGroup",
        "identitystore:DescribeUser",
        "identitystore:ListGroups",
        "identitystore:ListUsers"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

次に、CLI または SDK を使用する場合は、IAM にアクセスする Amazon CloudWatch Logs ロールとポリシーも作成する必要があります。コンソールを使用している場合は、このために IAM ロールとポリシーを作成する必要はありません。これはコンソール手順の一部として作成します。

Amazon Kendra が Amazon CloudWatch Logs にアクセスできるようにする、AWS CLI および SDK 用の IAM ロールとポリシーを作成する方法。

1. AWS Management Console にサインインして、IAM コンソール (https://console.aws.amazon.com/iam/) を開きます。
2. 左側のメニューで、[Policies] (ポリシー) を選択し、[Create policy] (ポリシーの作成) を選択します。
3. [JSON] を選択し、デフォルトのポリシーを以下に置き換えます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
```

```
                    "cloudwatch:namespace": "AWS/Kendra"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup"
            ],
            "Resource": [
                "arn:aws:logs:region:account ID:log-group:/aws/kendra/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogStreams",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account ID:log-group:/aws/kendra/*:log-
stream:*"
            ]
        }
    ]
}
```

4. [Review policy] (ポリシーの確認) を選択します。
5. ポリシー "KendraPolicyForGettingStartedIndex" に名前を付け、[ポリシーを作成] を選択します。
6. 左側のメニューから、[Roles] (ロール) を選択し、[Create role] (ロールの作成) を選択します。
7. [Another AWS account] (別の アカウント) を選択し、[Account ID] (アカウント ID) にアカウント ID を入力します。[Next: Permissions] (次へ: アクセス許可) を選択します。
8. 上記の手順で作成したポリシーを選択し、[Next: Tags] (次へ: タグ) を選択します。

9. タグを追加しないでください。[Next: Review] (次へ: レビュー)を選択します。
10. ロール "KendraRoleForGettingStartedIndex" に名前を付け、[ロールの作成] を選択します。
11. 先ほど作成したロールを検索します。ロールの名前を選択し、[概要] を開きます。[Trust relationships] (信頼関係) を選択し、[Edit trust relationship] (信頼関係の編集) を選択します。
12. 既存の信頼関係を、次のものに置き換えます。

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

13. [Update trust policy] (信頼ポリシーの更新) を選択します。

3 つ目に、Amazon S3 を使用してドキュメントを保存する場合や、S3 を使用して Amazon Kendra をテストする場合は、バケットにアクセスするための IAM ロールとポリシーも作成する必要があります。別のデータソースを使用している場合は、「IAM roles for data sources」を参照してください。

Amazon Kendra が Amazon S3 バケットにアクセスできるようにする、IAM ロールとポリシーを作成する方法。

1. AWS Management Console にサインインして、IAM コンソール (https://console.aws.amazon.com/iam/) を開きます。
2. 左側のメニューで、[Policies] (ポリシー) を選択し、[Create policy] (ポリシーの作成) を選択します。
3. [JSON] を選択し、デフォルトのポリシーを以下に置き換えます。

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

```
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket name/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket name"
            ],
            "Effect": "Allow"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kendra:BatchPutDocument",
                "kendra:BatchDeleteDocument"
            ],
            "Resource": "arn:aws:kendra:region:account ID:index/*"
        }
    ]
}
```

4. [Review policy] (ポリシーの確認) を選択します。
5. ポリシーに「KendraPolicyForGettingStartedDataSource」という名前を付けて、[Create policy] (ポリシーの作成) を選択します。
6. 左側のメニューから、[Roles] (ロール) を選択し、[Create role] (ロールの作成) を選択します。
7. [Another AWS account] (別の アカウント) を選択し、[Account ID] (アカウント ID) にアカウント ID を入力します。[Next: Permissions] (次へ: アクセス許可) を選択します。
8. 上記の手順で作成したポリシーを選択し、[Next: Tags] (次へ: タグ) を選択します。
9. タグを追加しないでください。[Next: Review] (次へ: レビュー)を選択します。
10. ロールに「KendraRoleForGettingStartedDataSource」という名前を付けて、[Create role] (ロールの作成) を選択します。
11. 先ほど作成したロールを検索します。ロールの名前を選択し、[概要] を開きます。[Trust relationships] (信頼関係) を選択し、[Edit trust relationship] (信頼関係の編集) を選択します。

12. 既存の信頼関係を、次のものに置き換えます。

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

13. [Update trust policy] (信頼ポリシーの更新) を選択します。

Amazon Kendra API の使用方法に応じて、次のいずれかの操作を実行します。

- 開始方法 (AWS CLI)
- 開始方法 (AWS SDK for Java)
- 開始方法 (AWS SDK for Python (Boto3))

Amazon Kendra コンソールの使用をスタートする

次の手順は、AWS コンソールを使用して Amazon Kendra インデックスを作成し、テストする方法を示しています。この手順では、インデックスのインデックスとデータソースを作成します。最後に、検索リクエストを作成してインデックスをテストします。

ステップ 1: インデックスを作成するには (コンソール)

1. AWS マネジメントコンソールにサインインして https://console.aws.amazon.com/kendra/ の Amazon Kendra コンソールを開きます。
2. [インデックス] セクションで、[インデックスの作成] を選択します。
3. [インデックスの詳細の指定] ページで、インデックスに名前と説明を付けます。
4. [IAM ロール] で、[新しいロールを作成] を選択してから、ロールに名前を付けます。IAM ロールには、プレフィックス「AmazonKendra-」が付いています。

5. その他のフィールドはすべてデフォルトのままにしておきます。[Next] (次へ) をクリックします。
6. [Configure user access control] (ユーザーアクセスコントロールの設定) ページで、[Next] (次へ) をクリックします。
7. [Provisioning details] (プロビジョニングの詳細) ページで、[Developer edition] (デベロッパーエディション) を選択します。
8. [Create] (作成) を選択してインデックスを作成します。
9. インデックスが作成されるまで待ちます。Amazon Kendra はインデックスのハードウェアをプロビジョニングします。この演算には時間がかかる場合があります。

ステップ 2: データソースをインデックスに追加するには (コンソール)

1. Amazon Kendra をドキュメントに接続してインデックスを作成できるデータソースを表示します。
2. ナビゲーションペインで [データソース] を選択してから、選択したデータソースの [データソースを追加] を選択します。
3. 手順に従ってデータソースを設定します。

ステップ 3: インデックスを検索するには (コンソール)

1. ナビゲーションペインで、インデックスを検索するオプションを選択します。
2. インデックスに適した検索用語を入力します。上位の結果および上位のドキュメント結果が表示されます。

開始方法 (AWS CLI)

次の手順は、AWS CLI を使用して Amazon Kendra インデックスを作成する方法を示しています。手順は、データソースとインデックスを作成し、インデックスに対してクエリを実行します。

Amazon Kendra インデックスを作成するには (CLI)

1. 前提条件 を実行します。
2. 以下のコマンドを入力してインデックスを作成します。

```
aws kendra create-index \
```

```
 --name cli-getting-started-index \
 --description "Index for CLI getting started guide." \
 --role-arn arn:aws:iam::account id:role/KendraRoleForGettingStartedIndex
```

3. Amazon Kendra がインデックスを作成するのを待ちます。次のコマンドを使用してポリシーをチェックします。ステータスフィールドが ACTIVE の場合、次のステップに進みます。

```
aws kendra describe-index \
 --id index id
```

4. コマンドプロンプトで、次のコマンドを入力してデータソースを作成します。

```
aws kendra create-data-source \
 --index-id index id \
 --name data source name \
 --role-arn arn:aws:iam::account id:role/KendraRoleForGettingStartedDataSource \
 --type S3 \
 --configuration '{"S3Configuration":{"BucketName":"S3 bucket name"}}'
```

テンプレートスキーマを使用してデータソースに接続する場合は、テンプレートスキーマを設定します。

```
aws kendra create-data-source \
 --index-id index id \
 --name data source name \
 --role-arn arn:aws:iam::account id:role/KendraRoleForGettingStartedDataSource \
 --type TEMPLATE \
 --configuration '{"TemplateConfiguration":{"Template":{JSON schema}}}'
```

5. Amazon Kendra がデータソースを作成するのにはしばらくかかります。次のコマンドを入力してプロセスをチェックします。ステータスが ACTIVE の場合、次のステップに進みます。

```
aws kendra describe-data-source \
 --id data source ID \
 --index-id index ID
```

6. 次のコマンドを入力して、データソースを同期します。

```
aws kendra start-data-source-sync-job \
 --id data source ID \
 --index-id index ID
```

7. Amazon Kendra はデータソースにインデックスを作成します。かかる時間は、ドキュメントの数によって異なります。次のコマンドを使用して、ジョブ同期のステータスをチェックできます。ステータスが ACTIVE の場合、次のステップに進みます。

```
aws kendra describe-data-source \
 --id data source ID \
 --index-id index ID
```

8. 次のコマンドを入力してクエリを保存します。

```
aws kendra query \
 --index-id index ID \
 --query-text "search term"
```

検索の結果が JSON 形式で表示されます。

開始方法 (AWS SDK for Python (Boto3))

次のプログラムは、Python プログラムで Amazon Kendra を使用する例です。このプログラムでは次のアクションを実行しています。

1. CreateIndex 演算を使用して新しいインデックスを作成します。
2. インデックスの作成が完了するのを待ちます。これは、DescribeIndex 演算を使用して、インデックスのステータスをモニタリングします。
3. インデックスがアクティブになると、このインデックスは、CreateDataSource 演算を使用してデータソースを作成します。
4. データソースの作成が完了するのを待ちます。これは、DescribeDataSource 演算を使用して、データソースのステータスをモニタリングします。
5. データソースがアクティブになると、StartDataSourceSyncJob 演算を使用して、インデックスがデータソースの内容と同期されます。

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")
```

```
print("Create an index.")

# Provide a name for the index
index_name = "python-getting-started-index"
# Provide an optional decription for the index
description = "Getting started index"
# Provide the IAM role ARN required for indexes
index_role_arn = "arn:aws:iam::${accountId}:role/KendraRoleForGettingStartedIndex"

try:
    index_response = kendra.create_index(
        Description = description,
        Name = index_name,
        RoleArn = index_role_arn
    )

    pprint.pprint(index_response)

    index_id = index_response["Id"]

    print("Wait for Amazon Kendra to create the index.")

    while True:
        # Get the details of the index, such as the status
        index_description = kendra.describe_index(
            Id = index_id
        )
        # When status is not CREATING quit.
        status = index_description["Status"]
        print(" Creating index. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break

    print("Create an S3 data source.")

    # Provide a name for the data source
    data_source_name = "python-getting-started-data-source"
    # Provide an optional description for the data source
    data_source_description = "Getting started data source."
    # Provide the IAM role ARN required for data sources
    data_source_role_arn = "arn:aws:iam::${accountId}:role/
KendraRoleForGettingStartedDataSource"
```

```
    # Provide the data source connection information
    S3_bucket_name = "S3-bucket-name"
    data_source_type = "S3"
    # Configure the data source
    configuration = {"S3Configuration":
        {
            "BucketName": S3_bucket_name
        }
    }

    """
    If you connect to your data source using a template schema,
    configure the template schema
    configuration = {"TemplateConfiguration":
        {
            "Template": {JSON schema}
        }
    }
    """

    data_source_response = kendra.create_data_source(
        Name = data_source_name,
        Description = data_source_name,
        RoleArn = data_source_role_arn,
        Type = data_source_type,
        Configuration = configuration,
        IndexId = index_id
    )

    pprint.pprint(data_source_response)

    data_source_id = data_source_response["Id"]

    print("Wait for Amazon Kendra to create the data source.")

    while True:
        # Get the details of the data source, such as the status
        data_source_description = kendra.describe_data_source(
            Id = data_source_id,
            IndexId = index_id
        )
        # If status is not CREATING, then quit
        status = data_source_description["Status"]
        print(" Creating data source. Status: "+status)
```

```
        time.sleep(60)
        if status != "CREATING":
            break

    print("Synchronize the data source.")

    sync_response = kendra.start_data_source_sync_job(
        Id = data_source_id,
        IndexId = index_id
    )

    pprint.pprint(sync_response)

    print("Wait for the data source to sync with the index.")

    while True:

        jobs = kendra.list_data_source_sync_jobs(
            Id = data_source_id,
            IndexId = index_id
        )

        # For this example, there should be one job
        status = jobs["History"][0]["Status"]

        print(" Syncing data source. Status: "+status)
        if status != "SYNCING":
            break
        time.sleep(60)

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

開始方法 (AWS SDK for Java)

次のプログラムは、Java プログラムで Amazon Kendra を使用する例です。このプログラムでは次のアクションを実行しています。

1. CreateIndex 演算を使用して新しいインデックスを作成します。

2. インデックスの作成が完了するのを待ちます。これは、DescribeIndex 演算を使用して、インデックスのステータスをモニタリングします。
3. インデックスがアクティブになると、このインデックスは、CreateDataSource 演算を使用してデータソースを作成します。
4. データソースの作成が完了するのを待ちます。これは、DescribeDataSource 演算を使用して、データソースのステータスをモニタリングします。
5. データソースがアクティブになると、StartDataSourceSyncJob 演算を使用して、インデックスがデータソースの内容と同期されます。

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.CreateIndexRequest;
import software.amazon.awssdk.services.kendra.model.CreateIndexResponse;
import software.amazon.awssdk.services.kendra.model.DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.DataSourceStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJob;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJobStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceType;
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.DescribeIndexRequest;
import software.amazon.awssdk.services.kendra.model.DescribeIndexResponse;
import software.amazon.awssdk.services.kendra.model.IndexStatus;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsRequest;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsResponse;
import software.amazon.awssdk.services.kendra.model.S3DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobRequest;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobResponse;


public class CreateIndexAndDataSourceExample {

    public static void main(String[] args) throws InterruptedException {
        System.out.println("Create an index");

        String indexDescription = "Getting started index for Kendra";
```

```
        String indexName = "java-getting-started-index";
        String indexRoleArn = "arn:aws:iam::<your AWS account ID>:role/<name of an IAM
role>";

        System.out.println(String.format("Creating an index named %s", indexName));
        KendraClient kendra = KendraClient.builder().build();

        CreateIndexRequest createIndexRequest = CreateIndexRequest
            .builder()
            .description(indexDescription)
            .name(indexName)
            .roleArn(indexRoleArn)
            .build();
        CreateIndexResponse createIndexResponse =
kendra.createIndex(createIndexRequest);
        System.out.println(String.format("Index response %s", createIndexResponse));

        String indexId = createIndexResponse.id();

        System.out.println(String.format("Waiting until the index with index ID %s is
created", indexId));
        while (true) {
            DescribeIndexRequest describeIndexRequest =
DescribeIndexRequest.builder().id(indexId).build();
            DescribeIndexResponse describeIndexResponse =
kendra.describeIndex(describeIndexRequest);
            IndexStatus status = describeIndexResponse.status();
            if (status != IndexStatus.CREATING) {
                break;
            }

            TimeUnit.SECONDS.sleep(60);
        }

        System.out.println("Creating an S3 data source");
        String dataSourceName = "java-getting-started-data-source";
        String dataSourceDescription = "Getting started data source";
        String s3BucketName = "an-aws-kendra-test-bucket";
        String dataSourceRoleArn = "arn:aws:iam::<your AWS account ID>:role/<name of an
IAM role>";

        CreateDataSourceRequest createDataSourceRequest = CreateDataSourceRequest
            .builder()
            .indexId(indexId)
```

```
            .name(dataSourceName)
            .description(dataSourceDescription)
            .roleArn(dataSourceRoleArn)
            .type(DataSourceType.S3)
            .configuration(
                DataSourceConfiguration
                    .builder()
                    .s3Configuration(
                        S3DataSourceConfiguration
                            .builder()
                            .bucketName(s3BucketName)
                            .build()
                    ).build()
            ).build();

       CreateDataSourceResponse createDataSourceResponse =
kendra.createDataSource(createDataSourceRequest);
       System.out.println(String.format("Response of creating data source: %s",
createDataSourceResponse));

       String dataSourceId = createDataSourceResponse.id();
       System.out.println(String.format("Waiting for Kendra to create the data source
%s", dataSourceId));
       DescribeDataSourceRequest describeDataSourceRequest = DescribeDataSourceRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

       while (true) {
           DescribeDataSourceResponse describeDataSourceResponse =
kendra.describeDataSource(describeDataSourceRequest);

           DataSourceStatus status = describeDataSourceResponse.status();
           System.out.println(String.format("Creating data source. Status: %s",
status));
           if (status != DataSourceStatus.CREATING) {
               break;
           }

           TimeUnit.SECONDS.sleep(60);
       }
```

```
        System.out.println(String.format("Synchronize the data source %s",
 dataSourceId));
        StartDataSourceSyncJobRequest startDataSourceSyncJobRequest =
 StartDataSourceSyncJobRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();
        StartDataSourceSyncJobResponse startDataSourceSyncJobResponse =
 kendra.startDataSourceSyncJob(startDataSourceSyncJobRequest);
        System.out.println(String.format("Waiting for the data
 source to sync with the index %s for execution ID %s", indexId,
 startDataSourceSyncJobResponse.executionId()));

        // For this particular list, there should be just one job
        ListDataSourceSyncJobsRequest listDataSourceSyncJobsRequest =
 ListDataSourceSyncJobsRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

        while (true) {
            ListDataSourceSyncJobsResponse listDataSourceSyncJobsResponse =
 kendra.listDataSourceSyncJobs(listDataSourceSyncJobsRequest);
            DataSourceSyncJob job = listDataSourceSyncJobsResponse.history().get(0);
            System.out.println(String.format("Syncing data source. Status: %s",
 job.status()));

            if (job.status() != DataSourceSyncJobStatus.SYNCING) {
                break;
            }

            TimeUnit.SECONDS.sleep(60);

        }

        System.out.println("Index setup is complete");
    }
}
```

Amazon S3 データソースの開始方法 (コンソール)

Amazon Kendra コンソールを使用して、データストアとして Amazon S3 バケットの使用を開始できます。コンソールを使用して、バケットのコンテンツのインデックス作成に必要なすべての接続情報を指定できます。詳細については、「Amazon S3」を参照してください。

デフォルト設定を使用して基本的な S3 バケットデータソースを作成するには、次の手順に従います。この手順では、Amazon Kendra コンソールの使用をスタートする のステップ 1 の手順に従ってインデックスを作成していることを前提としています。

Amazon Kendra コンソールを使用して S3 バケットデータソースを作成するには

1. AWS Management Console にサインインし、Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/home) を開きます。
2. インデックスのリストから、データソースを追加するインデックスを選択します。
3. [Add data sources] (データソースの追加) を選択します。
4. データソースコネクタのリストから、[Amazon S3] を選択します。
5. [Define attributes] (属性の定義) ページで、データソースに名前と説明 (オプション) を入力します。[Tags] (タグ) フィールドは空白のままにします。[Next] (次へ) を選択して続行します。
6. [Enter the data source location] (データソースの場所を入力) フィールドに、ドキュメントが含まれる S3 バケットの名前を入力します。名前を直接入力することも、[Browse] (ブラウズ) を選択して名前を参照することもできます。バケットはインデックスと同じリージョンにある必要があります。
7. [IAMIAM ロール] で、[新しいロールを作成] を選択し、ロールの名前を入力します。詳細については、「IAM roles for Amazon S3 data sources」を参照してください。
8. [Set sync run schedule] (同期実行スケジュールの設定) セクションで、[Run on demand] (オンデマンドで実行する) を選択します。
9. [Next] (次へ) を選択して続行します。
10. [Review and create] (確認と作成) ページで、S3 データソースの詳細を確認します。変更が必要な場合は、変更する項目の隣にある [Edit] (編集) ボタンを押します。選択した内容が正しければ、[Create] (作成) をクリックして S3 データソースを作成します。

[作成] を選択すると、Amazon Kendra がデータソースの作成を開始します。データソースが作成されるまでに数分かかる場合があります。終了すると、データソースのステータスは [Creating] (作成中) から [Active] (アクティブ) に変わります。

データソースを作成したら、Amazon Kendra インデックスをデータソースと同期する必要があります。[Sync now] (今すぐ同期) をクリックして、同期プロセスを開始します。ドキュメントの数とサイズによっては、データソースの同期に数分から数時間かかる場合があります。

MySQL データベースデータソースの開始方法 (コンソール)

Amazon Kendra コンソールを使用して、MySQL データベースのデータソースとしての使用を開始できます。コンソールを使用する場合は、MySQL データベースのコンテンツのインデックス作成をするために必要な接続情報を指定します。詳細については、データベースデータソースの使用を参照してください。

まず MySQL データベースを作成してから、データベースのデータソースを作成できます。

次の手順に従って、基本的な MySQL データベースを作成します。この手順では、Amazon Kendra コンソールの使用をスタートする のステップ 1 に従ってすでにインデックスを作成していることを前提としています。

MySQL データベースを作成するには

1. AWS Management Console にサインインし、Amazon RDS コンソール (https://console.aws.amazon.com/rds/) を開きます。
2. ナビゲーションペインで、[Subnet groups] (サブネットグループ)、[Create DB Subnet Group] (DB サブネットグループの作成) の順に選択します。
3. グループに名前を付けて、Virtual Private Cloud (VPC) を選択します。VPC の設定方法の詳細については、「Configuring Amazon Kendra to use a VPC」を参照してください。
4. VPC のプライベートサブネットを追加します。プライベートサブネットは、NAT に接続されていないサブネットです。[Create] (作成) を選択します。
5. ナビゲーションペインで、[Databases] (データベース)、[Create database] (データベースの作成) の順に選択します。
6. データベースを作成するには、以下のパラメータを使用します。その他のパラメータはすべてデフォルトのままにしておきます。

   - エンジンのオプション - MySQL
   - テンプレート - 無料利用枠
   - 認証情報の設定 - パスワードを入力して確認する
   - [Connectivity] (接続性) で、[Additional connectivity configuration] (追加の接続性設定) をクリックします。以下の選択を行います。

- サブネットグループ - ステップ 4 で作成したサブネットグループを選択します。
- VPC セキュリティグループ - VPC で作成したインバウンドルールとアウトバウンドルールの両方を含むグループを選択します。例えば、**DataSourceSecurityGroup**。VPC の設定方法の詳細については、「Configuring Amazon Kendra to use a VPC」を参照してください。
- [Additional configuration] (追加の設定) の [Initial database name] (初期データベース名)] を **content** に設定します。

7. [Create database] (データベース) の作成を選択します。
8. データベースのリストから、新しいデータベースを選択します。データベースエンドポイントを書きとめておきます。
9. データベースを作成した後、ドキュメントを格納するためのテーブルを作成する必要があります。テーブルの作成は、これらの手順の適用範囲外です。テーブルを作成する際は、以下の点に注意してください。

   - データベース名 - **content**
   - テーブル名 - **documents**
   - 列 - **ID**、**Title**、**Body**、**LastUpdate**。必要に応じて追加の列を含めることができます。

これで MySQL データベースを作成したので、データベースのデータソースを作成できます。

MySQL データソースを作成するには

1. AWS Management Console にサインインして Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/home) を開きます。
2. ナビゲーションペインで、[Indexes] (インデックス)を選択してインデックスを選択します。
3. [Add data sources] (データソースの追加) を選択して [Amazon RDS] を選択します。
4. データソースの名前と説明を入力し、[Next] (次へ) を選択します。
5. [MySQL] を選択します。
6. [Connection access] (接続アクセス) で以下の情報を入力します。

   - [エンドポイント] - 前に作成したデータベースのエンドポイント。
   - [ポート] - データベースのポート番号。MySQL のデフォルトポートは 3306 です。
   - [認証のタイプ] - [新規] を選択します。

- [新しいシークレットコンテナ名] - データベース認証情報の Secrets Manager コンテナの名前。
- [ユーザー名] - データベースへの管理者アクセス権を持つユーザーの名前。
- [パスワード] - ユーザーのパスワードで、入力後に [認証を保存] を選択します。
- [データベース名] - **`content`**。
- [テーブル名] - **`documents`**。
- [IAM ロール] - [新しいロールの作成] を選択してから、ロールの名前を入力します。

7. [列設定] に、次の内容を入力します。
   - [ドキュメント ID 列名] - **`ID`**
   - [ドキュメントタイトル列名] - **`Title`**
   - [ドキュメントデータ列名] - **`Body`**
8. [Column change detection] (列変更検出) に、次のように入力します。
   - [検出列の変更] - **`LastUpdate`**
9. [Configure VPC & security group] (VPC とセキュリティグループを設定する) で、以下を指定します。
   - [Virtual Private Cloud (VPC)] で、VPC を選択します。
   - [Subnets] (サブネット) で、VPC で作成したプライベートサブネットを選択します。
   - [VPC security groups] (VPC セキュリティグループ) で、MySQL の VPC で作成したインバウンドルールとアウトバウンドルールの両方を含むセキュリティグループを選択します。例えば、**`DataSourceSecurityGroup`**。
10. [Set sync run schedule] (同期実行スケジュールの設定) で、[Run on demand] (オンデマンドで実行)、[Next] (次へ) の順に選択します。
11. [Data source field mapping] (データソースフィールドのマッピング) で、[Next] (次へ) をクリックします。
12. データソースの設定が正しいことを確認します。すべてが正しいことを確認したら、[Create] (作成) をクリックします。

AWS IAM Identity Center ID ソースの開始方法 (コンソール)

AWS IAM Identity Center ID ソースには、ユーザーとグループに関する情報が含まれています。これは、ユーザーコンテキストフィルタリングを設定する場合に便利です。 は、ユーザーまたはそのグ

ループのドキュメントへのアクセスに基づいて、さまざまなユーザーの検索結果を Amazon Kendra フィルタリングします。

IAM Identity Center のアイデンティティソースを作成するには、IAM Identity Center をアクティベートし、 AWS Organizations で組織を作成する必要があります。IAM Identity Center をアクティベートし、組織を初めて作成すると、デフォルトのアイデンティティソースとして Identity Center ディレクトリが自動的に作成されます。アイデンティティソースとしてアクティブディレクトリ (Amazon が管理するまたは自己管理の) または外部アイデンティティプロバイダーに変更できます。このためには、正しいガイダンスに従う必要があります。「Changing your IAM Identity Center identity source」を参照してください。ID ソースは 組織あたり 1 つのみ持つことができます。

ユーザーとグループにドキュメントに対するさまざまなレベルのアクセス権を割り当てるには、ドキュメントをインデックスに取り込むときに、アクセスコントロールリストにユーザーとグループを含める必要があります。これにより、ユーザーとグループは、アクセスレベルに応じて Amazon Kendra でドキュメントを検索できます。クエリを発行する場合、ユーザー ID は、IAM Identity Center のユーザー名と完全に一致する必要があります。

また、 で IAM Identity Center を使用するために必要なアクセス許可を付与する必要があります Amazon Kendra。詳細については、「IAM roles for IAM Identity Center」を参照してください。

IAM Identity Center アイデンティティソースを設定するには

1. IAM Identity Center コンソールを開きます。
2. 「IAM Identity Center を有効にする」を選択し、 AWS 「組織の作成」を選択します。

   Identity Center ディレクトリがデフォルトで作成され、組織に関連付けられている E メールアドレスを検証するための E メールが送信されます。
3. AWS 組織にグループを追加するには、ナビゲーションペインでグループ を選択します。
4. [Groups page] (グループページ) で、[Create group] (グループの作成) を選択し、ダイアログボックスにグループ名と説明を入力します。[Create] (作成) を選択します。
5. 組織にユーザーを追加するには、ナビゲーションペインで [ユーザー] を選択します。
6. [Users] (ユーザー) ページで、[Add user] (ユーザーを追加) を選択します。[User details] (ユーザーの詳細) で、すべての必須フィールドを指定します。[Password] (パスワード) で、[Send an email to the user] (ユーザーにメールを送信) を選択します。[次へ] を選択します。
7. ユーザーをグループに追加するには、[Groups] (グループ) をクリックし、グループを選択します。

8. グループの [Details] (詳細) ページにある [Group members] (グループメンバー) で、[Add user] (ユーザーの追加) をクリックします。
9. [Add users to group] (ユーザーをグループに追加) ページで、グループのメンバーとして追加するユーザーを選択します。複数のユーザーを選択してグループに追加できます。
10. ユーザーおよびグループのリストを IAM Identity Center と同期するには、アイデンティティソースをアクティブディレクトリまたは外部アイデンティティプロバイダーに変更します。

    Identity Center ディレクトリはデフォルトのアイデンティティソースであり、プロバイダーによって管理される独自のリストがない場合は、このソースを使用してユーザーおよびグループを手動で追加する必要があります。アイデンティティソースを変更するには、正しいガイダンスに従う必要があります。「Changing your IAM Identity Center identity source」を参照してください。

> **Note**
>
> アクティブディレクトリまたは外部アイデンティティプロバイダーをアイデンティティソースとして使用する場合は、ユーザーの E メールアドレスをクロスドメインアイデンティティ管理 (SCIM) プロトコルを指定する際の IAM Identity Center ユーザー名にマッピングする必要があります。詳細については、「IAM Identity Center guide on SCIM for enabling IAM Identity Center」を参照してください。

IAM Identity Center アイデンティティソースをセットアップしたら、インデックスの作成または編集時に、コンソールでこれを有効にできます。インデックスの設定の [ユーザーアクセスコントロール] に移動し、IAM Identity Center からユーザーグループ情報を取得できるように設定を編集します。

UserGroupResolutionConfiguration オブジェクトを使用して IAM Identity Center を有効化することもできます。を `UserGroupResolutionMode`として指定`AWS_SSO`し、、`sso:ListDirectoryAssociations`、`sso-directory:SearchUsers`、 を呼び出すアクセス許可を付与する IAM ロールを作成します`sso-directory:ListGroupsForUsers``sso-directory:DescribeGroups`。

> **Warning**
>
> Amazon Kendra は現在、IAM Identity Center アイデンティティソースの組織メンバーアカウント`UserGroupResolutionConfiguration`での AWS の使用をサポートしていませ

ん。UserGroupResolutionConfiguration を使用するには、その組織の管理アカウントでインデックスを作成する必要があります。

次のものは、UserGroupResolutionConfiguration でデータソースを設定する方法、およびユーザーコンテキストで検索結果をフィルタリングするユーザーアクセス制御の概要です。これは、インデックスのインデックスと IAM ロールが既に作成されていることを前提としています。CreateIndex API を使用してインデックスを作成し、 IAM ロールを指定します。

UserGroupResolutionConfiguration およびユーザーコンテキストフィルタリングによるデータソースの設定

1. IAM Identity Center アイデンティティソースにアクセスする権限を付与する IAM ロールを作成します。
2. モードを UserGroupResolutionConfigurationに設定AWS_SSOし、 を呼び出しUpdateIndexて IAM Identity Center を使用するようにインデックスを更新します。
3. トークンベースのユーザーアクセスコントロールを使用してユーザーコンテキストで検索結果をフィルタリングする場合は、 を呼び出すUSER_TOKENときに UserContextPolicyを に設定しますUpdateIndex。それ以外の場合は、ほとんどのデータソースコネクタの各ドキュメントのアクセスコントロールリストを Amazon Kendra クロールします。また、UserContext にユーザーやグループの情報を入力して、クエリ API のユーザーコンテキストに基づいて検索結果をフィルターすることもできます。また、 を使用してユーザーをグループにマッピングPutPrincipalMappingして、クエリを発行するときにユーザー ID を指定するだけで済みます。
4. データソースにアクセスするためのアクセス許可を付与する IAM ロール を作成します。
5. データソースを設定します。データソースに接続するために必要な接続情報を提供する必要があります。
6. CreateDataSource API を使用してデータソースを作成します。TemplateConfiguration、インデックスの ID、データソースの IAM ロール、データソースタイプを含む DataSourceConfiguration オブジェクトを指定し、データソースに名前を付けます。データソースを更新することもできます。

IAM Identity Center アイデンティティソースの変更

> **Warning**
>
> IAM Identity Center の [設定] でアイデンティティソースを変更すると、ユーザーおよびグループ情報の保存が影響を受ける可能性があります。これを安全に行うには、アイデンティティソースの変更に関する考慮事項を確認することをお勧めします。アイデンティティソースを変更する場合、新しいアイデンティティソース ID が生成されます。AWS_SSO でモードを に設定する前に、正しい ID を使用していることを確認してくださいUserGroupResolutionConfiguration。

IAM Identity Center アイデンティティソースを変更するには

1. [IAM Identity Center] > [コンソール] を開きます。
2. [Settings] (設定) を選択します。
3. [Settings] (設定) ページの [Identity source] (アイデンティティソース) で、[Change] (変更) を選択します。
4. [Change identity source] (アイデンティティソースの変更) ページで、優先するアイデンティティソースを選択し、[Next] (次へ) をクリックします。

インデックスの作成

インデックスは、コンソールを使用するか、 CreateIndex API を呼び出して作成できます。 AWS Command Line Interface （ AWS CLI) または SDK を API で使用できます。インデックスを作成したら、ドキュメントをインデックスに直接追加したり、データソースからドキュメントを追加したりできます。

インデックスを作成するには、インデックスが にアクセスするための () ロールの Amazon リソースネーム AWS Identity and Access Management (ARNIAM) を指定する必要があります CloudWatch。詳細については、「IAM roles for indexes」を参照してください。

次のタブでは、、 AWS Management Console、Python および Java SDK を使用してインデックスを作成する手順と AWS CLI、 のコード例を示します。 SDKs

Console

インデックスを作成するには

1. AWS マネジメントコンソールにサインインし、https://console.aws.amazon.com/kendra/ で Amazon Kendra コンソールを開きます。
2. [インデックス] セクションで、[インデックスの作成] を選択します。
3. [インデックスの詳細の指定] で、インデックスに名前と説明を付けます。
4. IAM ロールには IAM ロールを指定します。ロールを見つけるには、「kendra」という単語を含むアカウントでロールから選択するか、別のロールの名前を入力します。ロールが必要とするアクセス許可の詳細については、「IAM roles for indexes」を参照してください。
5. [次へ] を選択します。
6. [ユーザーアクセスコントロールの設定] ページで、[次へ] をクリックします。インデックスを作成した後、アクセス制御にトークンを使用するようにインデックスを更新できます。詳細については、「Controlling access to documents」を参照してください。
7. [プロビジョニングの詳細] ページで、[作成] を選択します。
8. インデックスが作成されるまでにしばらく時間がかかることがあります。インデックスのリストをチェックして、インデックスの作成の進行状況を確認します。インデックスのステータスが ACTIVE の場合、インデックスが使用する準備ができています。

AWS CLI

インデックスを作成するには

1. 以下のコマンドを使用してインデックスを作成します。は、 Amazon Kendra アクションを実行できる IAM ロールの Amazon リソースネーム (ARN) `role-arn`である必要があります。詳細については、「IAM roles」を参照してください。

   次のコマンドは、Linux と macOS 用にフォーマットされています。Windows を使用している場合、Unix 行連結記号 (\) をキャレット (^) に置き換えます。

   ```
   aws kendra create-index \
    --name index name \
    --description "index description" \
    --role-arn arn:aws:iam::account ID:role/role name
   ```

2. インデックスが作成されるまでにしばらく時間がかかることがあります。インデックスの状態をチェックするには、以下のコマンドで `create-index` によって返されるインデックス ID を使用します。インデックスのステータスが `ACTIVE` の場合、インデックスが使用する準備ができています。

   ```
   aws kendra describe-index \
    --index-id index ID
   ```

Python

インデックスを作成するには

- 次のコード例で、次の変数の値を指定します。
  - `description` - 作成するインデックスの説明。これはオプションです。
  - `index_name` - 作成するインデックスの名前。
  - `role_arn`— Amazon Kendra APIs を実行できるロールの Amazon リソースネーム (ARN)。詳細については、「IAM roles」を参照してください。

  ```
  import boto3
  from botocore.exceptions import ClientError
  import pprint
  ```

```
import time

kendra = boto3.client("kendra")

print("Create an index.")

# Provide a name for the index
index_name = "index-name"
# Provide an optional description for the index
description = "index description"
# Provide the IAM role ARN required for indexes
role_arn = "arn:aws:iam::${account id}:role/${role name}"

try:
    index_response = kendra.create_index(
        Name = index_name,
        Description = description,
        RoleArn = role_arn
    )

    pprint.pprint(index_response)

    index_id = index_response["Id"]

    print("Wait for Amazon Kendra to create the index.")

    while True:
        # Get the details of the index, such as the status
        index_description = kendra.describe_index(
            Id = index_id
        )
        # If status is not CREATING, then quit
        status = index_description["Status"]
        print(" Creating index. Status: "+status)
        if status != "CREATING":
            break
        time.sleep(60)

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

インデックスを作成するには

- 次のコード例で、次の変数の値を指定します。
  - description - 作成するインデックスの説明。これはオプションです。
  - index_name - 作成するインデックスの名前。
  - role_arn— Amazon Kendra APIs を実行できるロールの Amazon リソースネーム (ARN)。詳細については、「IAM roles」を参照してください。

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateIndexRequest;
import software.amazon.awssdk.services.kendra.model.CreateIndexResponse;
import software.amazon.awssdk.services.kendra.model.DescribeIndexRequest;
import software.amazon.awssdk.services.kendra.model.DescribeIndexResponse;
import software.amazon.awssdk.services.kendra.model.IndexStatus;


public class CreateIndexExample {

    public static void main(String[] args) throws InterruptedException {

        String indexDescription = "Getting started index for Kendra";
        String indexName = "java-getting-started-index";
        String indexRoleArn = "arn:aws:iam::<your AWS account ID>:role/
KendraRoleForGettingStartedIndex";

        System.out.println(String.format("Creating an index named %s",
 indexName));
        CreateIndexRequest createIndexRequest = CreateIndexRequest
            .builder()
            .description(indexDescription)
            .name(indexName)
            .roleArn(indexRoleArn)
            .build();
        KendraClient kendra = KendraClient.builder().build();
```

```
        CreateIndexResponse createIndexResponse =
 kendra.createIndex(createIndexRequest);
        System.out.println(String.format("Index response %s",
 createIndexResponse));

        String indexId = createIndexResponse.id();

        System.out.println(String.format("Waiting until the index with ID %s is
 created.", indexId));
        while (true) {
            DescribeIndexRequest describeIndexRequest =
 DescribeIndexRequest.builder().id(indexId).build();
            DescribeIndexResponse describeIndexResponse =
 kendra.describeIndex(describeIndexRequest);
            IndexStatus status = describeIndexResponse.status();
            if (status != IndexStatus.CREATING) {
                break;
            }

            TimeUnit.SECONDS.sleep(60);
        }

        System.out.println("Index creation is complete.");
    }
}
```

インデックスを作成したら、そのインデックスにドキュメントを追加します。これらは直接追加することも、定期的にインデックスを更新するデータソースを作成することもできます。

トピック

- バッチアップロードを使用したドキュメントのインデックスへの直接追加
- よくある質問 (FAQ) のインデックスへの追加
- カスタムドキュメントフィールドの作成
- トークンによるドキュメントへのアクセスの制御

バッチアップロードを使用したドキュメントのインデックスへの直接追加

BatchPutDocument API を使用して、ドキュメントをインデックスに直接追加できます。コンソールを使用してドキュメントを直接追加することはできません。コンソールを使用する場合、データソースに接続して、ドキュメントをインデックスに追加します。ドキュメントは S3 バケットから追加することも、バイナリデータとして指定することもできます。でサポートされているドキュメントタイプのリストについては、「ドキュメントのタイプ Amazon Kendra 」を参照してください。

BatchPutDocument を使用したインデックスへのドキュメントの追加は、非同期演算です。BatchPutDocument API を呼び出した後、 BatchGetDocumentStatus API を使用してドキュメントのインデックス作成の進行状況をモニタリングします。ドキュメント ID のリストで BatchGetDocumentStatus API を呼び出すと、ドキュメントのステータスが返されます。ドキュメントのステータスが INDEXED または FAILED の場合、ドキュメントの処理は完了しています。ステータスが FAILED の場合、BatchGetDocumentStatus API は、ドキュメントにインデックス作成できなかった理由を返します。

ドキュメント取り込みプロセス中にコンテンツやドキュメントメタデータのフィールドや属性を変更する場合は、「Amazon Kendra Custom Document Enrichment」を参照してください。カスタムデータソースを使用する場合、BatchPutDocument API を使用して送信する各ドキュメントには、属性またはフィールドとしてデータソース ID と実行 ID が必要です。詳細については、「Required attributes for custom data sources」を参照してください。

> **Note**
>
> 各ドキュメント ID は、インデックスごとに一意である必要があります。一意の ID でドキュメントにインデックスを付けるデータソースを作成してから、BatchPutDocument API を使用して同じドキュメントにインデックスを付けることはできません。その逆も同様です。データソースを削除してから BatchPutDocument API を使用して同じドキュメントにインデックスを付けることができます。その逆も可能です。BatchPutDocument と BatchDeleteDocument API を同じドキュメントセットの Amazon Kendra データソースコネクタと組み合わせて使用すると、データに不整合が生じる可能性があります。代わりに、Amazon Kendra カスタムデータソースコネクタの使用をお勧めします。

次のデベロッパーガイドドキュメントでは、ドキュメントをインデックスに直接追加する方法を示します。

トピック

API を使用した BatchPutDocumentドキュメントの追加

次の例では、 を呼び出してテキストの BLOB をインデックスに追加しますBatchPutDocument。BatchPutDocument API を使用して、インデックスにドキュメントを直接追加できます。でサポートされているドキュメントタイプのリストについては、「ドキュメントのタイプ Amazon Kendra 」を参照してください。

AWS CLI および SDKs「インデックスの作成」を参照してください。CLI と SDK をセットアップするには、「Setting up Amazon Kendra」を参照してください。

> **Note**
>
> インデックスに追加されるファイルは、UTF-8 でエンコードされたバイトストリームに存在する必要があります。

次の例では、UTF-8 でエンコードされたテキストをインデックスに追加します。

CLI

で AWS Command Line Interface、次のコマンドを使用します。次のコマンドは、Linux と macOS 用にフォーマットされています。Windows を使用している場合、Unix 行連結記号 (\) をキャレット (^) に置き換えます。

```
aws kendra batch-put-document \
   --index-id index-id \
   --documents '{"Id":"doc-id-1", "Blob":"Amazon.com is an online retailer.",
 "ContentType":"PLAIN_TEXT", "Title":"Information about Amazon.com"}'
```

Python

```
import boto3

kendra = boto3.client("kendra")
```

```
# Provide the index ID
index_id = "index-id"

# Provide the title and text
title = "Information about Amazon.com"
text = "Amazon.com is an online retailer."

document = {
    "Id": "1",
    "Blob": text,
    "ContentType": "PLAIN_TEXT",
    "Title": title
}

documents = [
    document
]

result = kendra.batch_put_document(
    IndexId = index_id,
    Documents = documents
)

print(result)
```

Java

```
package com.amazonaws.kendra;


import software.amazon.awssdk.core.SdkBytes;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.BatchPutDocumentRequest;
import software.amazon.awssdk.services.kendra.model.BatchPutDocumentResponse;
import software.amazon.awssdk.services.kendra.model.ContentType;
import software.amazon.awssdk.services.kendra.model.Document;

public class AddDocumentsViaAPIExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String indexId = "yourIndexId";
```

```
        Document testDoc = Document
            .builder()
            .title("The title of your document")
            .id("a_doc_id")
            .blob(SdkBytes.fromUtf8String("your text content"))
            .contentType(ContentType.PLAIN_TEXT)
            .build();

        BatchPutDocumentRequest batchPutDocumentRequest = BatchPutDocumentRequest
            .builder()
            .indexId(indexId)
            .documents(testDoc)
            .build();

        BatchPutDocumentResponse result =
 kendra.batchPutDocument(batchPutDocumentRequest);

        System.out.println(String.format("BatchPutDocument Result: %s", result));
    }
}
```

S3 バケットからのドキュメントの追加

BatchPutDocument API を使用して、 Amazon S3 バケットからインデックスに直接ドキュメントを追加できます。同じコールで最大 10 個のドキュメントを追加できます。S3 バケットを使用する場合は、 IAM ロールにドキュメントを含むバケットへのアクセス許可を付与する必要があります。`RoleArn` パラメータでロールを指定します。

BatchPutDocument API を使用して Amazon S3 バケットからドキュメントを追加することは、1 回限りのオペレーションです。インデックスをバケットのコンテンツと同期させるには、 Amazon S3 データソースを作成します。詳細については、「Amazon S3 data source」を参照してください。

AWS CLI および SDKs「インデックスの作成」を参照してください。CLI と SDK をセットアップするには、「Setting up Amazon Kendra」を参照してください。S3 バケットの作成については、Amazon Simple Storage Service ドキュメントを参照してください。

次の使用例は、`BatchPutDocument` API を使用して、インデックスに 2 つの Microsoft Word ドキュメントを追加します。

Python

```
import boto3

kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the IAM role ARN required to index documents in an S3 bucket
role_arn = "arn:aws:iam::${acccountID}:policy/${roleName}"

doc1_s3_file_data = {
    "Bucket": "bucket-name",
    "Key": "document1.docx"
}

doc1_document = {
    "S3Path": doc1_s3_file_data,
    "Title": "Document 1 title",
    "Id": "doc_1"
}

doc2_s3_file_data = {
    "Bucket": "bucket-name",
    "Key": "document2.docx"
}

doc2_document = {
    "S3Path": doc2_s3_file_data,
    "Title": "Document 2 title",
    "Id": "doc_2"
}

documents = [
    doc1_document,
    doc2_document
]

result = kendra.batch_put_document(
    Documents = documents,
    IndexId = index_id,
    RoleArn = role_arn
)
```

```
print(result)
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.BatchPutDocumentRequest;
import software.amazon.awssdk.services.kendra.model.BatchPutDocumentResponse;
import software.amazon.awssdk.services.kendra.model.Document;
import software.amazon.awssdk.services.kendra.model.S3Path;

public class AddFilesFromS3Example {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String indexId = "yourIndexId";
        String roleArn = "yourIndexRoleArn";

        Document pollyDoc = Document
            .builder()
            .s3Path(
                S3Path.builder()
                .bucket("an-aws-kendra-test-bucket")
                .key("What is Amazon Polly.docx")
                .build())
            .title("What is Amazon Polly")
            .id("polly_doc_1")
            .build();

        Document rekognitionDoc = Document
            .builder()
            .s3Path(
                S3Path.builder()
                .bucket("an-aws-kendra-test-bucket")
                .key("What is Amazon Rekognition.docx")
                .build())
            .title("What is Amazon rekognition")
            .id("rekognition_doc_1")
            .build();

        BatchPutDocumentRequest batchPutDocumentRequest = BatchPutDocumentRequest
            .builder()
```

```
            .indexId(indexId)
            .roleArn(roleArn)
            .documents(pollyDoc, rekognitionDoc)
            .build();

        BatchPutDocumentResponse result =
 kendra.batchPutDocument(batchPutDocumentRequest);

        System.out.println(String.format("BatchPutDocument result: %s", result));
    }
}
```

よくある質問 (FAQ) のインデックスへの追加

コンソールまたは CreateFaq API を使用して、よくある質問 (FAQsをインデックスに直接追加できます。インデックスへのよくある質問の追加は、非同期演算です。よくある質問のデータは、Amazon Simple Storage Service バケットに保存するファイルに入れます。CSV ファイルまたは JSON ファイルをよくある質問の入力として使用できます。

- 基本 CSV — 各行に質問、回答、オプションのソース URI が含まれる CSV ファイル。
- カスタム CSV - 質問、回答、およびよくある質問のレスポンスのファセット、表示、またはソートに使用できるカスタムフィールド/属性のヘッダーを含む CSV ファイル。アクセスコントロールフィールドを定義して、よくある質問のレスポンスを、よくある質問のレスポンスを許可されている特定のユーザーおよびグループに制限することもできます。
- JSON - 質問、回答、およびよくある質問のレスポンスのファセット、表示、またはソートに使用できるカスタムフィールド/属性を含む JSON ファイル。アクセスコントロールフィールドを定義して、よくある質問のレスポンスを、よくある質問のレスポンスを許可されている特定のユーザーおよびグループに制限することもできます。

例えば、以下は、米国ワシントン州スポケーンと米国ミズーリ州マウンテンビューにある無料診療所に関する質問に対する回答を提供する基本 CSV ファイルです。

```
How many free clinics are in Spokane WA?, 13
How many free clinics are there in Mountain View Missouri?, 7
```

> **Note**
>
> FAQ ファイルは UTF-8 でエンコードされたファイルである必要があります。

トピック

よくある質問ファイルのインデックスフィールドの作成

入力にカスタム CSV または JSON ファイルを使用する場合は、よくある質問の質問のカスタムフィールドを宣言できます。例えば、各よくある質問の質問にビジネス部門を割り当てるカスタムフィールドを作成できます。よくある質問がレスポンスで返されたら、部門をファセットとして使用して、例えば「HR」や「Finance」のみに検索を絞り込めます。

カスタムフィールドは、インデックスフィールドにマッピングする必要があります。コンソールでは、[ファセットの定義] ページを使用して、インデックスフィールドを作成します。API を使用する場合は、まず UpdateIndex API を使用してインデックスフィールドを作成する必要があります。

よくある質問ファイルのフィールド/属性タイプは、関連するインデックスフィールドのタイプと一致する必要があります。例えば、「Department」フィールドは `STRING_LIST` タイプフィールドです。そのため、よくある質問ファイルでは、部門フィールドの値を、文字列リストとして指定する必要があります。インデックスフィールドのタイプは、コンソールのファセット定義ページまたは DescribeIndex API を使用して確認できます。

カスタム属性にマッピングするインデックスフィールドを作成する場合、その属性を表示可能、ファセット可能、またはソート可能にマークできます。カスタム属性を検索可能にすることはできません。

カスタム属性に加えて、カスタム CSV または JSON ファイル内の Amazon Kendra 予約フィールドや共通フィールドを使用することもできます。詳細については、「Document attributes or fields」を参照してください。

基本 CSV ファイル

よくある質問に単純な構造を使用する場合は、基本 CSV ファイルを使用します。基本 CSV ファイルには、各行に、質問、回答、および詳細情報を含むドキュメントを参照するオプションのソース URI の 2 つまたは 3 つのフィールドがあります。

ファイルの内容は、カンマ区切り値 (CSV) ファイルの RFC 4180 共通形式と MIME タイプに従う必要があります。

以下は、基本 CSV 形式のよくある質問ファイルです。

```
How many free clinics are in Spokane WA?, 13, https://s3.region.company.com/bucket-
name/directory/faq.csv
How many free clinics are there in Mountain View Missouri?, 7, https://
s3.region.company.com/bucket-name/directory/faq.csv
```

カスタム CSV ファイル

よくある質問の質問にカスタムフィールド/属性を追加する場合は、カスタム CSV ファイルを使用します。カスタム CSV ファイルでは、CSV ファイルのヘッダー行を使用して、追加の属性を定義します。

CSV ファイルには、2 つの必須フィールドが含まれている必要があります。

- `_question` - よくある質問
- `_answer` - よくある質問への回答

ファイルには、 Amazon Kendra 予約済みフィールドとカスタムフィールドの両方を含めることができます。次はカスタム CSV ファイルの例です。

```
_question,_answer,_last_updated_at,custom_string
How many free clinics are in Spokane WA?, 13, 2012-03-25T12:30:10+01:00, Note: Some
 free clinics require you to meet certain criteria in order to use their services
How many free clinics are there in Mountain View Missouri?, 7,
 2012-03-25T12:30:10+01:00, Note: Some free clinics require you to meet certain
 criteria in order to use their services
```

カスタムファイルの内容は、カンマ区切り値 (CSV) ファイルの RFC 4180 共通形式と MIME タイプに従う必要があります。

カスタムフィールドの種類を以下に示します。

- 日付 — ISO 8601 でエンコードされた日付と時刻の値。

  例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- 長整数 — 1234 などの数字。
- 文字列 — 文字列値。文字列にカンマが含まれている場合は、値全体を二重引用符 (") で囲みます (例えば、"custom attribute, and more")。
- 文字列リスト - 文字列値のリスト。値を引用符 (") で囲んだカンマ区切りリストの数値を一覧表示します (例えば、"item1, item2, item3")。リストにエントリが 1 つしか含まれていない場合は、引用符を省略できます (例えば、item1)。

カスタム CSV ファイルにはユーザーアクセスコントロールフィールドを含められます。これらのフィールドを使用して、よくある質問へのアクセスを特定のユーザーおよびグループに制限できます。ユーザーコンテキストでフィルタリングするには、ユーザーはクエリでユーザーおよびグループ情報を提供する必要があります。それ以外の場合は、関連するよくある質問がすべて返されます。詳細については、「User context filtering」を参照してください。

よくある質問のユーザーコンテキストフィルターを以下に示します。

- _acl_user_allow — 許可リストのユーザーは、クエリレスポンスでよくある質問を確認できます。よくある質問は他のユーザーには返されません。
- _acl_user_deny — 拒否リストのユーザーは、クエリレスポンスでよくある質問を確認できません。よくある質問は、クエリに関連する場合に、他のすべてのユーザーに返されます。
- _acl_group_allow — 許可されたグループのメンバーであるユーザーは、クエリレスポンスでよくある質問を確認できます。よくある質問は、別のグループのメンバーであるユーザーには返されません。
- _acl_group_deny — 拒否されたグループのメンバーであるユーザーは、クエリレスポンスでよくある質問を確認できません。よくある質問は、クエリに関連する場合に、他のグループに返されます。

許可リストと拒否リストの値を、引用符で囲んだコンマ区切りのリストで指定します (例えば、"user1,user2,user3")。許可リストまたは拒否リストのいずれかにユーザーまたはグループを含めることはできますが、同一ユーザーが個別に許可されているもののグループでは拒否されてい

るような場合は、両方に含めることはできません。両方にユーザーまたはグループを含めると、エラーが発生します。

次はユーザーコンテキスト情報を含むカスタム CSV ファイルの例です。

```
_question, _answer, _acl_user_allow, _acl_user_deny, _acl_group_allow, _acl_group_deny
How many free clinics are in Spokane WA?, 13, "userID6201,userID7552",
 "userID1001,userID2020", groupBasicPlusRate, groupPremiumRate
```

JSON ファイル

JSON ファイルを使用して、インデックスの質問、回答、およびフィールドを指定できます。よくある質問には、 Amazon Kendra 任意の予約済みフィールドまたはカスタムフィールドを追加できます。

以下に、JSON のスキーマを示します。

```
{
    "SchemaVersion": 1,
    "FaqDocuments": [
        {
            "Question": string,
            "Answer": string,
            "Attributes": {
                string: object
                additional attributes
            },
            "AccessControlList": [
               {
                   "Name": string,
                   "Type": enum( "GROUP" | "USER" ),
                   "Access": enum( "ALLOW" | "DENY" )
               },
               additional user context
            ]
        },
        additional FAQ documents
    ]
}
```

次の JSON ファイルの例は、2 つのよくある質問ドキュメントを示しています。1 つのドキュメントには、必要な質問と回答のみが含まれています。もう 1 つのドキュメントには、追加のフィールド情報、ユーザーコンテキスト、またはアクセスコントロール情報も含まれています。

```
{
    "SchemaVersion": 1,
    "FaqDocuments": [
        {
            "Question": "How many free clinics are in Spokane WA?",
            "Answer": "13"
        },
        {
            "Question": "How many free clinics are there in Mountain View Missouri?",
            "Answer": "7",
            "Attributes": {
                "_source_uri": "https://s3.region.company.com/bucket-name/directory/
faq.csv",
                "_category": "Charitable Clinics"
            },
            "AccessControlList": [
               {
                   "Name": "user@amazon.com",
                   "Type": "USER",
                   "Access": "ALLOW"
               },
               {
                   "Name": "Admin",
                   "Type": "GROUP",
                   "Access": "ALLOW"
               }
            ]
        }
    ]
}
```

カスタムフィールドの種類を以下に示します。

- 日付 - ISO 8601 でエンコードされた日付と時刻値を持つ JSON 文字列値。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- 長整数 — 1234 などの JSON 数値。

- 文字列 - JSON 文字列値 (例えば、"custom attribute")。
- 文字列リスト - 文字列値の JSON 配列 (例えば、["item1,item2,item3"])。

JSON ファイルには、ユーザーアクセスコントロールフィールドを含めることができます。これらのフィールドを使用して、よくある質問へのアクセスを特定のユーザーおよびグループに制限できます。ユーザーコンテキストでフィルタリングするには、ユーザーはクエリでユーザーおよびグループ情報を提供する必要があります。それ以外の場合は、関連するよくある質問がすべて返されます。詳細については、「User context filtering」を参照してください。

許可リストまたは拒否リストのいずれかにユーザーまたはグループを含めることはできますが、同一ユーザーが個別に許可されているもののグループでは拒否されているような場合は、両方に含めることはできません。両方にユーザーまたはグループを含めると、エラーが発生します。

以下は、JSON のよくある質問に、ユーザーアクセスコントロールを含める例です。

```
"AccessControlList": [
                {
                    "Name": "group or user name",
                    "Type": "GROUP | USER",
                    "Access": "ALLOW | DENY"
                },
                additional user context
            ]
```

よくある質問ファイルの使用

よくある質問入力ファイルを S3 バケットに保存した後、コンソールまたは CreateFaq API を使用して、質問と回答をインデックスに入れます。よくある質問を更新する場合は、よくある質問を削除してもう一度作成します。DeleteFaq API を使用してよくある質問を削除します。

ソースファイルを含む S3 バケットにアクセスできる IAM ロールを指定する必要があります。ロールは、コンソールまたは RoleArn パラメータで指定します。次は、よくある質問ファイルをインデックスに追加する例です。

Python

```
import boto3
```

```
kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the IAM role ARN required to index documents in an S3 bucket
role_arn = "arn:aws:iam::${accountId}:role/${roleName}"

# Provide the S3 bucket path information to the FAQ file
faq_path = {
    "Bucket": "bucket-name",
    "Key": "FreeClinicsUSA.csv"
}

response = kendra.create_faq(
    S3Path =  faq_path,
    Name = "FreeClinicsUSA",
    IndexId = index_id,
    RoleArn = role_arn
)

print(response)
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateFaqRequest;
import software.amazon.awssdk.services.kendra.model.CreateFaqResponse;
import software.amazon.awssdk.services.kendra.model.S3Path;

public class AddFaqExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String indexId = "yourIndexId";
        String roleArn = "your role for accessing S3 files";

        CreateFaqRequest createFaqRequest = CreateFaqRequest
            .builder()
            .indexId(indexId)
            .name("FreeClinicsUSA")
            .roleArn(roleArn)
```

```
            .s3Path(
                S3Path
                    .builder()
                    .bucket("an-aws-kendra-test-bucket")
                    .key("FreeClinicsUSA.csv")
                    .build())
            .build();

        CreateFaqResponse response = kendra.createFaq(createFaqRequest);

        System.out.println(String.format("The result of creating FAQ: %s",
 response));

    }
}
```

英語以外の言語のよくある質問ファイル

言語を指定しない場合、サポートされている language. Amazon Kendra indexes のFAQsは、デフォルトで英語でインデックス作成できます。CreateFaq オペレーションを呼び出すときに言語コードを指定するか、よくある質問メタデータによくある質問の言語コードをフィールドとして含めることができます。よくある質問のメタデータフィールドで指定されたメタデータに言語コードがない場合、よくある質問は、CreateFAQ 演算を呼び出したときに指定した言語コードを使用してインデックス作成されます。コンソールでサポートされている言語でよくある質問ドキュメントのインデックス作成をするには、[FAQs] (よくある質問) い n 移動し、[Add FAQ] (よくある質問の追加) を選択します。[言語] のドロップダウンから言語を選択します。

カスタムドキュメントフィールドの作成

Amazon Kendra インデックスでドキュメントのカスタム属性またはフィールドを作成できます。例えば、「HR」、「Sales」、「Manufacturing」という値を持つ「Department」というカスタムフィールドまたは属性を作成できます。これらのカスタムフィールドまたは属性を Amazon Kendra インデックスにマッピングする場合、それらを使用して検索結果をフィルタリングし、「HR」部門属性でドキュメントを含めることができます。

カスタムフィールドまたは属性を使用するには、まずインデックスにフィールドを作成する必要があります。コンソールを使用してデータソースフィールドマッピングを編集してカスタムフィールドを追加するか、 UpdateIndex API を使用してインデックスフィールドを作成します。フィールドを作成すると、フィールドデータ型を変更することはできません。

ほとんどのデータソースの場合、外部データソースのフィールドを Amazon Kendra の対応するフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。S3 データソースでは、JSON メタデータファイルを使用してカスタムフィールドまたは属性を作成できます。

最大 500 のカスタムフィールドまたは属性を作成できます。

Amazon Kendra 予約済みフィールドまたは共通フィールドを使用することもできます。詳細については、「Document attributes or fields」を参照してください。

トピック

- カスタムドキュメントフィールドの更新

カスタムドキュメントフィールドの更新

UpdateIndex API では、DocumentMetadataConfigurationUpdates パラメータを使用してカスタムフィールドまたは属性を追加します

次の JSON の例では、DocumentMetadataConfigurationUpdates を使用して「Department」というフィールドをインデックスに追加します。

```
"DocumentmetadataConfigurationUpdates": [
   {
       "Name": "Department",
       "Type": "STRING_VALUE"
   }
]
```

以下のセクションでは、Amazon S3 データソースの BatchPutDocumentおよび を使用してカスタム属性またはフィールドを追加する例を示します。

トピック

- BatchPutDocument API を使用したカスタム属性またはフィールドの追加
- Amazon S3 データソースへのカスタム属性またはフィールドの追加

BatchPutDocument API を使用したカスタム属性またはフィールドの追加

BatchPutDocument API を使用してドキュメントをインデックスに追加するときは、 の一部としてカスタムフィールドまたは属性を指定しますAttributes。API を呼び出すと、複数のフィールドま

たは属性を追加できます。最大 500 のカスタムフィールドまたは属性を作成できます。次の例は、ドキュメントに「Department」を追加するカスタムフィールドまたは属性です。

```
"Attributes":
    {
        "Department": "HR",
        "_category": "Vacation policy"
    }
```

Amazon S3 データソースへのカスタム属性またはフィールドの追加

S3 バケットをインデックスのデータソースとして使用する場合は、コンパニオンメタデータファイルを使用してドキュメントにメタデータを追加します。メタデータ JSON ファイルは、ドキュメントと平行なディレクトリ構造に配置します。詳細については、「S3 document metadata」を参照してください。

カスタムフィールドまたは属性は、`Attributes` JSON 構造で指定します。最大 500 のカスタムフィールドまたは属性を作成できます。例えば、以下の例では `Attributes` を使用して 3 つのカスタムフィールドまたは属性と 1 つの予約フィールドを定義しています。

```
"Attributes": {
        "brand": "Amazon Basics",
        "price": 1595,
        "_category": "sports",
        "subcategories": ["outdoors", "electronics"]
    }
```

次のステップでは、Amazon S3 データソースにカスタム属性を追加する方法について説明します。

トピック

- ステップ 1: Amazon Kendra インデックスを作成する
- ステップ 2: インデックスを更新してカスタムドキュメントフィールドを追加する
- ステップ 3: Amazon S3 データソースを作成し、データソースフィールドをカスタム属性にマッピングする

ステップ 1: Amazon Kendra インデックスを作成する

の手順に従ってインデックスの作成、Amazon Kendra インデックスを作成します。

ステップ 2: インデックスを更新してカスタムドキュメントフィールドを追加する

インデックスを作成したら、そのインデックスにフィールドを追加します。次の手順は、コンソールと CLI を使用してインデックスにフィールドを追加する方法を示しています。

Console

インデックスフィールドを作成するには

1. インデックス が作成されていることを確認します。
2. 次に、左側のナビゲーションメニューから、データ管理 からファセット定義 を選択します。
3. インデックスフィールド設定ガイド で、インデックスフィールド から、フィールドの追加 を選択してカスタムフィールドを追加します。
4. [Add index field] (インデックスフィールドの追加) ダイアログボックスで、以下の操作を行います。

   - フィールド名 – フィールド名を追加します。
   - データ型 - 文字列 、文字列リスト 、日付 のいずれかのデータ型を選択します。
   - 使用タイプ - Facetable 、検索可能、表示可能 、ソート可能 のいずれを使用するかを選択します。

   次に、追加を選択します。

マッピングする他のフィールドに対して、最後のステップを繰り返します。

CLI

```
aws kendra update-index  \
--region $region \
--endpoint-url $endpoint \
--application-id $applicationId \
--index-id $indexId  \
--document-metadata-configuration-updates \
"[
    {
        "Name": "string",
        "Type": "STRING_VALUE"|"STRING_LIST_VALUE"|"LONG_VALUE"|"DATE_VALUE",
        "Relevance": {
```

```
            "Freshness": true|false,
            "Importance": integer,
            "Duration": "string",
            "RankOrder": "ASCENDING"|"DESCENDING",
            "ValueImportanceMap": {"string": integer
            ...}
    },
    "Search": {
        "Facetable": true|false,
        "Searchable": true|false,
        "Displayable": true|false,
        "Sortable": true|false
        }
    }
...
]"
```

ステップ 3: Amazon S3 データソースを作成し、データソースフィールドをカスタム属性にマッピングする

Amazon S3 データソースを作成し、フィールドにフィールドをマッピングするには、「」の手順に従いますAmazon S3。

API を使用している場合は、 CreateDataSource API を使用するconfigurationときに の fieldMappings 属性を使用します。

データソースフィールドのマッピング方法の概要については、「」を参照してくださいデータソースフィールドのマッピング。

トークンによるドキュメントへのアクセスの制御

インデックス内の特定のドキュメントにアクセスしたり検索結果に特定のドキュメントを表示したりできるユーザーまたはグループを制御できます。これはユーザーコンテキストのフィルタリングと呼ばれます。ドキュメントへのアクセスをコントロールできるという利点を持つ、パーソナライズされた検索の一種です。例えば、企業ポータルで情報を検索するすべてのチームが会社の極秘文書にアクセスする必要があるわけではなく、これらの文書がすべてのユーザーに関連しているわけでもありません。極秘文書へのアクセス許可を与えられた特定のユーザーまたはチームグループのみが、検索結果でこれらの文書を参照できます。

Amazon Kendra は、次のトークンタイプを使用したトークンベースのユーザーアクセス制御をサポートしています。

- オープン ID
- 共有シークレットを持つ JWT
- パブリックキーを持つ JWT
- JSON

Amazon Kendra は、検索アプリケーションに対して非常に安全なエンタープライズ検索を提供します。検索結果には、組織のセキュリティモデルが反映されます。お客様は、ユーザーの検索アプリケーションへのアクセスを認証し、認可する責任を負うものとします。検索時に、Amazon Kendra サービスは、お客様の検索アプリケーションによって提供されたユーザー ID と、クロール/インデックス作成時に Amazon Kendra コネクタによって収集されたアクセスコントロールリスト (ACL) をドキュメントに基づいて検索結果をフィルタリングします。検索結果は、元のドキュメントリポジトリを参照する URL と短い抜粋を返します。ドキュメント全体へのアクセスは、元のリポジトリによって引き続き適用されます。

トピック

OpenID の使用

アクセス制御に OpenID Amazon Kendra トークンを使用するようにインデックスを設定するには、OpenID プロバイダーの JWKS (JSON Web キーセット) URL が必要です。ほとんどの場合、JWKS URL は `https://domain-name/.well_known/jwks.json` の形式になります (OpenID ディスカバリに従っている場合)。

次の例は、インデックスの作成時にユーザーアクセスコントロールに OpenID トークンを使用する方法を示しています。

Console

1. [Create index] (インデックスの作成) を選択して、新しいインデックスの作成を開始します。

2. [Specify index details] (インデックスの詳細の指定) ページで、インデックスに名前と説明を付けます。
3. [IAM ロール] には、[ロール] を選択するか、または [新規ロールを作成] を選択し、新しいロールを作成してロール名を指定します。IAM ロールには「-」というプレフィックスが付きます。AmazonKendra
4. その他のフィールドはすべてデフォルトのままにしておきます。[次へ] を選択します。
5. [Configure user access control] (ユーザーアクセスコントロールの設定) ページの、[Access control settings] (アクセスコントロールの設定) で、[Yes] (はい) を選択し、アクセス制御にトークンを使用します。
6. [Token configuration] (トークンの設定) を選択し、[Token type] (トークンタイプ) は [OpenID] を選択します。
7. [Signing key URL] (署名キー URL) を指定します。URL は JSON ウェブキーのセットを参照する必要があります。
8. [Advanced configuration] (詳細設定) の [Optional] (オプション):
    a. [Username] (ユーザーネーム) を指定して ACL チェックで使用します。
    b. 1 つ以上の [Groups] (グループ) を指定して、ACL チェックで使用します。
    c. トークン発行者を検証する [Issuer] (発行者) を指定します。
    d. [Client Id(s)] (クライアント ID) を指定します。JWT のオーディエンスと一致する正規表現を指定する必要があります。
9. [Provisioning details] (プロビジョニングの詳細) ページで、[Developer edition] (デベロッパーエディション) を選択します。
10. [Create] (作成) を選択してインデックスを作成します。
11. インデックスが作成されるまでお待ちください。 Amazon Kendra インデックスのハードウェアをプロビジョニングします。この演算には時間がかかる場合があります。

CLI

JSON AWS CLI 入力ファイルを使用してを使用してインデックスを作成するには、まず必要なパラメータを含む JSON ファイルを作成します。

```
{
    "Name": "user-context",
    "Edition": "ENTERPRISE_EDITION",
    "RoleArn": "arn:aws:iam::account-id:role:/my-role",
```

```
    "UserTokenConfigurations": [
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "URL",
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "URL": "https://example.com/.well-known/jwks.json"
            }
        }
    ],
    "UserContextPolicy": "USER_TOKEN"
}
```

デフォルトのユーザーフィールド名とグループフィールド名を上書きできます。UserNameAttributeField のデフォルト値は「ユーザー」です。GroupAttributeField のデフォルト値は「グループ」です。

次に、入力ファイルを使用して、create-index を呼び出します。例えば、JSON ファイルの名前が create-index-openid.json の場合、以下を使用できます。

```
aws kendra create-index --cli-input-json file://create-index-openid.json
```

Python

```
response = kendra.create_index(
    Name='user-context',
    Edition='ENTERPRISE_EDITION',
    RoleArn='arn:aws:iam::account-id:role:/my-role',
    UserTokenConfigurations=[
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "URL",
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "URL": "https://example.com/.well-known/jwks.json"
            }
        }
    ],
```

```
    UserContextPolicy='USER_TOKEN'
)
```

共有シークレットで JSON ウェブトークン (JWT) を使用する

次の例は、インデックスを作成するときに JSON Web Token (JWT) を共有シークレットトークンとともに使用してユーザーアクセスを制御する方法を示しています。

Console

1. [Create index] (インデックスの作成) を選択して、新しいインデックスの作成を開始します。
2. [Specify index details] (インデックスの詳細の指定) ページで、インデックスに名前と説明を付けます。
3. [IAM role] (IAM ロール) には、ロールを選択するか、または [Create a new role] (新規ロールの作成) を選択し、新しいロールを作成してロール名を指定します。 IAM ロールには「AmazonKendra-」というプレフィックスが付きます。
4. その他のフィールドはすべてデフォルトのままにしておきます。[次へ] を選択します。
5. [Configure user access control] (ユーザーアクセスコントロールの設定) ページの、[Access control settings] (アクセスコントロールの設定) で、[Yes] (はい) を選択し、アクセス制御にトークンを使用します。
6. [Token configuration] (トークンの設定) で、[JWT with shared secret] (共有シークレットを使用したJWT) を [Token type] (トークンタイプ) として選択します。
7. [共有シークレットに署名するためのパラメータ] で、[シークレットのタイプ] を選択します。既存の AWS Secrets Manager 共有シークレット、または新しい共有シークレットを使用できます。

   新しい共有シークレットを作成するには、[New] (新規) を選択し、次に、以下のステップを実行します。

   a. 「 AWS Secrets Manager 新規シークレット」で、シークレット名を指定します。プレフィックス `AmazonKendra-` は、パブリックキーを保存すると追加されます。
   b. [Key ID] (キー ID) を指定します。キー ID は、トークンの JSON ウェブ署名をセキュア化するために使用されたキーを示すヒントです。
   c. トークンに署名 [Algorithm] (アルゴリズム) を選択します。これは、ID トークンの保護に使用される暗号化アルゴリズムです。RSA の詳細については、RSA Cryptography を参照してください。

d. base64 URL でエンコードされたシークレットを入力して、[共有シークレット] を指定します。また、[シークレットの生成] を選択して、自分のシークレットを生成できます。シークレットが base64 URL でエンコードされていることを確認する必要があります。

e. (オプション) 共有シークレットが有効になるタイミングを指定します。シークレットの有効開始日、有効期限、またはその両方を指定できます。シークレットは、指定された時間間隔の間、有効です。

f. [Save secret] (シークレットの保存) を選択して新しいシークレットを保存します。

8. (オプション) [詳細設定]:

a. [Username] (ユーザーネーム) を指定して ACL チェックで使用します。

b. 1 つ以上の [Groups] (グループ) を指定して、ACL チェックで使用します。

c. トークン発行者を検証する [Issuer] (発行者) を指定します。

d. [クレーム ID] を指定します。JWT のオーディエンスと一致する正規表現を指定する必要があります。

9. [Provisioning details] (プロビジョニングの詳細) ページで、[Developer edition] (デベロッパーエディション) を選択します。

10. [Create] (作成) を選択してインデックスを作成します。

11. インデックスが作成されるのを待ちます。 Amazon Kendra インデックスのハードウェアをプロビジョニングします。この演算には時間がかかる場合があります。

CLI

内部に共有シークレットを含む JWT トークンを使用できます。 AWS Secrets Managerシークレットは、base64 URL でエンコードされている必要があります。 Secrets Manager ARN が必要で、 Amazon Kendra `GetSecretValue` Secrets Manager ロールにはリソースへのアクセス権が必要です。 Secrets Manager を使用してリソースを暗号化する場合 AWS KMS、ロールには復号化アクションへのアクセス権も必要です。

JSON AWS CLI 入力ファイルを使用してインデックスを作成するには、まず必要なパラメータを含む JSON ファイルを作成します。

```
{
    "Name": "user-context",
    "Edition": "ENTERPRISE_EDITION",
    "RoleArn": "arn:aws:iam::account-id:role:/my-role",
```

```
    "UserTokenConfigurations": [
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "SECRET_MANAGER",
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "SecretManagerArn": "arn:aws:secretsmanager:us-west-2:account
 id:secret:/my-user-context-secret
            }
        }
    ],
    "UserContextPolicy": "USER_TOKEN"
}
```

デフォルトのユーザーフィールド名とグループフィールド名を上書きできます。UserNameAttributeField のデフォルト値は「ユーザー」です。GroupAttributeField のデフォルト値は「グループ」です。

次に、入力ファイルを使用して、create-index を呼び出します。例えば、JSON ファイルの名前が create-index-openid.json の場合、以下を使用できます。

```
aws kendra create-index --cli-input-json file://create-index-openid.json
```

シークレットは次のような形式になっている必要があります AWS Secrets Manager。

```
{
  "keys": [
    {
      "kid": "key_id",
      "alg": "HS256|HS384|HS512",
      "kty": "OCT",
      "use": "sig", //this value can be sig only for now
      "k": "secret",
      "nbf":"ISO1806 date format"
      "exp":"ISO1806 date format"
    }
  ]
}
```

JWT の詳細については、jwt.io を参照してください。

Python

JWT トークンを内部に共有シークレットとともに使用できます。 AWS Secrets Managerシークレットは、base64 URL でエンコードされている必要があります。 Secrets Manager ARN が必要で、 Amazon Kendra `GetSecretValue` Secrets Manager ロールにはリソースへのアクセス権が必要です。 Secrets Manager を使用してリソースを暗号化する場合 AWS KMS、ロールには復号化アクションへのアクセス権も必要です。

```
response = kendra.create_index(
    Name='user-context',
    Edition='ENTERPRISE_EDITION',
    RoleArn='arn:aws:iam::account-id:role:/my-role',
    UserTokenConfigurations=[
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "URL",
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "SecretManagerArn": "arn:aws:secretsmanager:us-west-2:account
 id:secret:/my-user-context-secret"
            }
        }
    ],
    UserContextPolicy='USER_TOKEN'
)
```

パブリックキーでの JSON ウェブトークン (JWT) の使用

次の例は、インデックスを作成するときに JSON Web Token (JWT) を公開鍵とともに使用してユーザーアクセスを制御する方法を示しています。JWT の詳細については、jwt.io を参照してください。

Console

1. [Create index] (インデックスの作成) を選択して、新しいインデックスの作成を開始します。
2. [Specify index details] (インデックスの詳細の指定) ページで、インデックスに名前と説明を付けます。

3. [IAM role] (IAM ロール) には、ロールを選択するか、または [Create a new role] (新規ロールの作成) を選択し、新しいロールを作成してロール名を指定します。 IAM ロールには「AmazonKendra-」というプレフィックスが付きます。
4. その他のフィールドはすべてデフォルトのままにしておきます。[次へ] を選択します。
5. [Configure user access control] (ユーザーアクセスコントロールの設定) ページの、[Access control settings] (アクセスコントロールの設定) で、[Yes] (はい) を選択し、アクセス制御にトークンを使用します。
6. [Token configuration] (トークンの設定) で、[JWT with public key] (パブリックキーを使用したJWT) を [Token type] (トークンタイプ) として選択します。
7. [Parameters for signing public key] (パブリックキーに署名するためのパラメータ) で、[Type of secret] (シークレットのタイプ) を選択します。既存の AWS Secrets Manager シークレットを使用するか、新しいシークレットを作成できます。

   新しいシークレットを作成するには、[New] (新規) を選択し、次に、以下のステップを実行します。

   a. 「 AWS Secrets Manager 新規シークレット」で、シークレット名を指定します。プレフィックス `AmazonKendra-` は、パブリックキーを保存すると追加されます。
   b. [Key ID] (キー ID) を指定します。キー ID は、トークンの JSON ウェブ署名をセキュア化するために使用されたキーを示すヒントです。
   c. トークンに署名 [Algorithm] (アルゴリズム) を選択します。これは、ID トークンの保護に使用される暗号化アルゴリズムです。RSA の詳細については、RSA Cryptography を参照してください。
   d. [Certificate attributes] (証明書属性) で、オプションの [Certificate chain] (証明書チェーン) を指定します。証明書チェーンは、証明書のリストで構成されます。サーバーの証明書で始まり、ルート証明書で終了します。
   e. オプション [Thumbprint or fingerprint] (サムプリントまたはフィンガープリント) を指定します。これは、すべての証明書データとその署名に対してコンピューティングされた証明書のハッシュです。
   f. [Exponent] (指数) を指定します。これは RSA パブリックキーの指数値です。これは、Base64urlUInt でエンコードされた値として表されます。
   g. [Modulus] (係数) を指定します。これは RSA パブリックキーの指数値です。これは、Base64urlUInt でエンコードされた値として表されます。
   h. [Save key] (キーを保存) を選択して新しいキーを保存します。
8. [Advanced configuration] (詳細設定) の [Optional] (オプション):

a. [Username] (ユーザーネーム) を指定して ACL チェックで使用します。

b. 1 つ以上の [Groups] (グループ) を指定して、ACL チェックで使用します。

c. トークン発行者を検証する [Issuer] (発行者) を指定します。

d. [Client Id(s)] (クライアント ID) を指定します。JWT のオーディエンスと一致する正規表現を指定する必要があります。

9. [Provisioning details] (プロビジョニングの詳細) ページで、[Developer edition] (デベロッパーエディション) を選択します。

10. [Create] (作成) を選択してインデックスを作成します。

11. インデックスが作成されるのを待ちます。 Amazon Kendra インデックスのハードウェアをプロビジョニングします。この演算には時間がかかる場合があります。

CLI

JWT は、 AWS Secrets Managerの内部のパブリックキーで使用できます。 Secrets Manager ARN が必要で、 Amazon Kendra `GetSecretValue` Secrets Manager ロールにはリソースへのアクセス権が必要です。 Secrets Manager を使用してリソースを暗号化する場合 AWS KMS、ロールには復号化アクションへのアクセス権も必要です。

JSON AWS CLI 入力ファイルを使用してインデックスを作成するには、まず必要なパラメータを含む JSON ファイルを作成します。

```
{
    "Name": "user-context",
    "Edition": "ENTERPRISE_EDITION",
    "RoleArn": "arn:aws:iam::account id:role:/my-role",
    "UserTokenConfigurationList": [
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "SECRET_MANAGER",
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "SecretManagerArn": "arn:aws:secretsmanager:us-west-2:account
 id:secret:/my-user-context-secret
            }
        }
    ],    "UserContextPolicy": "USER_TOKEN"
```

```
}
```

デフォルトのユーザーフィールド名とグループフィールド名を上書きできます。UserNameAttributeField のデフォルト値は「ユーザー」です。GroupAttributeField のデフォルト値は「グループ」です。

次に、入力ファイルを使用して、create-index を呼び出します。例えば、JSON ファイルの名前が create-index-openid.json の場合、以下を使用できます。

```
aws kendra create-index --cli-input-json file://create-index-openid.json
```

シークレットは次のような形式になっている必要があります Secrets Manager。

```
{
  "keys": [
    {
      "alg": "RS256|RS384|RS512",
      "kty": "RSA", //this can be RSA only for now
      "use": "sig", //this value can be sig only for now
      "n": "modulus of standard pem",
      "e": "exponent of standard pem",
      "kid": "key_id",
      "x5t": "certificate thumprint for x.509 cert",
      "x5c": [
        "certificate chain"
      ]
    }
  ]
}
```

JWT の詳細については、jwt.io を参照してください。

Python

```
response = kendra.create_index(
    Name='user-context',
    Edition='ENTERPRISE_EDITION',
    RoleArn='arn:aws:iam::account id:role:/my-role',
    UserTokenConfigurationList=[
        {
            "JwtTokenTypeConfiguration": {
                "KeyLocation": "URL",
```

```
                "Issuer": "optional: specify the issuer url",
                "ClaimRegex": "optional: regex to validate claims in the token",
                "UserNameAttributeField": "optional: user",
                "GroupAttributeField": "optional: group",
                "SecretManagerArn": "arn:aws:secretsmanager:us-west-2:account
 id:secret:/my-user-context-secret"
            }
        }
    ],
    UserContextPolicy='USER_TOKEN'
)
```

JSON の使用

次の例は、インデックスを作成するときに JSON を使用してユーザーアクセス制御を行う方法を示しています。

> **Warning**
>
> JSON トークンは検証されていないペイロードです。これは、 Amazon Kendra へのリクエストが信頼できるサーバーから送信され、ブラウザからのリクエストではない場合にのみ使用してください。

Console

1. [Create index] (インデックスの作成) を選択して、新しいインデックスの作成を開始します。
2. [Specify index details] (インデックスの詳細の指定) ページで、インデックスに名前と説明を付けます。
3. [IAM ロール] には、[ロール] を選択するか、または [新規ロールを作成] を選択し、新しいロールを作成してロール名を指定します。 IAM ロールには「AmazonKendra-」というプレフィックスが付きます。
4. その他のフィールドはすべてデフォルトのままにしておきます。[次へ] を選択します。
5. [Configure user access control] (ユーザーアクセスコントロールの設定) ページの、[Access control settings] (アクセスコントロールの設定) で、[Yes] (はい) を選択し、アクセス制御にトークンを使用します。
6. [Token configuration] (トークンの設定) で、[Token type] (トークンタイプ) は [JSON] を選択します。

7. [ユーザー名] を指定して ACL チェックで使用します。
8. 1 つ以上の [Groups] (グループ) を指定して、ACL チェックで使用します。
9. [次へ] を選択します。
10. [Provisioning details] (プロビジョニングの詳細) ページで、[Developer edition] (デベロッパーエディション) を選択します。
11. [Create] (作成) を選択してインデックスを作成します。
12. インデックスが作成されるまでお待ちください。 Amazon Kendra インデックスのハードウェアをプロビジョニングします。この演算には時間がかかる場合があります。

CLI

JSON AWS CLI 入力ファイルを使用してを使用してインデックスを作成するには、まず必要なパラメータを含む JSON ファイルを作成します。

```
{
    "Name": "user-context",
    "Edition": "ENTERPRISE_EDITION",
    "RoleArn": "arn:aws:iam::account-id:role:/my-role",
    "UserTokenConfigurations": [
        {
            "JsonTokenTypeConfiguration": {
                "UserNameAttributeField": "user",
                "GroupAttributeField": "group"
            }
        }
    ],
    "UserContextPolicy": "USER_TOKEN"
}
```

次に、入力ファイルを使用して、`create-index` を呼び出します。例えば、JSON ファイルの名前が `create-index-openid.json` の場合、以下を使用できます。

```
aws kendra create-index --cli-input-json file://create-index-openid.json
```

Open ID for を使用していない場合は AWS IAM Identity Center、JSON 形式のトークンをお送りください。その場合は、JSON トークンのどのフィールドにユーザー名が含まれ、どのフィールドにグループが含まれているかを指定する必要があります。グループフィールドの値は JSON 文

字列配列でなければなりません。例えば、SAML を使用している場合、トークンは次のようになります。

```
{
     "username" : "user1",
     "groups": [
        "group1",
        "group2"
     ]
}
```

TokenConfiguration は、ユーザー名とグループフィールド名を指定します。

```
{
    "UserNameAttributeField":"username",
    "GroupAttributeField":"groups"
}
```

Python

```
response = kendra.create_index(
    Name='user-context',
    Edition='ENTERPRISE_EDITION',
    RoleArn='arn:aws:iam::account-id:role:/my-role',
    UserTokenConfigurations=[
        {
            "JwtTokenTypeConfiguration": {
                "UserNameAttributeField": "user",
                "GroupAttributeField": "group",
            }
        }
    ],
    UserContextPolicy='USER_TOKEN'
)
```

データソースコネクタの作成

のデータソースコネクタを作成して、 ドキュメント Amazon Kendra に接続してインデックスを作成できます。 は、Microsoft SharePoint、Google Drive、その他多くのプロバイダーに接続 Amazon Kendra できます。データソースコネクタを作成するときは、ソースリポジトリへの接続に必要な Amazon Kendra 構成情報を指定します。ドキュメントをインデックスに直接追加する場合とは異なり、データソースを定期的にスキャンしてインデックスを更新できます。

例えば、 Amazon S3 バケットに税務書類のリポジトリが格納されているとします。ときどき、既存のドキュメントが変更され、新しいドキュメントが随時リポジトリに追加されます。リポジトリをデータソース Amazon Kendra として に追加すると、データソースとインデックスの間の定期的な同期を設定することで、インデックスを最新の状態に保つことができます。

コンソールまたは StartDataSourceSyncJob API を使用して、インデックスを手動で更新することもできます。それ以外の場合は、インデックスを更新してデータソースと同期させるスケジュールを設定します。

インデックスには複数のデータソースを使用できます。各データソースには、独自の更新スケジュールを設定できます。例えば、アーカイブが変更されるたびに、アーカイブされたドキュメントを手動で更新しながら、作業中のドキュメントのインデックスを毎日更新したり、時間ごとに更新したりできます。

ドキュメント取り込みプロセス中にドキュメントメタデータまたは属性とコンテンツを変更する場合は、「Amazon Kendra Custom Document Enrichment」を参照してください。

> **Note**
>
> 各ドキュメント ID は、インデックスごとに一意である必要があります。一意の ID でドキュメントにインデックスを付けるデータソースを作成してから、`BatchPutDocument` API を使用して同じドキュメントにインデックスを付けることはできません。その逆も同様です。データソースを削除してから `BatchPutDocument` API を使用して同じドキュメントにインデックスを付けることができます。その逆も可能です。`BatchPutDocument` および `BatchDeleteDocument` APIs を同じドキュメントセットの Amazon Kendra データソースコネクタと組み合わせて使用すると、データに不整合が生じる可能性があります。代わりに、Amazon Kendra カスタムデータソースコネクタの使用をお勧めします。

> **Note**
>
> インデックスに追加されるファイルは、UTF-8 でエンコードされたバイトストリームに存在する必要があります。のドキュメントの詳細については Amazon Kendra、「ドキュメント」を参照してください。

更新スケジュールの設定

データソースを作成または更新するときに、コンソールか、または `Schedule` パラメータを使用して、データソースを定期的に更新するように構成します。パラメータの内容は、`cron` 形式スケジュール文字列、またはインデックスをオンデマンドで更新することを示す空の文字列のいずれかを保持する文字列です。cron 式の形式については、 ユーザーガイドのルールのスケジュール式を参照してください。 は cron 式のみ Amazon Kendra をサポートします。 Amazon CloudWatch Events rate 式はサポートしていません。

言語設定

サポートされている言語で、データソース内のすべてのドキュメントにインデックスを作成できます。を呼び出すときに、データソース内のすべてのドキュメントの言語コードを指定しますCreateDataSource。ドキュメントにメタデータフィールドで指定された言語コードがない場合、データソースレベルですべてのドキュメントに指定された言語コードを使用して、ドキュメントのインデックスが作成されます。言語を指定しない場合、 Amazon Kendra はデフォルトで英語でデータソースのドキュメントをインデックス作成します。コードを含む、サポートされている言語の詳細については、英語以外の言語でドキュメントを追加するを参照してください。

コンソールを使用するサポートされている言語で、データソース内のすべてのドキュメントにインデックスを作成できます。新しいデータソースを追加する場合、[データソース] に移動してデータソースを編集するか、[データソースを追加] します。[Specify data source details] (データソースの詳細を指定) ページで、[Language] (言語) のドロップダウンから言語を選択します。Update (更新) を選択するか、続けて構成情報を入力してデータソースに接続します。

データソースコネクタ

このセクションでは、 および Amazon Kendra APIs Amazon Kendra で を使用して、サポートされているデータベース AWS Management Console とデータソースリポジトリ Amazon Kendra に接続する方法を示します。

トピック

- データソーステンプレートスキーマ
- Adobe Experience Manager
- Alfresco
- Aurora (MySQL)
- Aurora (PostgreSQL)
- Amazon FSx (ウィンドウズ)
- Amazon FSx (NetApp ONTAP)
- Amazon RDS/Aurora
- Amazon RDS (Microsoft SQL サーバー)
- Amazon RDS (MySQL)
- Amazon RDS (Oracle)
- Amazon RDS (PostgreSQL)
- Amazon S3
- Amazon Kendra ウェブクローラー
- Amazon WorkDocs
- [Box] (ボックス)
- Confluence
- カスタムデータソースコネクタ
- Dropbox
- Drupal
- GitHub
- Gmail
- Google ドライブ
- IBM DB2
- Jira
- Microsoft Exchange
- Microsoft OneDrive
- Microsoft SharePoint

- Microsoft SQL Server
- Microsoft Teams
- Microsoft Yammer
- MySQL
- Oracle Database
- PostgreSQL
- Quip
- Salesforce
- ServiceNow
- Slack
- Zendesk

データソーステンプレートスキーマ

以下は、テンプレートがサポートされているデータソースのテンプレートスキーマです。

トピック

- Adobe Experience Manager テンプレートスキーマ
- Amazon FSx （ Windows ） テンプレートスキーマ
- Amazon FSx (NetApp ONTAP) テンプレートスキーマ
- Alfresco テンプレートスキーマ
- Aurora (MySQL) テンプレートスキーマ
- Aurora (PostgreSQL) テンプレートスキーマ
- Amazon RDS (Microsoft SQL サーバー) テンプレートスキーマ
- Amazon RDS (MySQL) テンプレートスキーマ
- Amazon RDS (Oracle) テンプレートスキーマ
- Amazon RDS (PostgreSQL) テンプレートスキーマ
- Amazon S3 テンプレートスキーマ
- Amazon Kendra Web Crawler テンプレートスキーマ
- Confluence テンプレートスキーマ

- Dropbox テンプレートスキーマ
- Drupal テンプレートスキーマ
- GitHub テンプレートスキーマ
- Gmail テンプレートスキーマ
- Google Drive テンプレートスキーマ
- IBM DB2 テンプレートスキーマ
- Microsoft Exchange テンプレートスキーマ
- Microsoft OneDrive テンプレートスキーマ
- Microsoft SharePoint テンプレートスキーマ
- Microsoft SQL サーバーテンプレートスキーマ
- Microsoft Teams テンプレートスキーマ
- Microsoft Yammer テンプレートスキーマ
- MySQL テンプレートスキーマ
- Oracle Database テンプレートスキーマ
- PostgreSQL テンプレートスキーマ
- Salesforce テンプレートスキーマ
- ServiceNow テンプレートスキーマ
- Slack テンプレートスキーマ
- Zendesk テンプレートスキーマ

Adobe Experience Manager テンプレートスキーマ

データソーススキーマを含む JSON を TemplateConfiguration オブジェクトの一部として含めます。Adobe Experience Manager ホスト URL、認証タイプ、接続設定またはリポジトリエンドポイントの詳細の一部として、Adobe Experience Manager (AEM) をクラウドサービスとして使用するか AEM オンプレミスとして使用するかを指定します。また、データソースのタイプを `AEM` に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、CreateDataSource を呼び出すときに `Type` として `TEMPLATE` を指定します。

このデベロッパーガイドで提供されているテンプレートを使用できます。詳細については、「Adobe Experience Manager JSON スキーマ」を参照してください。

次の表では、AEM JSON スキーマのパラメーターについて説明します。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
aemUrl	Adobe Experience Manager ホスト URL。例えば、AEM オンプレミスを使用する場合は、ホスト名とポートを含めます。https://hostname:port。または、AEM をクラウドサービスとして使用する場合は、作成者 URL を使用できます。https://author-xxxxxx-xxxxxxx.adobeaemcloud.com。
authType	使用する認証のタイプ (`Basic` または `OAuth2`)。
deploymentType	使用する Adobe Experience Manager のタイプ (`CLOUD` または `ON_PREMISE`)。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• ページで • アセット	Adobe Experience Manager Amazon Kendra ページやアセットの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
timeZoneId	AEM On-Premise を使用していて、サーバーのタイムゾーンが AEM コネクタまたはインデックスのタイムゾーンと異なる場合は、 Amazon

構成	説明
	Kendra AEM コネクタまたはインデックスに合わせてサーバーのタイムゾーンを指定できます。 AEM On-Premise のデフォルトのタイムゾーンは、AEM コネクタまたはインデックスのタイムゾーンです。 Amazon Kendra クラウドサービスとしての AEM のデフォルトのタイムゾーンはグリニッジ標準時です。
• pageRootPaths • assetRootPaths	ページとアセットのルートパスのリスト。例えば、ページのルートパスは /content/sub で、アセットのルートパスは /content/sub/asset1 という場合があります。
crawlAssets	アセットをクロールする場合は、`true` にします。
crawlPages	ページをクロールする場合は、`true` にします。
• pagePathInclusionパターン • pageNameInclusionパターン • assetPathInclusionパターン • assetTypeInclusionパターン • assetNameInclusionパターン	特定のページやアセットを Adobe Experience Manager データソースに含めるための正規表現のパターンのリスト。パターンに一致するページやアセットは、インデックスに含まれます。パターンに一致しないページやアセットは、インデックスから除外されます。ページやアセットが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのコンテンツはインデックスに含まれません。

構成	説明
• pagePathExclusionパターン • pageNameExclusionパターン • assetPathExclusionパターン • assetTypeInclusionパターン • assetNameInclusionパターン	Adobe Experience Manager データソースにある特定のページやアセットを除外するための正規表現のパターンのリスト。パターンに一致するページやアセットは、インデックスから除外されます。パターンに一致しないページやアセットは、インデックスに含まれます。ページやアセットが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのコンテンツはインデックスに含まれません。
pageComponents	インデックスを作成する特定のページコンポーネントの名前のリスト。
contentFragmentVariations	インデックスを作成する Adobe Experience Manager コンテンツフラグメントの特定の保存済みバリエーションの名前のリスト。
type	データソースのタイプ。データソースタイプとして `AEM` を指定します。
enableIdentityCrawler	`true` Amazon KendraのIDクローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループのID/プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	Adobe Experience Manager AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。これらのキーと値のペアについて詳しくは、「Adobe Experience Manager の接続手順」を参照してください。
version	現在サポートされているこのテンプレートのバージョン。

Adobe Experience Manager JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties":
  {
```

```
    "connectionConfiguration": {
      "type": "object",
      "properties":
      {
        "repositoryEndpointMetadata":
        {
          "type": "object",
          "properties":
          {
            "aemUrl":
            {
              "type": "string",
              "pattern": "https:.*"
            },
            "authType": {
              "type": "string",
              "enum": ["Basic", "OAuth2"]
            },
            "deploymentType": {
              "type": "string",
              "enum": ["CLOUD","ON_PREMISE"]
            }
          },
          "required":
          [
            "aemUrl",
            "authType",
            "deploymentType"
          ]
        }
      },
      "required":
      [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties":
      {
        "page":
        {
          "type": "object",
          "properties":
```

```
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
```

```
      },
      "required":
      [
        "fieldMappings"
      ]
    },
    "asset":
    {
      "type": "object",
      "properties":
      {
        "fieldMappings":
        {
          "type": "array",
          "items":
          [
            {
              "type": "object",
              "properties":
              {
                "indexFieldName":
                {
                  "type": "string"
                },
                "indexFieldType":
                {
                  "type": "string",
                  "enum":
                  [
                    "STRING",
                    "STRING_LIST",
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName":
                {
                  "type": "string"
                },
                "dateFieldFormat":
                {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
```

```
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      }
    }
  },
  "additionalProperties": {
    "type": "object",
    "properties":
    {
      "timeZoneId": {
        "type": "string",
        "enum": [
          "Africa/Abidjan",
          "Africa/Accra",
          "Africa/Addis_Ababa",
          "Africa/Algiers",
          "Africa/Asmara",
          "Africa/Asmera",
          "Africa/Bamako",
          "Africa/Bangui",
          "Africa/Banjul",
          "Africa/Bissau",
          "Africa/Blantyre",
          "Africa/Brazzaville",
          "Africa/Bujumbura",
          "Africa/Cairo",
          "Africa/Casablanca",
          "Africa/Ceuta",
          "Africa/Conakry",
          "Africa/Dakar",
          "Africa/Dar_es_Salaam",
```

```
            "Africa/Djibouti",
            "Africa/Douala",
            "Africa/El_Aaiun",
            "Africa/Freetown",
            "Africa/Gaborone",
            "Africa/Harare",
            "Africa/Johannesburg",
            "Africa/Juba",
            "Africa/Kampala",
            "Africa/Khartoum",
            "Africa/Kigali",
            "Africa/Kinshasa",
            "Africa/Lagos",
            "Africa/Libreville",
            "Africa/Lome",
            "Africa/Luanda",
            "Africa/Lubumbashi",
            "Africa/Lusaka",
            "Africa/Malabo",
            "Africa/Maputo",
            "Africa/Maseru",
            "Africa/Mbabane",
            "Africa/Mogadishu",
            "Africa/Monrovia",
            "Africa/Nairobi",
            "Africa/Ndjamena",
            "Africa/Niamey",
            "Africa/Nouakchott",
            "Africa/Ouagadougou",
            "Africa/Porto-Novo",
            "Africa/Sao_Tome",
            "Africa/Timbuktu",
            "Africa/Tripoli",
            "Africa/Tunis",
            "Africa/Windhoek",
            "America/Adak",
            "America/Anchorage",
            "America/Anguilla",
            "America/Antigua",
            "America/Araguaina",
            "America/Argentina/Buenos_Aires",
            "America/Argentina/Catamarca",
            "America/Argentina/ComodRivadavia",
            "America/Argentina/Cordoba",
```

```
            "America/Argentina/Jujuy",
            "America/Argentina/La_Rioja",
            "America/Argentina/Mendoza",
            "America/Argentina/Rio_Gallegos",
            "America/Argentina/Salta",
            "America/Argentina/San_Juan",
            "America/Argentina/San_Luis",
            "America/Argentina/Tucuman",
            "America/Argentina/Ushuaia",
            "America/Aruba",
            "America/Asuncion",
            "America/Atikokan",
            "America/Atka",
            "America/Bahia",
            "America/Bahia_Banderas",
            "America/Barbados",
            "America/Belem",
            "America/Belize",
            "America/Blanc-Sablon",
            "America/Boa_Vista",
            "America/Bogota",
            "America/Boise",
            "America/Buenos_Aires",
            "America/Cambridge_Bay",
            "America/Campo_Grande",
            "America/Cancun",
            "America/Caracas",
            "America/Catamarca",
            "America/Cayenne",
            "America/Cayman",
            "America/Chicago",
            "America/Chihuahua",
            "America/Ciudad_Juarez",
            "America/Coral_Harbour",
            "America/Cordoba",
            "America/Costa_Rica",
            "America/Creston",
            "America/Cuiaba",
            "America/Curacao",
            "America/Danmarkshavn",
            "America/Dawson",
            "America/Dawson_Creek",
            "America/Denver",
            "America/Detroit",
```

```
            "America/Dominica",
            "America/Edmonton",
            "America/Eirunepe",
            "America/El_Salvador",
            "America/Ensenada",
            "America/Fort_Nelson",
            "America/Fort_Wayne",
            "America/Fortaleza",
            "America/Glace_Bay",
            "America/Godthab",
            "America/Goose_Bay",
            "America/Grand_Turk",
            "America/Grenada",
            "America/Guadeloupe",
            "America/Guatemala",
            "America/Guayaquil",
            "America/Guyana",
            "America/Halifax",
            "America/Havana",
            "America/Hermosillo",
            "America/Indiana/Indianapolis",
            "America/Indiana/Knox",
            "America/Indiana/Marengo",
            "America/Indiana/Petersburg",
            "America/Indiana/Tell_City",
            "America/Indiana/Vevay",
            "America/Indiana/Vincennes",
            "America/Indiana/Winamac",
            "America/Indianapolis",
            "America/Inuvik",
            "America/Iqaluit",
            "America/Jamaica",
            "America/Jujuy",
            "America/Juneau",
            "America/Kentucky/Louisville",
            "America/Kentucky/Monticello",
            "America/Knox_IN",
            "America/Kralendijk",
            "America/La_Paz",
            "America/Lima",
            "America/Los_Angeles",
            "America/Louisville",
            "America/Lower_Princes",
            "America/Maceio",
```

```
            "America/Managua",
            "America/Manaus",
            "America/Marigot",
            "America/Martinique",
            "America/Matamoros",
            "America/Mazatlan",
            "America/Mendoza",
            "America/Menominee",
            "America/Merida",
            "America/Metlakatla",
            "America/Mexico_City",
            "America/Miquelon",
            "America/Moncton",
            "America/Monterrey",
            "America/Montevideo",
            "America/Montreal",
            "America/Montserrat",
            "America/Nassau",
            "America/New_York",
            "America/Nipigon",
            "America/Nome",
            "America/Noronha",
            "America/North_Dakota/Beulah",
            "America/North_Dakota/Center",
            "America/North_Dakota/New_Salem",
            "America/Nuuk",
            "America/Ojinaga",
            "America/Panama",
            "America/Pangnirtung",
            "America/Paramaribo",
            "America/Phoenix",
            "America/Port-au-Prince",
            "America/Port_of_Spain",
            "America/Porto_Acre",
            "America/Porto_Velho",
            "America/Puerto_Rico",
            "America/Punta_Arenas",
            "America/Rainy_River",
            "America/Rankin_Inlet",
            "America/Recife",
            "America/Regina",
            "America/Resolute",
            "America/Rio_Branco",
            "America/Rosario",
```

```
            "America/Santa_Isabel",
            "America/Santarem",
            "America/Santiago",
            "America/Santo_Domingo",
            "America/Sao_Paulo",
            "America/Scoresbysund",
            "America/Shiprock",
            "America/Sitka",
            "America/St_Barthelemy",
            "America/St_Johns",
            "America/St_Kitts",
            "America/St_Lucia",
            "America/St_Thomas",
            "America/St_Vincent",
            "America/Swift_Current",
            "America/Tegucigalpa",
            "America/Thule",
            "America/Thunder_Bay",
            "America/Tijuana",
            "America/Toronto",
            "America/Tortola",
            "America/Vancouver",
            "America/Virgin",
            "America/Whitehorse",
            "America/Winnipeg",
            "America/Yakutat",
            "America/Yellowknife",
            "Antarctica/Casey",
            "Antarctica/Davis",
            "Antarctica/DumontDUrville",
            "Antarctica/Macquarie",
            "Antarctica/Mawson",
            "Antarctica/McMurdo",
            "Antarctica/Palmer",
            "Antarctica/Rothera",
            "Antarctica/South_Pole",
            "Antarctica/Syowa",
            "Antarctica/Troll",
            "Antarctica/Vostok",
            "Arctic/Longyearbyen",
            "Asia/Aden",
            "Asia/Almaty",
            "Asia/Amman",
            "Asia/Anadyr",
```

```
            "Asia/Aqtau",
            "Asia/Aqtobe",
            "Asia/Ashgabat",
            "Asia/Ashkhabad",
            "Asia/Atyrau",
            "Asia/Baghdad",
            "Asia/Bahrain",
            "Asia/Baku",
            "Asia/Bangkok",
            "Asia/Barnaul",
            "Asia/Beirut",
            "Asia/Bishkek",
            "Asia/Brunei",
            "Asia/Calcutta",
            "Asia/Chita",
            "Asia/Choibalsan",
            "Asia/Chongqing",
            "Asia/Chungking",
            "Asia/Colombo",
            "Asia/Dacca",
            "Asia/Damascus",
            "Asia/Dhaka",
            "Asia/Dili",
            "Asia/Dubai",
            "Asia/Dushanbe",
            "Asia/Famagusta",
            "Asia/Gaza",
            "Asia/Harbin",
            "Asia/Hebron",
            "Asia/Ho_Chi_Minh",
            "Asia/Hong_Kong",
            "Asia/Hovd",
            "Asia/Irkutsk",
            "Asia/Istanbul",
            "Asia/Jakarta",
            "Asia/Jayapura",
            "Asia/Jerusalem",
            "Asia/Kabul",
            "Asia/Kamchatka",
            "Asia/Karachi",
            "Asia/Kashgar",
            "Asia/Kathmandu",
            "Asia/Katmandu",
            "Asia/Khandyga",
```

```
            "Asia/Kolkata",
            "Asia/Krasnoyarsk",
            "Asia/Kuala_Lumpur",
            "Asia/Kuching",
            "Asia/Kuwait",
            "Asia/Macao",
            "Asia/Macau",
            "Asia/Magadan",
            "Asia/Makassar",
            "Asia/Manila",
            "Asia/Muscat",
            "Asia/Nicosia",
            "Asia/Novokuznetsk",
            "Asia/Novosibirsk",
            "Asia/Omsk",
            "Asia/Oral",
            "Asia/Phnom_Penh",
            "Asia/Pontianak",
            "Asia/Pyongyang",
            "Asia/Qatar",
            "Asia/Qostanay",
            "Asia/Qyzylorda",
            "Asia/Rangoon",
            "Asia/Riyadh",
            "Asia/Saigon",
            "Asia/Sakhalin",
            "Asia/Samarkand",
            "Asia/Seoul",
            "Asia/Shanghai",
            "Asia/Singapore",
            "Asia/Srednekolymsk",
            "Asia/Taipei",
            "Asia/Tashkent",
            "Asia/Tbilisi",
            "Asia/Tehran",
            "Asia/Tel_Aviv",
            "Asia/Thimbu",
            "Asia/Thimphu",
            "Asia/Tokyo",
            "Asia/Tomsk",
            "Asia/Ujung_Pandang",
            "Asia/Ulaanbaatar",
            "Asia/Ulan_Bator",
            "Asia/Urumqi",
```

```
            "Asia/Ust-Nera",
            "Asia/Vientiane",
            "Asia/Vladivostok",
            "Asia/Yakutsk",
            "Asia/Yangon",
            "Asia/Yekaterinburg",
            "Asia/Yerevan",
            "Atlantic/Azores",
            "Atlantic/Bermuda",
            "Atlantic/Canary",
            "Atlantic/Cape_Verde",
            "Atlantic/Faeroe",
            "Atlantic/Faroe",
            "Atlantic/Jan_Mayen",
            "Atlantic/Madeira",
            "Atlantic/Reykjavik",
            "Atlantic/South_Georgia",
            "Atlantic/St_Helena",
            "Atlantic/Stanley",
            "Australia/ACT",
            "Australia/Adelaide",
            "Australia/Brisbane",
            "Australia/Broken_Hill",
            "Australia/Canberra",
            "Australia/Currie",
            "Australia/Darwin",
            "Australia/Eucla",
            "Australia/Hobart",
            "Australia/LHI",
            "Australia/Lindeman",
            "Australia/Lord_Howe",
            "Australia/Melbourne",
            "Australia/NSW",
            "Australia/North",
            "Australia/Perth",
            "Australia/Queensland",
            "Australia/South",
            "Australia/Sydney",
            "Australia/Tasmania",
            "Australia/Victoria",
            "Australia/West",
            "Australia/Yancowinna",
            "Brazil/Acre",
            "Brazil/DeNoronha",
```

```
            "Brazil/East",
            "Brazil/West",
            "CET",
            "CST6CDT",
            "Canada/Atlantic",
            "Canada/Central",
            "Canada/Eastern",
            "Canada/Mountain",
            "Canada/Newfoundland",
            "Canada/Pacific",
            "Canada/Saskatchewan",
            "Canada/Yukon",
            "Chile/Continental",
            "Chile/EasterIsland",
            "Cuba",
            "EET",
            "EST5EDT",
            "Egypt",
            "Eire",
            "Etc/GMT",
            "Etc/GMT+0",
            "Etc/GMT+1",
            "Etc/GMT+10",
            "Etc/GMT+11",
            "Etc/GMT+12",
            "Etc/GMT+2",
            "Etc/GMT+3",
            "Etc/GMT+4",
            "Etc/GMT+5",
            "Etc/GMT+6",
            "Etc/GMT+7",
            "Etc/GMT+8",
            "Etc/GMT+9",
            "Etc/GMT-0",
            "Etc/GMT-1",
            "Etc/GMT-10",
            "Etc/GMT-11",
            "Etc/GMT-12",
            "Etc/GMT-13",
            "Etc/GMT-14",
            "Etc/GMT-2",
            "Etc/GMT-3",
            "Etc/GMT-4",
            "Etc/GMT-5",
```

```
            "Etc/GMT-6",
            "Etc/GMT-7",
            "Etc/GMT-8",
            "Etc/GMT-9",
            "Etc/GMT0",
            "Etc/Greenwich",
            "Etc/UCT",
            "Etc/UTC",
            "Etc/Universal",
            "Etc/Zulu",
            "Europe/Amsterdam",
            "Europe/Andorra",
            "Europe/Astrakhan",
            "Europe/Athens",
            "Europe/Belfast",
            "Europe/Belgrade",
            "Europe/Berlin",
            "Europe/Bratislava",
            "Europe/Brussels",
            "Europe/Bucharest",
            "Europe/Budapest",
            "Europe/Busingen",
            "Europe/Chisinau",
            "Europe/Copenhagen",
            "Europe/Dublin",
            "Europe/Gibraltar",
            "Europe/Guernsey",
            "Europe/Helsinki",
            "Europe/Isle_of_Man",
            "Europe/Istanbul",
            "Europe/Jersey",
            "Europe/Kaliningrad",
            "Europe/Kiev",
            "Europe/Kirov",
            "Europe/Kyiv",
            "Europe/Lisbon",
            "Europe/Ljubljana",
            "Europe/London",
            "Europe/Luxembourg",
            "Europe/Madrid",
            "Europe/Malta",
            "Europe/Mariehamn",
            "Europe/Minsk",
            "Europe/Monaco",
```

```
            "Europe/Moscow",
            "Europe/Nicosia",
            "Europe/Oslo",
            "Europe/Paris",
            "Europe/Podgorica",
            "Europe/Prague",
            "Europe/Riga",
            "Europe/Rome",
            "Europe/Samara",
            "Europe/San_Marino",
            "Europe/Sarajevo",
            "Europe/Saratov",
            "Europe/Simferopol",
            "Europe/Skopje",
            "Europe/Sofia",
            "Europe/Stockholm",
            "Europe/Tallinn",
            "Europe/Tirane",
            "Europe/Tiraspol",
            "Europe/Ulyanovsk",
            "Europe/Uzhgorod",
            "Europe/Vaduz",
            "Europe/Vatican",
            "Europe/Vienna",
            "Europe/Vilnius",
            "Europe/Volgograd",
            "Europe/Warsaw",
            "Europe/Zagreb",
            "Europe/Zaporozhye",
            "Europe/Zurich",
            "GB",
            "GB-Eire",
            "GMT",
            "GMT0",
            "Greenwich",
            "Hongkong",
            "Iceland",
            "Indian/Antananarivo",
            "Indian/Chagos",
            "Indian/Christmas",
            "Indian/Cocos",
            "Indian/Comoro",
            "Indian/Kerguelen",
            "Indian/Mahe",
```

```
            "Indian/Maldives",
            "Indian/Mauritius",
            "Indian/Mayotte",
            "Indian/Reunion",
            "Iran",
            "Israel",
            "Jamaica",
            "Japan",
            "Kwajalein",
            "Libya",
            "MET",
            "MST7MDT",
            "Mexico/BajaNorte",
            "Mexico/BajaSur",
            "Mexico/General",
            "NZ",
            "NZ-CHAT",
            "Navajo",
            "PRC",
            "PST8PDT",
            "Pacific/Apia",
            "Pacific/Auckland",
            "Pacific/Bougainville",
            "Pacific/Chatham",
            "Pacific/Chuuk",
            "Pacific/Easter",
            "Pacific/Efate",
            "Pacific/Enderbury",
            "Pacific/Fakaofo",
            "Pacific/Fiji",
            "Pacific/Funafuti",
            "Pacific/Galapagos",
            "Pacific/Gambier",
            "Pacific/Guadalcanal",
            "Pacific/Guam",
            "Pacific/Honolulu",
            "Pacific/Johnston",
            "Pacific/Kanton",
            "Pacific/Kiritimati",
            "Pacific/Kosrae",
            "Pacific/Kwajalein",
            "Pacific/Majuro",
            "Pacific/Marquesas",
            "Pacific/Midway",
```

```
            "Pacific/Nauru",
            "Pacific/Niue",
            "Pacific/Norfolk",
            "Pacific/Noumea",
            "Pacific/Pago_Pago",
            "Pacific/Palau",
            "Pacific/Pitcairn",
            "Pacific/Pohnpei",
            "Pacific/Ponape",
            "Pacific/Port_Moresby",
            "Pacific/Rarotonga",
            "Pacific/Saipan",
            "Pacific/Samoa",
            "Pacific/Tahiti",
            "Pacific/Tarawa",
            "Pacific/Tongatapu",
            "Pacific/Truk",
            "Pacific/Wake",
            "Pacific/Wallis",
            "Pacific/Yap",
            "Poland",
            "Portugal",
            "ROK",
            "Singapore",
            "SystemV/AST4",
            "SystemV/AST4ADT",
            "SystemV/CST6",
            "SystemV/CST6CDT",
            "SystemV/EST5",
            "SystemV/EST5EDT",
            "SystemV/HST10",
            "SystemV/MST7",
            "SystemV/MST7MDT",
            "SystemV/PST8",
            "SystemV/PST8PDT",
            "SystemV/YST9",
            "SystemV/YST9YDT",
            "Turkey",
            "UCT",
            "US/Alaska",
            "US/Aleutian",
            "US/Arizona",
            "US/Central",
            "US/East-Indiana",
```

```
            "US/Eastern",
            "US/Hawaii",
            "US/Indiana-Starke",
            "US/Michigan",
            "US/Mountain",
            "US/Pacific",
            "US/Samoa",
            "UTC",
            "Universal",
            "W-SU",
            "WET",
            "Zulu",
            "EST",
            "HST",
            "MST",
            "ACT",
            "AET",
            "AGT",
            "ART",
            "AST",
            "BET",
            "BST",
            "CAT",
            "CNT",
            "CST",
            "CTT",
            "EAT",
            "ECT",
            "IET",
            "IST",
            "JST",
            "MIT",
            "NET",
            "NST",
            "PLT",
            "PNT",
            "PRT",
            "PST",
            "SST",
            "VST"
          ]
        },
        "pageRootPaths":
        {
```

```
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "assetRootPaths":
        {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "crawlAssets":
        {
          "type": "boolean"
        },
        "crawlPages":
        {
          "type": "boolean"
        },
        "pagePathInclusionPatterns":
        {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "pagePathExclusionPatterns":
        {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "pageNameInclusionPatterns":
        {
          "type": "array",
          "items":
          {
            "type": "string"
```

```
        }
      },
      "pageNameExclusionPatterns":
      {
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "assetPathInclusionPatterns":
      {
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "assetPathExclusionPatterns":
      {
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "assetTypeInclusionPatterns":
      {
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "assetTypeExclusionPatterns":
      {
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "assetNameInclusionPatterns":
      {
```

```
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "assetNameExclusionPatterns":
        {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "pageComponents": {
          "type": "array",
          "items": {
            "type": "object"
            }
        },
        "contentFragmentVariations": {
          "type": "array",
          "items": {
            "type": "object"
          }
        },
        "cugExemptedPrincipals": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      },
      "required":
      []
    },
    "type": {
      "type": "string",
      "pattern": "AEM"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
```

```
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

Amazon FSx（Windows）テンプレートスキーマ

データソーススキーマを含む JSON を TemplateConfiguration オブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として、ファイルシステム ID を指定します。また、データソースのタイプ、認証情報のシークレットFSX、その他の必要な設定も指定する必要があります。次に、CreateDataSource を呼び出すときに Type として TEMPLATE を指定します。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon FSx (Windows) JSON スキーマ を参照してください。

次の表では、 Amazon FSx (Windows) JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
fileSystemId	Amazon FSx ファイルシステムの識別子。ファイルシステム ID Amazon FSx はコンソールのファイルシステムダッシュボードで確認できます。
fileSystemType	Amazon FSx ファイルシステムのタイプ。Windows File Serverファイルシステムのタイプとして使用するには、を指定しますWINDOWS。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
すべて	Amazon FSx Amazon Kendra データソース内のファイルの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
isCrawlAcl	trueACL があって、それをアクセス制御に使用したい場合に、文書のアクセス制御リスト (ACL) 情報をクロールします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

構成	説明
inclusionPatterns	Amazon FSx データソースに特定のファイルを含めるための正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
exclusionPatterns	Amazon FSx データソース内の特定のファイルを除外するための正規表現パターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
enableIdentityCrawler	`true` Amazon KendraのIDクローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループのID/プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
type	データソースのタイプ。Windows ファイルシステムのデータソースの場合は、を指定しますFSX。

Amazon FSx (Windows) JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "fileSystemId": {
              "type": "string",
              "pattern": "fs-.*"
            },
            "fileSystemType": {
              "type": "string",
              "pattern": "WINDOWS"
```

```
            }
          },
          "required": ["fileSystemId", "fileSystemType"]
        }
      }
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "All": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "DATE"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": ["fieldMappings"]
        }
```

```
      },
      "required": ["All"]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "isCrawlAcl": {
          "type": "boolean"
        },
        "exclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      },
      "required": []
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL"
      ]
    },
    "type" : {
      "type" : "string",
      "pattern": "FSX"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
```

```
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "enableIdentityCrawler",
    "additionalProperties",
    "type"
  ]
}
```

Amazon FSx (NetApp ONTAP) テンプレートスキーマ

データソーススキーマを含む JSON を TemplateConfiguration オブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として、ファイルシステム ID と Storage Virtual Machine (SVM) を指定します。また、データソースのタイプ、認証情報のシークレットFSXONTAP、その他の必要な設定も指定する必要があります。次に、CreateDataSource を呼び出すときに Type として TEMPLATE を指定します。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon FSx (NetApp ONTAP) JSON スキーマ を参照してください。

次の表では、 Amazon FSx (NetApp ONTAP) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
fileSystemId	Amazon FSx ファイルシステムの識別子。ファイルシステム ID Amazon FSx はコンソールのファイルシステムダッシュボードで確認できます。ONTAP Amazon FSx のコンソールでファイルシステムを作成する方法については、『FSx for ONTAP ユーザガイド』の「 NetApp

構成	説明
	NetAppONTAP 入門ガイド」を参照してください。
fileSystemType	Amazon FSx ファイルシステムのタイプ。NetApp ONTAPファイルシステムのタイプとして使用するには、を指定しますONTAP。
SVMid	Amazon FSx のファイルシステムで使用されるストレージ仮想マシン (SVM) の識別子。NetApp ONTAPSVM ID は、 Amazon FSx コンソールのファイルシステムダッシュボードでファイルシステム ID を選択し、次に [ストレージ仮想マシン] を選択すると確認できます。 Amazon FSx のコンソールでファイルシステムを作成する方法についてはNetApp ONTAP、『FSx for ONTAP ユーザガイド』の「 NetAppONTAP 入門ガイド」を参照してください。
プロトコルタイプ	Windows では共通インターネットファイルシステム (CIFS) プロトコルを使用するか、Linux ではネットワークファイルシステム (NFS) プロトコルを使用するか。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
file	Amazon FSx Amazon Kendra データソース内のファイルの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。データソースのフィールド名はファイルのカスタムメタデータに存在している必要があります。

構成	説明
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
crawlAcl	`true`文書のアクセス制御リスト (ACL) 情報をクロールするには、ACL があって、それをアクセス制御に使用したい場合に使用します。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
inclusionPatterns	Amazon FSx データソースに特定のファイルを含めるための正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
exclusionPatterns	Amazon FSx データソース内の特定のファイルを除外するための正規表現パターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
type	データソースのタイプ。NetApp ONTAPファイルシステムのデータソースの場合は、を指定しますFSXONTAP。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。

secretArn	AWS Secrets Manager ファイルシステムへの接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。Amazon FSx シークレットには、次のキーを持つ JSON 構造を含める必要があります。
	`{ "username": " user@corp.example.com ", "password": " password" }`
	Amazon FSx ファイルシステムに NFS プロトコルを使用する場合、シークレットは次のキーを含む JSON 構造で保存されます。
	`{ "leftId": "left ID", "rightId": " right ID", "preSharedKey": " pre-shared key " }`

Amazon FSx (NetApp ONTAP) JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "fileSystemId": {
              "type": "string",
                "pattern": "^(fs-[0-9a-f]{8,21})$"
            },
            "fileSystemType": {
              "type": "string",
              "enum": ["ONTAP"]
            },
            "svmId": {
              "type": "string",
              "pattern": "^(svm-[0-9a-f]{17,21})$"
            },
            "protocolType": {
              "type": "string",
              "enum": [
                "CIFS",
                "NFS"
              ]
            }
          },
          "required": [
            "fileSystemId",
            "fileSystemType"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
```

```
      "properties": {
        "file": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string",
                      "pattern": "^([a-zA-Z_]{1,20})$"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string",
                      "pattern": "^([a-zA-Z_]{1,20})$"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ],
              "maxItems": 50
            }
          },
          "required": [
            "fieldMappings"
```

```
          ]
        }
      },
      "required": [
        "file"
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "crawlAcl": {
          "type": "boolean"
        },
        "inclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string",
            "maxLength": 30
          },
          "maxItems": 100
        },
        "exclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string",
            "maxLength": 30
          },
          "maxItems": 100
        }
      }
    },
    "type": {
      "type": "string",
      "pattern": "FSXONTAP"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL"
      ]
    },
    "secretArn": {
      "type": "string",
```

```
      "pattern": "arn:aws:secretsmanager:.*"
    }
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

Alfresco テンプレートスキーマ

データソーススキーマを含む JSON を TemplateConfiguration オブジェクトの一部として含めます。Alfresco サイト ID、リポジトリ URL、ユーザーインターフェイス URL、認証タイプ、クラウドとオンプレミスのどちらを使用するか、クロールするコンテンツのタイプを指定します。これは接続設定またはリポジトリエンドポイントの詳細の一部として指定します。また、データソースのタイプを ALFRESCO に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、CreateDataSource を呼び出すときに Type として TEMPLATE を指定します。

このデベロッパーガイドで提供されているテンプレートを使用できます。Alfresco JSON スキーマを参照してください。

次の表では、Alfresco JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
siteId	Alfresco サイトの識別子。
repoUrl	Alfresco リポジトリの URL。リポジトリ URL は Alfresco 管理者から取得できます。例えば、Alfresco クラウド (PaaS) を使用している場合、リポジトリ URL は、https://company.alfrescocloud.com になる可能性があります。または、Alfresco オンプレミスを使用している場

構成	説明
	合は、リポジトリ URL は https://company-alfresco-instance.company-domain.suffix:port になる可能性があります。
webAppUrl	Alfresco ユーザーインターフェイスの URL。Alfresco ユーザーインターフェイスの URL はAlfresco管理者から取得できます。例えば、ユーザーインターフェイス URL は https://example.com とすることができます。
repositoryAdditionalProperties	リポジトリ/データソースエンドポイントに接続するための追加プロパティ。
authType	使用する認証のタイプ (`OAuth2` または `Basic`)。
タイプ: (デプロイ)	使用する Alfresco のタイプ (`PAAS` または `ON-PREM`)
crawlType	クロールするコンテンツのタイプ。`ASPECT` (Alfresco で「アスペクト」とマークされているコンテンツ)、`SITE_ID` (特定の Alfresco サイト内のコンテンツ)、または `ALL_SITES` (すべての Alfresco サイトにわたるコンテンツ) のいずれかです。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• ドキュメント • コメント	Alfresco Amazon Kendra ドキュメントとコメントの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。

構成	説明
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
aspectName	インデックスを作成する特定の「アスペクト」の名前。
aspectProperties	インデックスを作成する特定の「アスペクト」コンテンツプロパティのリスト。
enableFineGrainedコントロール	「アスペクト」をクロールする場合は、`true` にします。
isCrawlComment	`true`コメントをクロールします。
• inclusionFileNameパターン • inclusionFileTypeパターン • inclusionFilePathパターン	特定のファイルを Alfresco データソースに含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• exclusionFileNameパターン • exclusionFileTypeパターン • exclusionFilePathパターン	Alfresco データソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
type	データソースのタイプ。データソースタイプとして `ALFRESCO` を指定します。

構成	説明

secretArn	AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。Alfrescoシークレットには、次のキーを持つ JSON 構造を含める必要があります。
	
	基本認証を使用している場合。
	
	`{ "username": " user name", "password": " password" }`
	
	OAuth 2.0 認証を使用している場合。
	
	`{ "clientId": " client ID", "clientSecret": " client secret", "tokenUrl": " token URL" }`

syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。

構成	説明
enableIdentityCrawler	`true` Amazon Kendraの ID クローラを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID 情報やプリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。
version	現在サポートされているこのテンプレートのバージョン。

Alfresco JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "siteId": {
              "type": "string"
            },
            "repoUrl": {
              "type": "string"
            },
            "webAppUrl": {
              "type": "string"
            },
            "repositoryAdditionalProperties": {
              "type": "object",
              "properties": {
```

```
                "authType": {
                  "type": "string",
                  "enum": [
                    "OAuth2",
                    "Basic"
                  ]
                },
                "type": {
                  "type": "string",
                  "enum": [
                    "PAAS",
                    "ON_PREM"
                  ]
                },
                "crawlType": {
                  "type": "string",
                  "enum": [
                    "ASPECT",
                    "SITE_ID",
                    "ALL_SITES"
                  ]
                }
              }
            }
          }
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
```

```
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "comment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
```

```
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "aspectName": {
        "type": "string"
      },
      "aspectProperties": {
        "type": "array"
```

```
        },
        "enableFineGrainedControl": {
          "type": "boolean"
        },
        "isCrawlComment": {
          "type": "boolean"
        },
        "inclusionFileNamePatterns": {
          "type": "array"
        },
        "exclusionFileNamePatterns": {
          "type": "array"
        },
        "inclusionFileTypePatterns": {
          "type": "array"
        },
        "exclusionFileTypePatterns": {
          "type": "array"
        },
        "inclusionFilePathPatterns": {
          "type": "array"
        },
        "exclusionFilePathPatterns": {
          "type": "array"
        }
      }
    },
    "type": {
      "type": "string",
      "pattern": "ALFRESCO"
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL"
      ]
    },
    "enableIdentityCrawler": {
```

```
      "type": "boolean"
    },
    "version": {
      "type": "string",
      "anyOf": [
        {
          "pattern": "1.0.0"
        }
      ]
    }
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "type",
    "secretArn"
  ]
}
```

Aurora (MySQL) テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを mysql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Aurora (MySQL) JSON スキーマ を参照してください。

次の表では、 Aurora (MySQL) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqlまたはのいずれでもかまいませんdb2。postgresql oracle sqlserver

構成	説明
	• dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。

構成	説明
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Aurora (MySQL) JSON スキーマ

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Aurora (PostgreSQL) テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを JDBC に指定し、データベースタイプを postgresql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Aurora (PostgreSQL) JSON スキーマ を参照してください。

次の表では、 Aurora (PostgreSQL) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、またはのいずれでもかまいませんmysql。db2 postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。

構成	説明
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。

構成	説明
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして JDBC を指定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Aurora (PostgreSQL) JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
```

```
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
```

```
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
```

```
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Amazon RDS (Microsoft SQL サーバー) テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを sqlserver に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon RDS (Microsoft SQL サーバー) JSON スキーマ を参照してください。

次の表では、 Amazon RDS (Microsoft SQL Server) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqldb2またはのいずれでもかまいませんpostgresql 。oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィール

構成	説明
	ドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。

構成	説明
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Amazon RDS (Microsoft SQL サーバー) JSON スキーマ

```
{
```

```
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
  "connectionConfiguration": {
    "type": "object",
    "properties": {
      "repositoryEndpointMetadata": {
        "type": "object",
        "properties": {
          "dbType": {
            "type": "string",
            "enum": [
              "mysql",
              "db2",
              "postgresql",
              "oracle",
              "sqlserver"
            ]
          },
          "dbHost": {
            "type": "string"
          },
          "dbPort": {
            "type": "string"
          },
          "dbInstance": {
            "type": "string"
          }
        },
        "required": [
          "dbType",
          "dbHost",
          "dbPort",
          "dbInstance"
        ]
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Amazon RDS (MySQL) テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを mysql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon RDS (MySQL) JSON スキーマ を参照してください。

次の表では、 Amazon RDS (MySQL) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqlまたはのいずれでもかまいませんdb2。postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。

構成	説明
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。

構成	説明
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして JDBC を指定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Amazon RDS (MySQL) JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
```

```
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
```

```
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    },
    "required": [
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "primaryKey": {
        "type": "string"
      },
      "titleColumn": {
        "type": "string"
      },
      "bodyColumn": {
        "type": "string"
      },
      "sqlQuery": {
        "type": "string",
        "not": {
          "pattern": ";+"
        }
      },
      "timestampColumn": {
        "type": "string"
      },
      "timestampFormat": {
        "type": "string"
      },
      "timezone": {
        "type": "string"
      },
      "changeDetectingColumns": {
        "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
```

```
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Amazon RDS (Oracle) テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを oracle に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon RDS (Oracle) JSON スキーマ を参照してください。

次の表では、 Amazon RDS (Oracle) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqldb2またはのいずれでもかまいませんpostgresql 。oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。

構成	説明
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。

構成	説明
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Amazon RDS (Oracle) JSON スキーマ

```
{
```

```
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
  "connectionConfiguration": {
    "type": "object",
    "properties": {
      "repositoryEndpointMetadata": {
        "type": "object",
        "properties": {
          "dbType": {
            "type": "string",
            "enum": [
              "mysql",
              "db2",
              "postgresql",
              "oracle",
              "sqlserver"
            ]
          },
          "dbHost": {
            "type": "string"
          },
          "dbPort": {
            "type": "string"
          },
          "dbInstance": {
            "type": "string"
          }
        },
        "required": [
          "dbType",
          "dbHost",
          "dbPort",
          "dbInstance"
        ]
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Amazon RDS (PostgreSQL) テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを JDBC に指定し、データベースタイプを postgresql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon RDS (PostgreSQL) JSON スキーマ を参照してください。

次の表では、 Amazon RDS (PostgreSQL) JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、またはのいずれでもかまいませんmysql。db2 postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。

構成	説明
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。

構成	説明
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして JDBC を指定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Amazon RDS (PostgreSQL) JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
```

```
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
```

```
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    },
    "required": [
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "primaryKey": {
        "type": "string"
      },
      "titleColumn": {
        "type": "string"
      },
      "bodyColumn": {
        "type": "string"
      },
      "sqlQuery": {
        "type": "string",
        "not": {
          "pattern": ";+"
        }
      },
      "timestampColumn": {
        "type": "string"
      },
      "timestampFormat": {
        "type": "string"
      },
      "timezone": {
        "type": "string"
      },
      "changeDetectingColumns": {
        "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
```

```
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Amazon S3 テンプレートスキーマ

テンプレート設定の一部として、データソーススキーマを含む JSON を含めます。接続設定またはリポジトリエンドポイントの詳細の一部として S3 バケットの名前を指定します。また、データソースのタイプを S3 として指定し、その他の必要な設定も指定します。次に、Type呼び出すときにを指定しますTEMPLATECreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。S3 JSON スキーマ を参照してください。

次の表では、 Amazon S3 JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
BucketName	Amazon S3 バケットの名前。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
• inclusionPatterns • exclusionPatterns • inclusionPrefixes • exclusionPrefixes	Amazon S3 データソース内の特定のファイルを含めたり除外したりする正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しない

構成	説明
	ファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
aclConfigurationFileパス	Amazon Kendra インデックス内のドキュメントへのアクセスを制御するファイルパス。
metadataFilesPrefix	バケット内のメタデータファイルの場所。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
type	データソースのタイプ。データソースタイプとして S3 を指定します。
version	サポートされているテンプレートのバージョン。

S3 JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
```

```
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "BucketName": {
              "type": "string"
            }
          },
          "required": [
            "BucketName"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
```

```
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
        "document"
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "inclusionPatterns": {
          "type": "array"
        },
        "exclusionPatterns": {
          "type": "array"
        },
        "inclusionPrefixes": {
          "type": "array"
        },
        "exclusionPrefixes": {
          "type": "array"
        },
        "aclConfigurationFilePath": {
          "type": "string"
        },
        "metadataFilesPrefix": {
          "type": "string"
        }
      }
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FULL_CRAWL",
```

```
        "FORCED_FULL_CRAWL"
      ]
    },
    "type": {
      "type": "string",
      "pattern": "S3"
    },
    "version": {
      "type": "string",
      "anyOf": [
        {
          "pattern": "1.0.0"
        }
      ]
    }
  },
  "required": [
    "connectionConfiguration",
    "type",
    "syncMode",
    "repositoryConfigurations"
  ]
}
```

Amazon Kendra Web Crawler テンプレートスキーマ

データソーススキーマを含む JSON を TemplateConfiguration オブジェクトの一部として含めます。

接続設定またはリポジトリエンドポイントの詳細の一部として、シード URL または開始ポイント URL を指定するか、サイトマップ URL を指定できます。すべての URL を手動で一覧表示する代わりに、シード URL またはサイトマップ XML Amazon S3 ファイルのリストのテキストファイルを保存するバケットへのパスを指定できます。これらのファイルは S3 の ZIP ファイルにまとめることができます。

また、データソースのタイプWEBCRAWLERV2、ウェブサイトで認証が必要な場合はウェブサイト認証認証情報、認証タイプ、およびその他の必要な設定も指定できます。

次に、CreateDataSource を呼び出すときに Type として TEMPLATE を指定します。

> **Important**
>
> Web Crawler v2.0 コネクタの作成はではサポートされていません。 AWS CloudFormation サポートが必要な場合は Web Crawler v1.0 コネクタを使用してください。 AWS CloudFormation

インデックス作成するウェブサイトを選択するときは、Amazon 利用規定ポリシーおよびその他の Amazon 規約のすべてに準拠している必要があります。 Amazon Kendra Web Crawler は、自分の Web ページ、またはインデックス作成を許可されている Web ページのインデックスを作成する場合にのみ使用する必要があることに注意してください。 Amazon Kendra ウェブクローラーによるウェブサイトのインデックスの作成を停止する方法については、「Amazon Kendra Web Crawler 用の *robots.txt* ファイルの設定」を参照してください。

このデベロッパーガイドで提供されているテンプレートを使用できます。Amazon Kendra Web クローラー JSON スキーマ を参照してください。

次の表では、 Amazon Kendra Web クローラー JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
siteMapUrls	クロールするウェブサイトのサイトマップ URL のリスト。サイトマップの URL は最大 3 つまで一覧表示できます。
s3 SeedUrl	シードまたは開始ポイント URL のリストを格納するテキストファイルへの S3 パス。例えば s3://bucket-name/directory/ です。テキストファイル内の各 URL は、別々の行にフォーマットする必要があります。最大 100 件のシード URL を一覧表示できます。
s3 SiteMapUrl	サイトマップ XML ファイルへの S3 パス。例えば s3://bucket-name/directory/ です。サイト

構成	説明
	マップ XML ファイルは最大 3 件まで一覧表示できます。複数のサイトマップファイルを ZIP ファイルにまとめ、その ZIP Amazon S3 ファイルをバケットに保存できます。
seedUrlConnections	クロールするウェブサイトのシードまたは開始ポイント URL のリスト。最大 100 件のシード URL を一覧表示できます。
seedUrl	シードまたは開始点 URL。
認証	ウェブサイトが同じ認証を必要とする場合は認証タイプ、それ以外の場合は、`NoAuthentication` を指定します。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• webPage • 添付	Amazon Kendra ウェブページやウェブページファイルの属性やフィールド名をインデックスフィールド名にマッピングするオブジェクトのリスト。例えば、HTML ウェブページのタイトルタグを `_document_title` インデックスフィールドにマッピングできます。詳細については、データソースフィールドのマッピングを参照してください。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
rateLimit	1 分あたりウェブサイトホストごとにクロールされる URL の最大数。
maxFileSize	クロールするウェブページまたは添付ファイルの最大サイズ (MB 単位)。
crawlDepth	シード URL からクロールするレベル数。例えば、シード URL ページは深度 1 で、このページ上でクロールされるハイパーリンクはすべて深度 2 です。
maxLinksPerURL	ウェブサイトをクロールするときに含めるウェブページ上の URL の最大数。この数字はウェブページごとです。ウェブサイトのウェブページがクロールされると、ウェブページがリンクしているすべての URL もクロールされます。ウェブページ上の URL は、表示順にクロールされます。

構成	説明
crawlSubDomain	ウェブサイトのホスト名をサブドメインでクロールする場合は、true にします。例えば、シード URL が 「abc.example.com」の場合、「a.abc.example.com」と「b.abc.example.com」もクロールされます。crawlSubDomain またはを設定しない場合true、crawlAllDomain Amazon Kendra クロールするウェブサイトのドメインのみをクロールします。
crawlAllDomain	ウェブページがリンクするサブドメインおよびその他のドメインでウェブサイトのドメインをクロールする場合は、true にします。crawlSubDomain crawlAllDomain またはを設定しない場合true、クロール対象の Web Amazon Kendra サイトのドメインのみがクロールされます。
honorRobots	クロールするウェブサイトの robots.txt ディレクティブを優先する場合は、true にします。これらのディレクティブは、 Amazon Kendra Web Crawler が Web サイトをクロールする方法を制御します。 Amazon Kendra 特定のコンテンツのみをクロールできるか、どのコンテンツもクロールできないかを制御します。
crawlAttachments	ウェブページのリンク先のファイルをクロールする場合は、true にします。

構成	説明
• インクルージョン URL CrawlPatterns • インクルージョン URL IndexPatterns	特定の URL のクロールや、これらの URL ウェブページ上のハイパーリンクのインデックス作成を含む正規表現パターンのリスト。パターンに一致する URL は、インデックスに含まれます。パターンに一致しない URL は、インデックスから除外されます。URL が包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、その URL/ウェブサイトのウェブページはインデックスに含まれません。
• 除外 URL CrawlPatterns • 除外 URL IndexPatterns	特定の URL のクロールや、これらの URL ウェブページ上のハイパーリンクのインデックス作成を除外する正規表現パターンのリスト。パターンに一致する URL は、インデックスから除外されます。パターンに一致しない URL は、インデックスに含まれます。URL が包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、その URL/ウェブサイトのウェブページはインデックスに含まれません。
inclusionFileIndexパターン	特定のウェブページを含めるための正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
exclusionFileIndexパターン	特定のウェブページを除外するための正規表現パターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
proxy	ウェブプロキシ経由で内部ウェブサイトに接続するために必要となる構成情報。
ホスト	内部ウェブサイトへの接続に使用するプロキシサーバーのホストの名前。例えば、https://a.example.com/page1.html のホスト名は「a.example.com」です。
port	内部ウェブサイトへの接続に使用するプロキシサーバーのポート数。例えば、443 は HTTPS の標準ポートです。
secretArn (proxy)	ウェブサイトホストへの接続にウェブプロキシ認証情報が必要な場合は、 AWS Secrets Manager 認証情報を保存するシークレットを作成できます。シークレットの Amazon リソースネーム (ARN) を指定します。
type	データソースのタイプ。データソースタイプとして `WEBCRAWLERV2` を指定します。

構成	説明
secretArn	AWS Secrets Manager ウェブサイトへのアクセスに認証が必要な場合に使用されるシークレットの Amazon リソースネーム (ARN)。ウェブサイトの認証情報は、JSON キーと値のペアを含むシークレットに保存します。 ベーシックまたは NTML/Kerberos を使用している場合は、ユーザー名とパスワードを入力します。シークレットの JSON キーは、`userName` と `password` である必要があります。NTLM 認証プロトコルにはパスワードハッシュが含まれ、Kerberos 認証プロトコルにはパスワード暗号化が含まれます。 SAML 認証またはフォーム認証を使用する場合は、ユーザー名とパスワード、ユーザー名フィールド (SAML を使用する場合はユーザー名ボタン) に XPath、パスワードフィールドとボタンには XPath、ログインページの URL を入力します。シークレットの JSON キーは、`userName`、`password`、`userNameFieldXpath` 、`userNameButtonXpath` 、`passwordFieldXpath` 、`passwordButtonXpath` 、と `loginPageUrl` である必要があります。要素の XPath (XML パス言語) は、ウェブブラウザのデベロッパーツールを使用して確認できます。XPath は通常、次の形式に従います。`//tagname[@Attribute='Value']` 。 Amazon Kendra また、シークレットに含まれるエンドポイント情報 (シード URL) が、データソースエンドポイント設定の詳細で指定されているエンドポイント情報と同じかどうかも確認します。

構成	説明
version	現在サポートされているこのテンプレートのバージョン。

Amazon Kendra Web クローラー JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "siteMapUrls": {
              "type": "array",
              "items":{
                "type": "string",
                "pattern": "https://.*"
              }
            },
            "s3SeedUrl": {
              "type": "string",
              "pattern": "s3:.*"
            },
            "s3SiteMapUrl": {
              "type": "string",
              "pattern": "s3:.*"
            },
            "seedUrlConnections": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "seedUrl":{
                      "type": "string",
                      "pattern": "https://.*"
                    }
```

```
                },
                "required": [
                  "seedUrl"
                ]
              }
            ]
          },
          "authentication": {
            "type": "string",
            "enum": [
              "NoAuthentication",
              "BasicAuth",
              "NTLM_Kerberos",
              "Form",
              "SAML"
            ]
          }
        }
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
      "webPage": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": [
                      "STRING",
                      "DATE",
```

```
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "attachment": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE",
                    "LONG"
                  ]
                },
```

```
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "syncMode": {
    "type": "string",
    "enum": [
      "FORCED_FULL_CRAWL",
      "FULL_CRAWL"
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "rateLimit": {
        "type": "string",
        "default": "300"
      },
      "maxFileSize": {
        "type": "string",
        "default": "50"
      },
      "crawlDepth": {
        "type": "string",
        "default": "2"
```

```
        },
        "maxLinksPerUrl": {
          "type": "string",
          "default": "100"
        },
        "crawlSubDomain": {
          "type": "boolean",
          "default": false
        },
        "crawlAllDomain": {
          "type": "boolean",
          "default": false
        },
        "honorRobots": {
          "type": "boolean",
          "default": false
        },
        "crawlAttachments": {
          "type": "boolean",
          "default": false
        },
        "inclusionURLCrawlPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionURLCrawlPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionURLIndexPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionURLIndexPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
```

```
        },
        "inclusionFileIndexPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileIndexPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "proxy": {
          "type": "object",
          "properties": {
            "host": {
              "type": "string"
            },
            "port": {
              "type": "string"
            },
            "secretArn": {
              "type": "string",
              "minLength": 20,
              "maxLength": 2048
            }
          }
        }
      },
      "required": [
        "rateLimit",
        "maxFileSize",
        "crawlDepth",
        "crawlSubDomain",
        "crawlAllDomain",
        "maxLinksPerUrl",
        "honorRobots"
      ]
    },
    "type": {
      "type": "string",
      "pattern": "WEBCRAWLERV2"
    },
```

```
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "type",
    "additionalProperties"
  ]
}
```

Confluence テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfiguration接続設定またはリポジトリエンドポイントの詳細の一部として、Confluence ホスト URL、ホスティング方法、認証タイプを指定します。また、データソースのタイプを CONFLUENCEV2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Confluence JSON スキーマ を参照してください。

次の表では、Confluence JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。

構成	説明
hostUrl	Confluence インスタンスの URL。例えば、*https://example.confluence.com* などです。
type	Confluence インスタンスのホスティング方法 (SAAS および ON_PREM)。
authType	Confluence インスタンスの認証方法 (Basic、OAuth2、Personal-token)。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• スペース • ページで • ブログ • コメント • 添付	Confluence スペース、ページ、ブログ、コメント、 Amazon Kendra 添付ファイルの属性またはフィールド名をインデックスフィールド名にマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。Confluence データソースのフィールド名は、Confluence カスタムメタデータ内に存在する必要があります。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
isCrawlAcl	true ACL があり、それをアクセス制御に使用したい場合に、ドキュメントのアクセス制御リスト (ACL) 情報をクロールします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

構成	説明
fieldForUserID	ユーザー ID `email` にユーザーメールを使用するかどうかを指定します。 `email`デフォルトで使用され、現在サポートされている唯一のユーザー ID タイプです。
• inclusionSpaceKeyフィルター • exclusionSpaceKey[フィルター] • pageTitleRegEX • blogTitleReg元 • commentTitleReg元 • attachmentTitleReg元 • inclusionFileTypeパターン • exclusionFileTypeパターン • inclusionUrlPatterns • exclusionUrlPatterns	Confluence データソースに特定のファイルを含めるか、除外するための正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
proxyHost	使用する Web プロキシのホスト名 。`http://https://`またはプロトコルは含まれません。
proxyPort	ホスト URL トランスポートプロトコルが使用するポート番号。これは 0～65535 の範囲の値にする必要があります。

構成	説明
• isCrawlPersonalスペース • isCrawlArchivedスペース • isCrawlArchivedページ • isCrawlPage • isCrawlBlog • isCrawlPage[コメント] • isCrawlPage添付ファイル • isCrawlBlog[コメント] • isCrawlBlog添付ファイル	`true`Confluence パーソナルスペース、ページ、ブログ、ページコメント、ページ添付ファイル、ブログコメント、ブログ添付ファイル内のファイルをクロールします。
maxFileSizeInMegaBytes	クロールできるファイルサイズの上限を MB 単位で指定します。 Amazon Kendra Amazon Kendra 定義したサイズ制限内のファイルのみをクロールします。既定のファイルサイズは 50 MB です。最大ファイルサイズは 0 MB 以上 50 MB 以下でなければなりません。
type	データソースのタイプ。データソースタイプとして `CONFLUENCEV2` を指定します。
enableIdentityCrawler	`true` Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーおよびグループの ID /プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
secretARN	Confluence AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。これらのキーと値のペアについては、「Confluence の接続手順」を参照してください。
version	現在サポートされているこのテンプレートのバージョン。

Confluence JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "hostUrl": {
              "type": "string",
```

```
            "pattern": "https:.*"
          },
          "type": {
            "type": "string",
            "enum": [
              "SAAS",
              "ON_PREM"
            ]
          },
          "authType": {
            "type": "string",
            "enum": [
              "Basic",
              "OAuth2",
              "Personal-token"
            ]
          }
        },
        "required": [
          "hostUrl",
          "type",
          "authType"
        ]
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
      "space": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
```

```
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "STRING_LIST",
                    "DATE"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "page": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
```

```
                    "STRING",
                    "STRING_LIST",
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "blog": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "STRING_LIST",
```

```
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "comment": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "STRING_LIST",
                    "DATE",
                    "LONG"
```

```
                    ]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "attachment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": [
                      "STRING",
                      "STRING_LIST",
                      "DATE",
                      "LONG"
                    ]
                  },
```

```
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "usersAclS3FilePath": {
        "type": "string"
      },
      "isCrawlAcl": {
        "type": "boolean"
      },
      "fieldForUserId": {
        "type": "string"
      },
      "inclusionSpaceKeyFilter": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "exclusionSpaceKeyFilter": {
        "type": "array",
        "items": {
```

```
            "type": "string"
          }
        },
        "pageTitleRegEX": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "blogTitleRegEX": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "commentTitleRegEX": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "attachmentTitleRegEX": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "isCrawlPersonalSpace": {
          "type": "boolean"
        },
        "isCrawlArchivedSpace": {
          "type": "boolean"
        },
        "isCrawlArchivedPage": {
          "type": "boolean"
        },
        "isCrawlPage": {
          "type": "boolean"
        },
        "isCrawlBlog": {
          "type": "boolean"
        },
        "isCrawlPageComment": {
          "type": "boolean"
```

```
        },
        "isCrawlPageAttachment": {
          "type": "boolean"
        },
        "isCrawlBlogComment": {
          "type": "boolean"
        },
        "isCrawlBlogAttachment": {
          "type": "boolean"
        },
        "maxFileSizeInMegaBytes":  {
          "type":"string"
        },
        "inclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionUrlPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionUrlPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "proxyHost": {
          "type": "string"
        },
        "proxyPort": {
          "type": "string"
        }
      },
```

```
      "required": []
    },
    "type": {
      "type": "string",
      "pattern": "CONFLUENCEV2"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FULL_CRAWL",
        "FORCED_FULL_CRAWL"
      ]
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

Dropbox テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfiguration認証情報を保存するシークレットの一部として、Dropbox アプリキー、アプリシークレット、アクセストークンを指定します。データソースのタイプを DROPBOX として指定し、使用するアクセストークンのタイプ (一時的または永続的)、その他の必要な設定も指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Dropbox JSON スキーマを参照してください。

以下の表では Dropbox JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。このデータソースは repositoryEndpoint Metadata のエンドポイントを指定していません。その代わり、 AWS Secrets Manager 接続情報はユーザーが提供したシークレットに含まれます。secretArn
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• file • paper • papert • shortcut	Dropbox ファイル、Dropbox Paper、 Amazon Kendra およびインデックスフィールド名へのショートカットの属性やフィールド名をマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
secretARN	Dropbox AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN) 。シークレッ

構成	説明
	トには、次のキーを持つ JSON 構造を含める必要があります。 `{ "appKey": "Dropbox app key", "appSecret": " Dropbox app secret", "accesstoken": " temporary access token or refresh access token" }`
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
• inclusionFileNameパターン • inclusionFileTypeパターン	特定のファイルを Dropbox データソースに含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• exclusionFileNameパターン • exclusionFileTypeパターン	Dropbox データソース内の特定のファイル名とタイプを除外するための正規表現パターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• crawlFile • crawlPaper • crawlPapert • crawlShortcut	`true`Dropbox に保存されている Dropbox、Dropbox Paper ドキュメント、Dropbox Paper テンプレート、ウェブページのショートカット内のファイルをクロールできます。

構成	説明
type	データソースのタイプ。データソースタイプとして DROPBOX を指定します。
useChangeLog	Dropbox の変更ログを使用して、インデックス内の追加、更新、削除する必要があるドキュメントを特定する場合は、true。変更ログのサイズによっては、Dropbox Amazon Kendra 内のすべてのドキュメントをスキャンするよりも変更ログを使用する方が時間がかかる場合があります。
tokenType	アクセストークンのタイプ (永続的アクセストークンまたは一時アクセストークン) を指定します。4 時間後に有効期限が切れる 1 回限りのアクセストークンに頼るのではなく、Dropbox で有効期限が切れることのない更新アクセストークンを作成することをお勧めします。Dropbox デベロッパーコンソールでアプリと更新アクセストークンを作成し、シークレットでアクセストークンを渡します。
version	現在サポートされているこのテンプレートのバージョン。

Dropbox JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
          }
```

```
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "file": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": [
                          "STRING",
                          "STRING_LIST",
                          "LONG",
                          "DATE"
                        ]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
                      },
                      "dateFieldFormat": {
                        "type": "string",
                        "pattern": "dd-MM-yyyy HH:mm:ss"
                      }
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
                    ]
```

```
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "paper": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "LONG",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
```

```
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "papert": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "LONG",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
```

```
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "shortcut": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "LONG",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
```

```
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "secretArn": {
    "type": "string"
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "inclusionFileNamePatterns": {
        "type": "array"
      },
      "exclusionFileNamePatterns": {
        "type": "array"
      },
      "inclusionFileTypePatterns": {
        "type": "array"
      },
      "exclusionFileTypePatterns": {
        "type": "array"
      },
      "crawlFile": {
        "type": "boolean"
      },
      "crawlPaper": {
        "type": "boolean"
      },
      "crawlPapert": {
        "type": "boolean"
      },
      "crawlShortcut": {
        "type": "boolean"
      }
    }
  },
  "type": {
```

```
      "type": "string",
      "pattern": "DROPBOX"
    },
    "useChangeLog": {
      "type": "string",
      "enum": [
        "true",
        "false"
      ]
    },
    "tokenType": {
      "type": "string",
      "enum": [
        "PERMANENT",
        "TEMPORARY"
      ]
    },
    "version": {
      "type": "string",
      "anyOf": [
        {
          "pattern": "1.0.0"
        }
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "useChangeLog",
    "secretArn",
    "type",
    "tokenType"
  ]
}
```

Drupal テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として、Drupal ホスト URL と認証タイプを指定します。また、データソースのタイプを DRUPAL に指定します。認証情報のシークレット、

およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Drupal JSON スキーマ を参照してください。

次の表では、Drupal JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
hostUrl	Drupal ウェブサイトのホスト URL。例えば、*https://<hostname>/<drupalsitename>* 。
repositoryConfigurations	データソースのコンテンツに関する設定情報。
• content • コメント • 添付	Drupal ファイルの属性またはフィールド名をマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。Drupal データソースのフィールド名は、Drupal カスタムメタデータ内に存在する必要があります。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
• inclusionFileNameパターン • articleTitleInclusionパターン • pageTitleInclusionパターン • customContentTitleInclusionPatterns • basicBlockTitleInclusionPatterns • customBlockTitleInclusionPatterns	Drupal データソースにある特定のファイルを含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
• exclusionFileNameパターン • articleTitleExclusionパターン • pageTitleExclusionパターン • customContentTitleExclusionPatterns • basicBlockTitleExclusionPatterns • customBlockTitleExclusionPatterns	Drupal データソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
contentDefinitions • contentType • fieldDefinition • isCrawlComments • isCrawlFiles • isCrawlArticle • isCrawlBasicページ • isCrawlBasic[ブロック] • isCrawlCustomContentTypesList	クロールするコンテンツタイプと、選択したコンテンツタイプのコメントと添付ファイルをクロールするかどうかを指定します。
type	データソースのタイプ。データソースタイプとして DRUPAL を指定します。
authType	使用する認証のタイプ (BASIC-AUTH または OAUTH2)。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
enableIdentityCrawler	true Amazon Kendraの ID クローラを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID 情報やプリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明

secretARN	Drupal AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。
	
	基本認証を使用している場合。
	
	`{ "username": "user name", "passwords": "password" }`
	
	OAuth 2.0 認証を使用している場合。
	
	`{ "username": "user name", "password": "password" , "clientId": "client id", "clientSecret": "client secret" }`

version	現在サポートされているこのテンプレートのバージョン。

Drupal JSON スキーマ

```
{
 "$schema": "http://json-schema.org/draft-04/schema#",
 "type": "object",
 "properties": {
  "connectionConfiguration": {
   "type": "object",
   "properties": {
    "repositoryEndpointMetadata": {
     "type": "object",
     "properties": {
```

```
      "hostUrl": {
       "type": "string",
       "pattern": "https:.*"
      }
     },
     "required": [
      "hostUrl"
     ]
    }
   },
   "required": [
    "repositoryEndpointMetadata"
   ]
  },
  "repositoryConfigurations": {
   "type": "object",
   "properties": {
    "content": {
     "type": "object",
     "properties": {
      "fieldMappings": {
       "type": "array",
       "items": [
        {
         "type": "object",
         "properties": {
          "indexFieldName": {
           "type": "string"
          },
          "indexFieldType": {
           "type": "string",
           "enum": [
            "STRING",
            "DATE"
           ]
          },
          "dataSourceFieldName": {
           "type": "string"
          },
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
```

```
        "required": [
         "indexFieldName",
         "indexFieldType",
         "dataSourceFieldName"
        ]
       }
      ]
     }
    },
    "required": [
     "fieldMappings"
    ]
   },
   "comment": {
    "type": "object",
    "properties": {
     "fieldMappings": {
      "type": "array",
      "items": [
       {
        "type": "object",
        "properties": {
         "indexFieldName": {
          "type": "string"
         },
         "indexFieldType": {
          "type": "string",
          "enum": [
           "STRING",
           "DATE"
          ]
         },
         "dataSourceFieldName": {
          "type": "string"
         },
         "dateFieldFormat": {
          "type": "string",
          "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
         }
        },
        "required": [
         "indexFieldName",
         "indexFieldType",
         "dataSourceFieldName"
```

```
        ]
       }
      ]
     }
    },
    "required": [
     "fieldMappings"
    ]
   },
   "attachment": {
    "type": "object",
    "properties": {
     "fieldMappings": {
      "type": "array",
      "items": [
       {
        "type": "object",
        "properties": {
         "indexFieldName": {
          "type": "string"
         },
         "indexFieldType": {
          "type": "string",
          "enum": [
           "STRING",
           "DATE"
          ]
         },
         "dataSourceFieldName": {
          "type": "string"
         },
         "dateFieldFormat": {
          "type": "string",
          "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
         }
        },
        "required": [
         "indexFieldName",
         "indexFieldType",
         "dataSourceFieldName"
        ]
       }
      ]
     }
```

```
     },
     "required": [
      "fieldMappings"
     ]
    }
   }
  },
  "additionalProperties": {
   "type": "object",
   "properties": {
    "isCrawlArticle": {
     "type": "boolean"
    },
    "isCrawlBasicPage": {
     "type": "boolean"
    },
    "isCrawlBasicBlock": {
     "type": "boolean"
    },
    "crawlCustomContentTypesList": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "crawlCustomBlockTypesList": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "filePath": {
     "anyOf": [
      {
       "type": "string",
       "pattern": "s3:.*"
      },
      {
       "type": "string",
       "pattern": ""
      }
     ]
    },
    "inclusionFileNamePatterns": {
```

```
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "exclusionFileNamePatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "articleTitleInclusionPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "articleTitleExclusionPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "pageTitleInclusionPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "pageTitleExclusionPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "customContentTitleInclusionPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "customContentTitleExclusionPatterns": {
      "type": "array",
      "items": {
```

```
      "type": "string"
     }
    },
    "basicBlockTitleInclusionPatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "basicBlockTitleExclusionPatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "customBlockTitleInclusionPatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "customBlockTitleExclusionPatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "contentDefinitions": {
     "type": "array",
     "items": {
      "properties": {
       "contentType": {
        "type": "string"
       },
       "fieldDefinition": {
        "type": "array",
        "items": [
         {
          "type": "object",
          "properties": {
           "machineName": {
            "type": "string"
           },
           "type": {
```

```
            "type": "string"
           }
          },
          "required": [
           "machineName",
           "type"
          ]
         }
        ]
       },
       "isCrawlComments": {
        "type": "boolean"
       },
       "isCrawlFiles": {
        "type": "boolean"
       }
      }
     },
     "required": [
      "contentType",
      "fieldDefinition",
      "isCrawlComments",
      "isCrawlFiles"
     ]
    }
   },
   "required": []
  },
  "type": {
   "type": "string",
   "pattern": "DRUPAL"
  },
  "authType": {
   "type": "string",
   "enum": [
    "BASIC-AUTH",
    "OAUTH2"
   ]
  },
  "syncMode": {
   "type": "string",
   "enum": [
    "FORCED_FULL_CRAWL",
    "FULL_CRAWL",
```

```
    "CHANGE_LOG"
   ]
  },
  "enableIdentityCrawler": {
   "type": "boolean"
  },
  "secretArn": {
   "type": "string",
   "minLength": 20,
   "maxLength": 2048
  }
 },
 "version": {
  "type": "string",
  "anyOf": [
   {
    "pattern": "1.0.0"
   }
  ]
 },
 "required": [
  "connectionConfiguration",
  "repositoryConfigurations",
  "syncMode",
  "additionalProperties",
  "secretArn",
  "type"
 ]
}
```

GitHub テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定やリポジトリエンドポイントの詳細の一部として、 GitHub ホスト URL、組織名、 GitHub GitHub クラウドとオンプレミスのどちらを使用するかを指定します。また、データソースのタイプを GITHUB に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TypeCreateDataSource呼び出し時にを指定しますTEMPLATE。

このデベロッパーガイドで提供されているテンプレートを使用できます。GitHub JSON スキーマ を参照してください。

次の表では、 GitHub JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
type	SAASタイプをまたはとして指定しますON_PREMISE 。
hostUrl	GitHub ホスト URL。たとえば、 GitHub SaaS/エンタープライズクラウドを使用している場合:. https://api.github.com または、 GitHubオンプレミス/エンタープライズサーバーを使用している場合:。https://on-prem-host-url/api/v3/
organizationName	GitHub デスクトップにログインして [プロフィール写真] ドロップダウンの [組織] に移動すると、組織名を確認できます。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• GHRepository • Hコミット • ghIssueDocument • ghIssueComment • ghIssueAttachment • GPR ドキュメント • GHPR コメント • GHPR アタッチメント	GitHub Amazon Kendra コンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
isCrawlAcl	trueACL があって、それをアクセス制御に使用したい場合に、文書のアクセス制御リスト

構成	説明
	(ACL) 情報をクロールします。ACL は、ユーザーとグループがアクセスして検索できるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
fieldForUserID	ACL クロールに使用するユーザ ID の種類を指定します。ユーザー ID にユーザーの電子メールを使用するかusername、ユーザー ID email にユーザー名を使用するかを指定します。オプションを指定しない場合、emailがデフォルトで使用されます。
レポジトリフィルタ	インデックスを作成したい特定のリポジトリ名とブランチ名のリスト。
Crawl/リポジトリ	trueリポジトリをクロールします。
crawlRepositoryDocuments	trueリポジトリドキュメントをクロールします。
クロール/イシュー	true課題をクロールします。
crawlIssueComment	true課題のコメントをクロールします。
crawlIssueComment添付ファイル	true課題コメントの添付ファイルをクロールします。
crawlPullRequest	trueプルリクエストをクロールします。
crawlPullRequest[コメント]	trueプルリクエストのコメントをクロールします。
crawlPullRequestCommentAttachment	trueプルリクエストのコメント添付ファイルをクロールします。

構成	説明
• inclusionFolderNameパターン • inclusionFileTypeパターン • inclusionFileNameパターン	GitHubデータソースに特定のコンテンツを含めるための正規表現パターンのリスト。パターンに一致するコンテンツは、インデックスに含まれます。パターンに一致しないコンテンツは、インデックスから除外されます。包含パターンと除外パターンの両方に一致するコンテンツがある場合、その除外パターンが優先され、コンテンツはインデックスに含まれません。
• exclusionFolderNameパターン • exclusionFileTypeパターン • exclusionFileNameパターン	GitHubデータソース内の特定のコンテンツを除外するための正規表現パターンのリスト。パターンに一致するコンテンツは、インデックスから除外されます。パターンに一致しないコンテンツは、インデックスに含まれます。包含パターンと除外パターンの両方に一致するコンテンツがある場合、その除外パターンが優先され、コンテンツはインデックスに含まれません。
type	データソースのタイプ。データソースタイプとして `GITHUB` を指定します。
enableIdentityCrawler	`true` Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID 情報/プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。 • データソースがインデックスと同期されるたびに、すべてのコンテンツを再クロールしたりFORCED_FULL_CRAWL 、既存のコンテンツを置き換えたりできます。 • FULL_CRAWL データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールできます。 • CHANGE_LOG データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールする場合に使用します。
secretArn	AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。 GitHubシークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "personalToken": " token" }`
version	このテンプレートの現在サポートされているバージョン。

GitHub JSON スキーマ

GitHub JSON スキーマは次のとおりです。

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "type": "object",
    "properties": {
        "connectionConfiguration": {
            "type": "object",
            "properties": {
                "repositoryEndpointMetadata": {
                    "type": "object",
                    "properties": {
                        "type": {
                            "type": "string"
                        },
                        "hostUrl": {
                            "type": "string",
                            "pattern": "https://.*"
                        },
                        "organizationName": {
                            "type": "string"
                        }
                    },
                    "required": [
                        "type",
                        "hostUrl",
                        "organizationName"
                    ]
                }
            },
            "required": [
                "repositoryEndpointMetadata"
            ]
        },
        "repositoryConfigurations": {
            "type": "object",
            "properties": {
                "ghRepository": {
                    "type": "object",
                    "properties": {
                        "fieldMappings": {
                            "type": "array",
                            "items": [
                                {
                                    "type": "object",
```

```
                                "properties": {
                                    "indexFieldName": {
                                        "type": "string"
                                    },
                                    "indexFieldType": {
                                        "type": "string",
                                        "enum": [
                                            "STRING",
                                            "STRING_LIST",
                                            "DATE"
                                        ]
                                    },
                                    "dataSourceFieldName": {
                                        "type": "string"
                                    },
                                    "dateFieldFormat": {
                                        "type": "string",
                                        "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                    }
                                },
                                "required": [
                                    "indexFieldName",
                                    "indexFieldType",
                                    "dataSourceFieldName"
                                ]
                            }
                        ]
                    }
                },
                "required": [
                    "fieldMappings"
                ]
            },
            "ghCommit": {
                "type": "object",
                "properties": {
                    "fieldMappings": {
                        "type": "array",
                        "items": [
                            {
                                "type": "object",
                                "properties": {
                                    "indexFieldName": {
                                        "type": "string"
```

```
                                    },
                                    "indexFieldType": {
                                        "type": "string",
                                        "enum": [
                                            "STRING",
                                            "STRING_LIST",
                                            "DATE"
                                        ]
                                    },
                                    "dataSourceFieldName": {
                                        "type": "string"
                                    },
                                    "dateFieldFormat": {
                                        "type": "string",
                                        "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                    }
                                },
                                "required": [
                                    "indexFieldName",
                                    "indexFieldType",
                                    "dataSourceFieldName"
                                ]
                            }
                        ]
                    }
                },
                "required": [
                    "fieldMappings"
                ]
            },
            "ghIssueDocument": {
                "type": "object",
                "properties": {
                    "fieldMappings": {
                        "type": "array",
                        "items": [
                            {
                                "type": "object",
                                "properties": {
                                    "indexFieldName": {
                                        "type": "string"
                                    },
                                    "indexFieldType": {
                                        "type": "string",
```

```
                                    "enum": [
                                        "STRING",
                                        "STRING_LIST",
                                        "DATE"
                                    ]
                                },
                                "dataSourceFieldName": {
                                    "type": "string"
                                },
                                "dateFieldFormat": {
                                    "type": "string",
                                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                }
                            },
                            "required": [
                                "indexFieldName",
                                "indexFieldType",
                                "dataSourceFieldName"
                            ]
                        }
                    ]
                }
            },
            "required": [
                "fieldMappings"
            ]
        },
        "ghIssueComment": {
            "type": "object",
            "properties": {
                "fieldMappings": {
                    "type": "array",
                    "items": [
                        {
                            "type": "object",
                            "properties": {
                                "indexFieldName": {
                                    "type": "string"
                                },
                                "indexFieldType": {
                                    "type": "string",
                                    "enum": [
                                        "STRING",
                                        "STRING_LIST",
```

```
                                    "DATE"
                                ]
                            },
                            "dataSourceFieldName": {
                                "type": "string"
                            },
                            "dateFieldFormat": {
                                "type": "string",
                                "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                            }
                        },
                        "required": [
                            "indexFieldName",
                            "indexFieldType",
                            "dataSourceFieldName"
                        ]
                    }
                ]
            }
        },
        "required": [
            "fieldMappings"
        ]
    },
    "ghIssueAttachment": {
        "type": "object",
        "properties": {
            "fieldMappings": {
                "type": "array",
                "items": [
                    {
                        "type": "object",
                        "properties": {
                            "indexFieldName": {
                                "type": "string"
                            },
                            "indexFieldType": {
                                "type": "string",
                                "enum": [
                                    "STRING",
                                    "STRING_LIST",
                                    "DATE"
                                ]
                            },
```

```
                                "dataSourceFieldName": {
                                    "type": "string"
                                },
                                "dateFieldFormat": {
                                    "type": "string",
                                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                }
                            },
                            "required": [
                                "indexFieldName",
                                "indexFieldType",
                                "dataSourceFieldName"
                            ]
                        }
                    ]
                }
            },
            "required": [
                "fieldMappings"
            ]
        },
        "ghPRDocument": {
            "type": "object",
            "properties": {
                "fieldMappings": {
                    "type": "array",
                    "items": [
                        {
                            "type": "object",
                            "properties": {
                                "indexFieldName": {
                                    "type": "string"
                                },
                                "indexFieldType": {
                                    "type": "string",
                                    "enum": [
                                        "STRING",
                                        "STRING_LIST",
                                        "DATE"
                                    ]
                                },
                                "dataSourceFieldName": {
                                    "type": "string"
                                },
```

```
                                "dateFieldFormat": {
                                    "type": "string",
                                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                }
                            },
                            "required": [
                                "indexFieldName",
                                "indexFieldType",
                                "dataSourceFieldName"
                            ]
                        }
                    ]
                }
            },
            "required": [
                "fieldMappings"
            ]
        },
        "ghPRComment": {
            "type": "object",
            "properties": {
                "fieldMappings": {
                    "type": "array",
                    "items": [
                        {
                            "type": "object",
                            "properties": {
                                "indexFieldName": {
                                    "type": "string"
                                },
                                "indexFieldType": {
                                    "type": "string",
                                    "enum": [
                                        "STRING",
                                        "STRING_LIST",
                                        "DATE"
                                    ]
                                },
                                "dataSourceFieldName": {
                                    "type": "string"
                                },
                                "dateFieldFormat": {
                                    "type": "string",
                                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
```

```
                                }
                            },
                            "required": [
                                "indexFieldName",
                                "indexFieldType",
                                "dataSourceFieldName"
                            ]
                        }
                    ]
                }
            },
            "required": [
                "fieldMappings"
            ]
        },
        "ghPRAttachment": {
            "type": "object",
            "properties": {
                "fieldMappings": {
                    "type": "array",
                    "items": [
                        {
                            "type": "object",
                            "properties": {
                                "indexFieldName": {
                                    "type": "string"
                                },
                                "indexFieldType": {
                                    "type": "string",
                                    "enum": [
                                        "STRING",
                                        "STRING_LIST",
                                        "DATE"
                                    ]
                                },
                                "dataSourceFieldName": {
                                    "type": "string"
                                },
                                "dateFieldFormat": {
                                    "type": "string",
                                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                                }
                            },
                            "required": [
```

```
                                    "indexFieldName",
                                    "indexFieldType",
                                    "dataSourceFieldName"
                                ]
                            }
                        ]
                    }
                },
                "required": [
                    "fieldMappings"
                ]
            }
        }
    },
    "additionalProperties": {
        "type": "object",
        "properties": {
            "isCrawlAcl": {
                "type": "boolean"
            },
            "fieldForUserId": {
                "type": "string"
            },
            "crawlRepository": {
                "type": "boolean"
            },
            "crawlRepositoryDocuments": {
                "type": "boolean"
            },
            "crawlIssue": {
                "type": "boolean"
            },
            "crawlIssueComment": {
                "type": "boolean"
            },
            "crawlIssueCommentAttachment": {
                "type": "boolean"
            },
            "crawlPullRequest": {
                "type": "boolean"
            },
            "crawlPullRequestComment": {
                "type": "boolean"
            },
```

```
                "crawlPullRequestCommentAttachment": {
                    "type": "boolean"
                },
                "repositoryFilter": {
                    "type": "array",
                    "items": [
                        {
                            "type": "object",
                            "properties": {
                                "repositoryName": {
                                    "type": "string"
                                },
                                "branchNameList": {
                                    "type": "array",
                                    "items": {
                                        "type": "string"
                                    }
                                }
                            }
                        }
                    ]
                },
                "inclusionFolderNamePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                },
                "inclusionFileTypePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                },
                "inclusionFileNamePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                },
                "exclusionFolderNamePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
```

```
                    }
                },
                "exclusionFileTypePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                },
                "exclusionFileNamePatterns": {
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                }
            },
            "required": []
        },
        "type": {
            "type": "string",
            "pattern": "GITHUB"
        },
        "syncMode": {
            "type": "string",
            "enum": [
                "FULL_CRAWL",
                "FORCED_FULL_CRAWL",
                "CHANGE_LOG"
            ]
        },
        "enableIdentityCrawler": {
            "type": "boolean"
        },
        "secretArn": {
            "type": "string",
            "minLength": 20,
            "maxLength": 2048
        }
    },
    "version": {
        "type": "string",
        "anyOf": [
            {
                "pattern": "1.0.0"
            }
```

```
        ]
    },
    "required": [
        "connectionConfiguration",
        "repositoryConfigurations",
        "syncMode",
        "additionalProperties",
        "enableIdentityCrawler"
    ]
}
```

Gmail テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを GMAIL に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Gmail JSON スキーマ を参照してください。

次の表では、Gmail JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
• message • 添付ファイル	Gmail Amazon Kendra のメッセージと添付ファイルの属性やフィールド名をインデックスフィールド名にマッピングするオブジェクトのリストです。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。

構成	説明
• inclusionLabelNameパターン • exclusionLabelNameパターン • inclusionAttachmentTypeパターン • exclusionAttachmentTypeパターン • inclusionAttachmentNameパターン • exclusionAttachmentNameパターン • inclusionSubjectFilter • exclusionSubjectFilter • isSubjectAnd • inclusionFromFilter • exclusionFromFilter • inclusionToFilter • exclusionToFilter • inclusionCcFilter • exclusionCcFilter • inclusionBccFilter • exclusionBccFilter	Gmail データソースにある特定の件名のメールを含めるまたは除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
beforeDateFilter	特定の日付より前に含めるメッセージと添付ファイルを指定します。
afterDateFilter	特定の日付以降に含めるメッセージと添付ファイルを指定します。
isCrawlAttachment	添付ファイルをクロールするかどうかを選択するブール値。メッセージは自動的にクロールされます。
type	データソースのタイプ。データソースタイプとして `GMAIL` を指定します。

構成	説明
shouldCrawlDraftメッセージ	ドラフトメッセージをクロールするかどうかを選択するブール値。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 **Important** 完全に削除された Gmail メッセージを更新する API がないため、[新規、変更、削除したコンテンツの同期] は以下のようになります。 • Gmail から完全に削除されたメッセージは Amazon Kendra インデックスから削除されません • Gmail のメールラベルの変更は同期されません。 Gmail Amazon Kendra のデータソースラベルの変更や完全に削除されたメールをインデックスに同期するには、定期的にフルクロールを実行する必要があります。

構成	説明

secretARN	Gmail への接続に必要なキーと値のペアが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。
	
	{
	"adminAccountEmailId": " *service account email*",
	"clientEmailId": " *user account email*",
	"privateKey": " *private key*"
	}

version	現在サポートされているテンプレートのバージョン。

Gmail JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
      }
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "message": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
```

```
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING", "STRING_LIST", "DATE"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        }
      },
      "attachments": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
```

```
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        }
      },
      "required": []
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "inclusionLabelNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionLabelNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionAttachmentTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionAttachmentTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionAttachmentNamePatterns": {
          "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "exclusionAttachmentNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionSubjectFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionSubjectFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "isSubjectAnd": {
          "type": "boolean"
        },
        "inclusionFromFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFromFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionToFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionToFilter": {
```

```
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionCcFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionCcFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionBccFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionBccFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "beforeDateFilter": {
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "afterDateFilter": {
          "anyOf": [
            {
```

```
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "isCrawlAttachment": {
          "type": "boolean"
        },
        "shouldCrawlDraftMessages": {
          "type": "boolean"
        }
      },
      "required": [
        "isCrawlAttachment",
        "shouldCrawlDraftMessages"
      ]
    },
    "type" : {
      "type" : "string",
      "pattern": "GMAIL"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL"
      ]
    },
    "secretArn": {
      "type": "string"
    },
    "version": {
      "type": "string",
      "anyOf": [
        {
          "pattern": "1.0.0"
        }
      ]
    }
  },
```

```
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "syncMode",
    "secretArn",
    "type"
  ]
}
```

Google Drive テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを GOOGLEDRIVE2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Google Drive JSON スキーマ を参照してください。

次の表では、Google ドライブ JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースに関する設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。このデータソースはエンドポイントを指定していません。認証タイプを選択します。serviceAccount と OAuth2。接続情報は、 AWS Secrets Manager 指定したシークレットに含まれますsecretArn 。
authType	ユースケースに基づいて serviceAccount と OAuth2 のどちらかを選んでください。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。

構成	説明
• file • コメント	Google Drive の属性またはフィールド名を Amazon Kendra インデックスフィールド名にマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
• maxFileSizeInMegaBytes	クロールするファイルサイズの制限を MB 単位で指定します。 Amazon Kendra
• iscrawlComment	`true`Google ドライブのデータソース内のコメントをクロールします。
• isCrawlMyDriveAndSharedWithMe	`true`Google ドライブのデータソース内の Shared With Me MyDrive ドライブをクロールして共有する。
• isCrawlSharedドライブ	`true`Google ドライブのデータソース内の共有ドライブをクロールします。
isCrawlAcl	`true`ACL があり、それをアクセス制御に使用したい場合に、ドキュメントのアクセス制御リスト (ACL) 情報をクロールします。ACL は、ユーザーとグループがアクセスして検索できるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

構成	説明
• excludeUserAccounts • excludeSharedDrives • excludeMimeTypes • exclusionFileTypeパターン • exclusionFileNameパターン • exclusionFilePathフィルター	Google Drive データソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• includeUserAccounts • includeSharedDrives • includeMimeTypes • inclusionFileTypeパターン • inclusionFileNameパターン • inclusionFilePathフィルター	Google Drive データソースにある特定のファイルを含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
type	データソースのタイプ。データソースタイプとして `GOOGLEDRIVEV2` を指定します。
enableIdentityCrawler	`true` Amazon KendraのIDクローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループのID/プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明

secretARN	Google AWS Secrets Manager ドライブへの接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム（ARN）。シークレットには、次のキーを持つ JSON 構造を含める必要があります。
	
	Google サービスアカウント認証を使用している場合。
	
	`{ "clientEmail": " user account email", "adminAccountEmail": " service account email", "privateKey": " private key" }`
	
	OAuth 2.0 認証を使用している場合。
	
	`{ "clientID": " OAuth client ID", "clientSecret": " client secret", "refreshToken": " refresh token" }`

version	現在サポートされているこのテンプレートのバージョン。

Google Drive JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
```

```
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "authType": {
              "type": "string",
              "enum": [
                "serviceAccount",
                "OAuth2"
              ]
            }
          },
          "required": [
            "authType"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "file": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST",
                        "LONG"
                      ]
                    },
```

```
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "comment": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE",
                    "STRING_LIST"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
```

```
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "maxFileSizeInMegaBytes": {
        "type": "string"
      },
      "isCrawlComment": {
        "type": "boolean"
      },
      "isCrawlMyDriveAndSharedWithMe": {
        "type": "boolean"
      },
      "isCrawlSharedDrives": {
        "type": "boolean"
      },
      "isCrawlAcl": {
        "type": "boolean"
      },
      "excludeUserAccounts": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
```

```
        "excludeSharedDrives": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "excludeMimeTypes": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "includeUserAccounts": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "includeSharedDrives": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "includeMimeTypes": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "includeTargetAudienceGroup": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFileNamePatterns": {
          "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "exclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFilePathFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFilePathFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "type": {
      "type": "string",
      "pattern": "GOOGLEDRIVEV2"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
```

```
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

IBM DB2 テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを JDBC に指定し、データベースタイプを db2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。IBM DB2 JSON スキーマを参照してください。

次の表では、IBM DB2 JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。

構成	説明
	• DBType — 使用する Java データベースのタイプ。、、、mysqlまたはのいずれでもかまいませんdb2。postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。

構成	説明
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。

構成	説明
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして JDBC を指定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

IBM DB2 JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
```

```
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
```

```
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    },
    "required": [
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "primaryKey": {
        "type": "string"
      },
      "titleColumn": {
        "type": "string"
      },
      "bodyColumn": {
        "type": "string"
      },
      "sqlQuery": {
        "type": "string",
        "not": {
          "pattern": ";+"
        }
      },
      "timestampColumn": {
        "type": "string"
      },
      "timestampFormat": {
        "type": "string"
      },
      "timezone": {
        "type": "string"
      },
      "changeDetectingColumns": {
        "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
```

```
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Microsoft Exchange テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。テナント ID は、接続設定またはリポジトリエンドポイントの詳細の一部として指定します。また、データソースのタイプを MSEXCHANGE に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Microsoft Exchange JSON スキーマ を参照してください。

次の表では、Microsoft Exchange JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
tenantId	Microsoft 365 テナント ID。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• email • 添付 • カレンダー	Microsoft Exchange Amazon Kendra データソースの属性またはフィールド名をインデックスフィールドにマップするオブジェクトのリス

構成	説明
• contacts • 注意事項	ト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
inclusionPatterns	Microsoft Exchange のデータソースにある特定のファイルを含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
exclusionPatterns	Microsoft Exchange のデータソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• inclusionUsersList • inclusionUsersFile[名前] • inclusionDomainUsers	Microsoft Exchange のデータソースにある特定のユーザーおよびユーザーファイルを含めるための正規表現のパターンのリスト。パターンに一致するユーザーは、インデックスに含まれます。パターンに一致しないユーザーは、インデックスから除外されます。ユーザーが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのユーザーはインデックスに含まれません。

構成	説明
• exclusionUsersList • exclusionUsersFile[名前] • exclusionDomainUsers	Microsoft Exchange のデータソースにある特定のユーザーおよびユーザーファイルを除外するための正規表現のパターンのリスト。パターンに一致するユーザーは、インデックスから除外されます。パターンに一致しないユーザーは、インデックスに含まれます。ユーザーが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのユーザーはインデックスに含まれません。
s3bucketName	S3 バケットの名前 (使用する場合)。
• crawlCalendar • crawlNotes • crawlContacts • crawlFolderAcl	`true`このような種類のコンテンツやアクセス制御情報を Microsoft Exchange データソースからクロールします。
startCalendarDate時間	カレンダーのコンテンツには特定の開始日時を設定できます。
endCalendarDate時間	カレンダーのコンテンツには特定の終了日時を設定できます。
subject	メールコンテンツには特定の件名を設定できます。
emailFrom	「差出人」または送信者のメールコンテンツに特定のメールを設定できます。
emailTo	「宛先」または受信者のメールコンテンツに特定のメールを設定できます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
type	データソースのタイプ。データソースタイプとして MSEXCHANGE を指定します。
secretARN	Microsoft Exchange AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットのAmazon リソースネーム (ARN)。これには、Azure Portal で OAuth アプリケーションを作成したときに生成されるクライアント ID とクライアントシークレットが含まれます。
version	現在サポートされているこのテンプレートのバージョン。

Microsoft Exchange JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "tenantId": {
              "type": "string",
              "pattern": "^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]
{12}$",
              "minLength": 36,
              "maxLength": 36
            }
          },
          "required": ["tenantId"]
        }
      }
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "email": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "DATE"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
```

```
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "attachment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING", "DATE","LONG"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
```

```
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "calendar": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING", "STRING_LIST", "DATE"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
```

```
          ]
        },
        "contacts": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "DATE"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "notes": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
```

```
            "items": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING", "DATE"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    },
    "required": ["email"
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "inclusionPatterns": {
        "type": "array",
        "items": {
          "type": "string"
        }
```

```
        },
        "exclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionUsersList": {
          "type": "array",
          "items": {
            "type": "string",
            "format": "email"
          }
        },
        "exclusionUsersList": {
          "type": "array",
          "items": {
            "type": "string",
            "format": "email"
          }
        },
        "s3bucketName": {
          "type": "string"
        },
        "inclusionUsersFileName": {
          "type": "string"
        },
        "exclusionUsersFileName": {
          "type": "string"
        },
        "inclusionDomainUsers": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionDomainUsers": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "crawlCalendar": {
          "type": "boolean"
```

```
        },
        "crawlNotes": {
          "type": "boolean"
        },
        "crawlContacts": {
          "type": "boolean"
        },
        "crawlFolderAcl": {
          "type": "boolean"
        },
        "startCalendarDateTime": {
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "endCalendarDateTime": {
          "anyOf": [
            {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "subject": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "emailFrom": {
          "type": "array",
          "items": {
            "type": "string",
```

```
            "format": "email"
          }
        },
        "emailTo": {
          "type": "array",
          "items": {
            "type": "string",
            "format": "email"
          }
        }
      },
      "required": [
      ]
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "type" : {
      "type" : "string",
      "pattern": "MSEXCHANGE"
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
```

```
    "type"
  ]
}
```

Microsoft OneDrive テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。テナント ID は接続設定またはリポジトリエンドポイントの詳細の一部として指定します。また、データソースのタイプを ONEDRIVEV2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Microsoft OneDrive JSON スキーマ を参照してください。

次の表では、Microsoft OneDrive JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
tenantId	Microsoft 365 テナント ID。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
file	Microsoft OneDrive Amazon Kendra ファイルの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。

構成	説明
• userNameFilter • userFilterPath • inclusionFileTypeパターン • exclusionFileTypeパターン • inclusionFileNameパターン • exclusionFileNameパターン • inclusionFilePathパターン • exclusionFilePathパターン • inclusionOneNoteSectionNamePatterns • exclusionOneNoteSectionNamePatterns • inclusionOneNotePageNamePatterns • exclusionOneNotepageNamePatterns	特定のファイル、 OneNote セクション、 OneNote ページのインデックスを作成したり、ユーザー名でフィルターしたりできます。
isUserNameS3 で	Amazon S3に保存されているファイル内のユーザー名のリストを提供する場合は、`true` にします。
type	データソースのタイプ。データソースタイプとして `ONEDRIVEV2` を指定します。
enableIdentityCrawler	`true` Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーおよびグループの ID /プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。
type	データソースのタイプ。データソースタイプとして `ONEDRIVEV2` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretARN	Microsoft AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。 OneDriveシークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "clientId": " client ID", "clientSecret": " client secret" }`
version	現在サポートされているこのテンプレートのバージョン。

Microsoft OneDrive JSON スキーマ

```
{
```

```
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
 "connectionConfiguration": {
  "type": "object",
  "properties": {
   "repositoryEndpointMetadata": {
    "type": "object",
    "properties": {
     "tenantId": {
      "type": "string",
      "pattern": "^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$",
      "minLength": 36,
      "maxLength": 36
     }
    },
    "required": [
     "tenantId"
    ]
   }
  },
  "required": [
   "repositoryEndpointMetadata"
  ]
 },
 "repositoryConfigurations": {
  "type": "object",
  "properties": {
   "file": {
    "type": "object",
    "properties": {
     "fieldMappings": {
      "type": "array",
      "items": [
       {
        "type": "object",
        "properties": {
         "indexFieldName": {
          "type": "string"
         },
         "indexFieldType": {
          "type": "string",
          "enum": [
           "STRING",
```

```
            "STRING_LIST",
            "DATE",
            "LONG"
           ]
          },
          "dataSourceFieldName": {
           "type": "string"
          },
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
         "required": [
          "indexFieldName",
          "indexFieldType",
          "dataSourceFieldName"
         ]
        }
       ]
      }
     },
     "required": [
      "fieldMappings"
     ]
    }
   }
  },
  "additionalProperties": {
   "type": "object",
   "properties": {
    "userNameFilter": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "userFilterPath": {
     "type": "string"
    },
    "isUserNameOnS3": {
     "type": "boolean"
    },
    "inclusionFileTypePatterns": {
```

```
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "exclusionFileTypePatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "inclusionFileNamePatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "exclusionFileNamePatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "inclusionFilePathPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "exclusionFilePathPatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "inclusionOneNoteSectionNamePatterns": {
      "type": "array",
      "items": {
       "type": "string"
      }
     },
     "exclusionOneNoteSectionNamePatterns": {
      "type": "array",
      "items": {
```

```
      "type": "string"
     }
    },
    "inclusionOneNotePageNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "exclusionOneNotePageNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    }
   },
   "required": []
  },

  "enableIdentityCrawler": {
   "type": "boolean"
  },
  "type": {
   "type": "string",
   "pattern": "ONEDRIVEV2"
  },
  "syncMode": {
   "type": "string",
   "enum": [
    "FULL_CRAWL",
    "FORCED_FULL_CRAWL",
    "CHANGE_LOG"
   ]
  },
  "secretArn": {
   "type": "string",
   "minLength": 20,
   "maxLength": 2048
  }
 },
 "version": {
  "type": "string",
  "anyOf": [
   {
```

```
    "pattern": "1.0.0"
   }
  ]
 },
 "required": [
  "connectionConfiguration",
  "repositoryConfigurations",
  "syncMode",
  "additionalProperties",
  "secretArn",
  "type"
 ]
}
```

Microsoft SharePoint テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として、 SharePoint サイトの URL/URL、ドメイン、および必要に応じてテナント ID を指定します。また、データソースのタイプを SHAREPOINTV2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATE呼び出し時にタイプとして指定します。CreateDataSource

このデベロッパーガイドで提供されているテンプレートを使用できます。SharePoint JSON スキーマ を参照してください。

次の表では、Microsoft SharePoint JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
tenantId	SharePoint アカウントのテナント ID。
ドメイン	SharePoint アカウントのドメイン。
siteUrls	SharePoint アカウントのホスト URL。
repositoryAdditionalProperties	リポジトリ/データソースエンドポイントに接続するための追加プロパティ。

構成	説明
s3bucketName	Azure AD の自己署名 X.509 Amazon S3 証明書を格納するバケットの名前。
s3certificateName	バケットに保存されている Azure AD 自己署名 X.509 証明書の名前。 Amazon S3
authType	使用する認証のタイプ (、、、、、、OAuth2、OAuth2Certificate OAuth2App Basic、OAuth2_RefreshToken など)。NTLM Kerberos
version	SharePoint 使用するバージョン (Serverまたはにかかわらず) Online。
onPremVersion	、、、など201320162019、 SharePoint 使用しているサーバーのバージョンSubscriptionEdition 。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• イベント • ページで • file • link (リンク) • 添付 • コメント	SharePoint Amazon Kendra コンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。

構成	説明
• eventTitleFilterRegEx • pageTitleFilterRegEx • linkTitleFilterRegEx • inclusionFilePath • exclusionFilePath • inclusionFileTypeパターン • exclusionFileTypeパターン • inclusionFileNameパターン • exclusionFileNameパターン • inclusionOneNoteSectionNamePatterns • exclusionOneNoteSectionNamePatterns • inclusionOneNotePageNamePatterns • exclusionOneNotePageNamePatterns	データソースに特定のコンテンツを含めたり除外したりする正規表現パターンのリスト。SharePoint インクルージョンパターンに一致するコンテンツアイテムがインデックスに含まれます。インクルージョンパターンに一致しないコンテンツアイテムはインデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• crawlFiles • crawlPages • crawlEvents • crawlComments • crawlLinks • crawlAttachments	`true`このような種類のコンテンツをクロールします。
crawlAcl	`true`ドキュメントのアクセス制御リスト (ACL) 情報をクロールすること。ACL があって、それをアクセス制御に使用したい場合です。ACL は、ユーザーとグループがアクセスして検索できるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

構成	説明
fieldForUserID	ユーザー ID にユーザーの電子メールを使用するかuserPrincipalName 、ユーザー ID email にユーザー名を使用するかを指定します。オプションを指定しない場合、emailがデフォルトで使用されます。
aclConfiguration	ACLWithLDAPEmailFmt 、ACLWithManualEmailFmt 、またはのいずれかを指定しますACLWithUsernameFmtM 。
emailDomain	E メールのドメイン。例: *"amazon.com"* 。 | • isCrawlLocalGroupMapping • isCrawlAdGroupMapping	trueグループマッピング情報をクロールします。
proxyHost	使用する Web プロキシのホスト名。http://または https://プロトコルは除きます。
proxyPort	ホスト URL トランスポートプロトコルが使用するポート番号。これは 0～65535 の範囲の値にする必要があります。
type	データソースタイプとして SHAREPOINTV2 を指定します。
enableIdentityCrawler	true Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーおよびグループの ID /プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretARN	AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。 SharePointこれらのキーと値のペアについては、「Online と Server の接続手順」を参照してください。 SharePoint SharePoint
version	現在サポートされているこのテンプレートのバージョン。

SharePoint JSON スキーマ

```
{
 "$schema": "http://json-schema.org/draft-04/schema#",
 "type": "object",
 "properties": {
  "connectionConfiguration": {
```

```
    "type": "object",
    "properties": {
     "repositoryEndpointMetadata": {
      "type": "object",
      "properties": {
       "tenantId": {
        "type": "string",
        "pattern": "^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$",
        "minLength": 36,
        "maxLength": 36
       },
       "domain": {
        "type": "string"
       },
       "siteUrls": {
        "type": "array",
        "items": {
         "type": "string",
         "pattern": "https://.*"
        }
       },
       "repositoryAdditionalProperties": {
        "type": "object",
        "properties": {
         "s3bucketName": {
          "type": "string"
         },
         "s3certificateName": {
          "type": "string"
         },
         "authType": {
          "type": "string",
          "enum": [
           "OAuth2",
           "OAuth2Certificate",
           "OAuth2App",
           "Basic",
           "OAuth2_RefreshToken",
           "NTLM",
           "Kerberos"
          ]
         },
         "version": {
          "type": "string",
```

```
          "enum": [
           "Server",
           "Online"
          ]
         },
         "onPremVersion": {
          "type": "string",
          "enum": [
           "",
           "2013",
           "2016",
           "2019",
           "SubscriptionEdition"
          ]
         }
        },
        "required": [
         "authType",
         "version"
        ]
       }
      },
      "required": [
       "siteUrls",
       "domain",
       "repositoryAdditionalProperties"
      ]
     }
    },
    "required": [
     "repositoryEndpointMetadata"
    ]
   },
   "repositoryConfigurations": {
    "type": "object",
    "properties": {
     "event": {
      "type": "object",
      "properties": {
       "fieldMappings": {
        "type": "array",
        "items": [
         {
          "type": "object",
```

```
        "properties": {
         "indexFieldName": {
          "type": "string"
         },
         "indexFieldType": {
          "type": "string",
          "enum": [
           "STRING",
           "STRING_LIST",
           "DATE"
          ]
         },
         "dataSourceFieldName": {
          "type": "string"
         },
         "dateFieldFormat": {
          "type": "string",
          "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
         }
        },
        "required": [
         "indexFieldName",
         "indexFieldType",
         "dataSourceFieldName"
        ]
       }
      ]
     }
    },
    "required": [
     "fieldMappings"
    ]
   },
   "page": {
    "type": "object",
    "properties": {
     "fieldMappings": {
      "type": "array",
      "items": [
       {
        "type": "object",
        "properties": {
         "indexFieldName": {
          "type": "string"
```

```
          },
          "indexFieldType": {
           "type": "string",
           "enum": [
            "STRING",
            "DATE",
            "LONG"
           ]
          },
          "dataSourceFieldName": {
           "type": "string"
          },
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
         "required": [
          "indexFieldName",
          "indexFieldType",
          "dataSourceFieldName"
         ]
        }
       ]
      }
     },
     "required": [
      "fieldMappings"
     ]
    },
    "file": {
     "type": "object",
     "properties": {
      "fieldMappings": {
       "type": "array",
       "items": [
        {
         "type": "object",
         "properties": {
          "indexFieldName": {
           "type": "string"
          },
          "indexFieldType": {
           "type": "string",
```

```
          "enum": [
           "STRING",
           "DATE",
           "LONG"
          ]
         },
         "dataSourceFieldName": {
          "type": "string"
         },
         "dateFieldFormat": {
          "type": "string",
          "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
         }
        },
        "required": [
         "indexFieldName",
         "indexFieldType",
         "dataSourceFieldName"
        ]
       }
      ]
     }
    },
    "required": [
     "fieldMappings"
    ]
   },
   "link": {
    "type": "object",
    "properties": {
     "fieldMappings": {
      "type": "array",
      "items": [
       {
        "type": "object",
        "properties": {
         "indexFieldName": {
          "type": "string"
         },
         "indexFieldType": {
          "type": "string",
          "enum": [
           "STRING",
           "STRING_LIST",
```

```
            "DATE"
           ]
          },
          "dataSourceFieldName": {
           "type": "string"
          },
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
         "required": [
          "indexFieldName",
          "indexFieldType",
          "dataSourceFieldName"
         ]
        }
       ]
      }
     },
     "required": [
      "fieldMappings"
     ]
    },
    "attachment": {
     "type": "object",
     "properties": {
      "fieldMappings": {
       "type": "array",
       "items": [
        {
         "type": "object",
         "properties": {
          "indexFieldName": {
           "type": "string"
          },
          "indexFieldType": {
           "type": "string",
           "enum": [
            "STRING",
            "STRING_LIST",
            "DATE"
           ]
          },
```

```
          "dataSourceFieldName": {
           "type": "string"
          },
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
         "required": [
          "indexFieldName",
          "indexFieldType",
          "dataSourceFieldName"
         ]
        }
       ]
      }
     },
     "required": [
      "fieldMappings"
     ]
    },
    "comment": {
     "type": "object",
     "properties": {
      "fieldMappings": {
       "type": "array",
       "items": [
        {
         "type": "object",
         "properties": {
          "indexFieldName": {
           "type": "string"
          },
          "indexFieldType": {
           "type": "string",
           "enum": [
            "STRING",
            "STRING_LIST",
            "DATE"
           ]
          },
          "dataSourceFieldName": {
           "type": "string"
          },
```

```
          "dateFieldFormat": {
           "type": "string",
           "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
          }
         },
         "required": [
          "indexFieldName",
          "indexFieldType",
          "dataSourceFieldName"
         ]
        }
       ]
      }
     },
     "required": [
      "fieldMappings"
     ]
    }
   }
  },
  "additionalProperties": {
   "type": "object",
   "properties": {
    "eventTitleFilterRegEx": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "pageTitleFilterRegEx": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "linkTitleFilterRegEx": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "inclusionFilePath": {
     "type": "array",
     "items": {
```

```
      "type": "string"
     }
    },
    "exclusionFilePath": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "inclusionFileTypePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "exclusionFileTypePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "inclusionFileNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "exclusionFileNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "inclusionOneNoteSectionNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "exclusionOneNoteSectionNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
```

```
    },
    "inclusionOneNotePageNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "exclusionOneNotePageNamePatterns": {
     "type": "array",
     "items": {
      "type": "string"
     }
    },
    "crawlFiles": {
     "type": "boolean"
    },
    "crawlPages": {
     "type": "boolean"
    },
    "crawlEvents": {
     "type": "boolean"
    },
    "crawlComments": {
     "type": "boolean"
    },
    "crawlLinks": {
     "type": "boolean"
    },
    "crawlAttachments": {
     "type": "boolean"
    },
    "crawlListData": {
     "type": "boolean"
    },
    "crawlAcl": {
     "type": "boolean"
    },
    "fieldForUserId": {
     "type": "string"
    },
    "aclConfiguration": {
     "type": "string",
     "enum": [
      "ACLWithLDAPEmailFmt",
```

```
      "ACLWithManualEmailFmt",
      "ACLWithUsernameFmt"
     ]
    },
    "emailDomain": {
     "type": "string"
    },
    "isCrawlLocalGroupMapping": {
     "type": "boolean"
    },
    "isCrawlAdGroupMapping": {
     "type": "boolean"
    },
    "proxyHost": {
     "type": "string"
    },
    "proxyPort": {
     "type": "string"
    }
   },
   "required": [
   ]
  },
  "type": {
   "type": "string",
   "pattern": "SHAREPOINTV2"
  },
  "enableIdentityCrawler": {
   "type": "boolean"
  },
  "syncMode": {
   "type": "string",
   "enum": [
    "FULL_CRAWL",
    "FORCED_FULL_CRAWL",
    "CHANGE_LOG"
   ]
  },
  "secretArn": {
   "type": "string",
   "minLength": 20,
   "maxLength": 2048
  }
 },
```

```
 "version": {
  "type": "string",
  "anyOf": [
   {
    "pattern": "1.0.0"
   }
  ]
 },
 "required": [
  "connectionConfiguration",
  "repositoryConfigurations",
  "enableIdentityCrawler",
  "syncMode",
  "additionalProperties",
  "secretArn",
  "type"
 ]
}
```

Microsoft SQL サーバーテンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを sqlserver に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Microsoft SQL サーバー JSON スキーマ を参照してください。

次の表では、マイクロソフト SQL Server JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、、、mysqlのいずれでもかまいません。db2 postgresql oracle sqlserver

構成	説明
	• dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。

構成	説明
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Microsoft SQL サーバー JSON スキーマ

```
{
```

```
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
  "connectionConfiguration": {
    "type": "object",
    "properties": {
      "repositoryEndpointMetadata": {
        "type": "object",
        "properties": {
          "dbType": {
            "type": "string",
            "enum": [
              "mysql",
              "db2",
              "postgresql",
              "oracle",
              "sqlserver"
            ]
          },
          "dbHost": {
            "type": "string"
          },
          "dbPort": {
            "type": "string"
          },
          "dbInstance": {
            "type": "string"
          }
        },
        "required": [
          "dbType",
          "dbHost",
          "dbPort",
          "dbInstance"
        ]
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Microsoft Teams テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。テナント ID は、接続設定またはリポジトリエンドポイントの詳細の一部として指定します。また、データソースのタイプを MSTEAMS に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Microsoft Teams JSON スキーマ を参照してください。

次の表では、Microsoft Teams JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
tenantId	Microsoft 365 テナント ID。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• chatMessage • chatAttachment • channelPost • channelWiki • channelAttachment • meetingChat • meetingFile • meetingNote • calendarMeeting	Microsoft Teams Amazon Kendra コンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
paymentModel	Microsoft Teams データソースで使用する支払いモデルのタイプを指定します。モデル A の支払いモデルは、セキュリティコンプライアンスを必要とするライセンスモデルと支払いモデルに限定されます。モデル B の支払いモデルは、セキュリティコンプライアンスを必要としないライセンスモデルや支払いモデルに適しています。

構成	説明
• inclusionTeamNameフィルター • inclusionChannelName[フィルター] • inclusionFileNameパターン • inclusionFileTypeパターン • inclusionUserEmailフィルター • inclusionOneNoteSectionNamePatterns • inclusionOneNotePageNamePatterns	Microsoft Teams データソースにある特定のコンテンツを含めるための正規表現のパターンのリスト。パターンに一致するコンテンツは、インデックスに含まれます。パターンに一致しないコンテンツは、インデックスから除外されます。コンテンツが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのコンテンツはインデックスに含まれません。
• exclusionTeamName[フィルター] • exclusionChannelName[フィルター] • exclusionFileNameパターン • exclusionFileTypeパターン • exclusionUserEmailフィルター • exclusionOneNoteSectionNamePatterns • exclusionOneNotePageNamePatterns	Microsoft Teams のデータソースにある特定のコンテンツを除外するための正規表現のパターンのリスト。パターンに一致するコンテンツは、インデックスから除外されます。パターンに一致しないコンテンツは、インデックスに含まれます。コンテンツが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのコンテンツはインデックスに含まれません。
• isCrawlChatメッセージ • isCrawlChat添付ファイル • isCrawlChannel投稿 • isCrawlChannel添付ファイル • isCrawlChannelウィキ • isCrawlCalendarミーティング • isCrawlMeetingチャット • isCrawlMeeting[ファイル] • isCrawlMeeting注記	`true`Microsoft Teams データソース内のこれらの種類のコンテンツをクロールできます。
startCalendarDate時間	カレンダーのコンテンツには特定の開始日時を設定できます。

構成	説明
endCalendarDate時間	カレンダーのコンテンツには特定の終了日時を設定できます。
type	データソースのタイプ。データソースタイプとして MSTEAMS を指定します。
enableIdentityCrawler	true Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID 情報/プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	Microsoft Teams AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットのAmazon リソースネーム (ARN)。これには、Azure Portal で OAuth アプリケーションを作成したときに生成されるクライアント ID とクライアントシークレットが含まれます。
version	現在サポートされているこのテンプレートのバージョン。

Microsoft Teams JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "tenantId": {
              "type": "string",
              "pattern": "^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]
{12}$",
              "minLength": 36,
              "maxLength": 36
            }
          },
          "required": [
            "tenantId"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
```

```
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "chatMessage": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
```

```
        "chatAttachment": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "channelPost": {
          "type": "object",
          "properties": {
```

```
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "channelWiki": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
```

```
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "channelAttachment": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
```

```
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "meetingChat": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
```

```
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "STRING_LIST",
                    "DATE"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "meetingFile": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
```

```
                    "STRING",
                    "DATE",
                    "LONG"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "meetingNote": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE"
                  ]
```

```
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
                "dateFieldFormat": {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required": [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "calendarMeeting": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": [
            {
              "type": "object",
              "properties": {
                "indexFieldName": {
                  "type": "string"
                },
                "indexFieldType": {
                  "type": "string",
                  "enum": [
                    "STRING",
                    "DATE"
                  ]
                },
                "dataSourceFieldName": {
                  "type": "string"
                },
```

```
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
   "additionalProperties": {
    "type": "object",
    "properties": {
      "paymentModel": {
        "type": "string",
        "enum": [
          "A",
          "B",
          "Evaluation Mode"
        ]
      },
      "inclusionTeamNameFilter": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "exclusionTeamNameFilter": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "inclusionChannelNameFilter": {
```

```
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionChannelNameFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFileNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionUserEmailFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionOneNoteSectionNamePatterns": {
          "type": "array",
          "items": {
```

```
            "type": "string"
          }
        },
        "exclusionOneNoteSectionNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionOneNotePageNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionOneNotePageNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "isCrawlChatMessage": {
          "type": "boolean"
        },
        "isCrawlChatAttachment": {
          "type": "boolean"
        },
        "isCrawlChannelPost": {
          "type": "boolean"
        },
        "isCrawlChannelAttachment": {
          "type": "boolean"
        },
        "isCrawlChannelWiki": {
          "type": "boolean"
        },
        "isCrawlCalendarMeeting": {
          "type": "boolean"
        },
        "isCrawlMeetingChat": {
          "type": "boolean"
        },
        "isCrawlMeetingFile": {
          "type": "boolean"
```

```
        },
        "isCrawlMeetingNote": {
          "type": "boolean"
        },
        "startCalendarDateTime": {
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "endCalendarDateTime": {
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        }
      },
      "required": []
    },
    "type": {
      "type": "string",
      "pattern": "MSTEAMS"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
```

```
      ]
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

Microsoft Yammer テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを YAMMER に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、呼び出し時に Type **TEMPLATE** として指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。

次の表では、Microsoft Yammer JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースに関する設定情報。

構成	説明
repositoryEndpointMetadata	データソースのエンドポイント情報。このデータソースは `repositoryEndpointMetadata` のエンドポイントを指定していません。むしろ、 AWS Secrets Manager 接続情報はユーザーが提供するシークレットに含まれています。`secretArn`
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• community • ユーザー • message • 添付	Microsoft Yammer の属性またはフィールド名を Amazon Kendra インデックスフィールド名にマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
inclusionPatterns	Microsoft Yammer データソースにある特定のファイルを含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
exclusionPatterns	Microsoft Yammer データソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
sinceDate	Microsoft Yammer コネクタが特定の `sinceDate` に基づいてコンテンツをクロールするように `sinceDate` パラメータを設定できます。
communityNameFilter	特定のコミュニティコンテンツのインデックスを作成できます。 | • isCrawlMessage • isCrawlAttachment • isCrawlPrivateメッセージ	`true`メッセージ、メッセージ添付ファイル、プライベートメッセージをクロールします。
type	データソースタイプとして `YAMMER` を指定します。
secretARN	Microsoft Yammer AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットのAmazon リソースネーム (ARN)。これには、Microsoft Yammer のユーザー名とパスワード、Azure ポータルで OAuth アプリケーションを作成したときに生成されるクライアント ID とクライアントシークレットが含まれます。

構成	説明
useChangeLog	`true`Microsoft Yammer の変更ログを使用して、インデックス内の更新が必要なドキュメントを特定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • `FORCED_FULL_CRAWL` は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • `FULL_CRAWL` は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • `CHANGE_LOG` は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
enableIdentityCrawler	`true` Amazon Kendraの ID クローラを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID 情報やプリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

Microsoft Yammer JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
          }
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "community": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": [
                          "STRING",
                          "DATE"
                        ]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
```

```
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "user": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
```

```
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "message": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": {
            "anyOf": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": [
                      "STRING",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
```

```
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "attachment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
```

```
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
                    ]
                  }
                ]
              }
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      }
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "inclusionPatterns": {
          "type": "array"
        },
        "exclusionPatterns": {
          "type": "array"
        },
        "sinceDate": {
          "type": "string",
          "pattern": "^(19|2[0-9])[0-9]{2}-(0[1-9]|1[012])-(0[1-9]|[12][0-9]|
3[01])T(0[0-9]|1[0-9]|2[0-3]):([0-5][0-9]):([0-5][0-9])((\\+|-)(0[0-9]|1[0-9]|2[0-3]):
([0-5][0-9]))?$"
        },
        "communityNameFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "isCrawlMessage": {
          "type": "boolean"
        },
        "isCrawlAttachment": {
          "type": "boolean"
        },
        "isCrawlPrivateMessage": {
```

```
          "type": "boolean"
        }
      },
      "required": [
        "sinceDate"
      ]
    },
    "type": {
      "type": "string",
      "pattern": "YAMMER"
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    },
    "useChangeLog": {
      "type": "string",
      "enum": [
        "true",
        "false"
      ]
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "version": {
      "type": "string",
      "anyOf": [
        {
          "pattern": "1.0.0"
        }
      ]
    }
  },
  "required": [
```

```
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "type",
    "secretArn",
    "syncMode"
  ]
}
```

MySQL テンプレートスキーマ

データソーススキーマを含む JSON をオブジェクトの一部として含めます。TemplateConfigurationデータソースのタイプを JDBC に指定し、データベースタイプを mysql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。MySQL JSON スキーマ を参照してください。

次の表では、MySQL JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqlまたはのいずれでもかまいませんdb2。postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。

構成	説明
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。

構成	説明
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

MySQL JSON スキーマ

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Oracle Database テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを oracle に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Oracle Database JSON スキーマ を参照してください。

次の表では、Oracle データベース JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、mysqldb2またはのいずれでもかまいませんpostgresql 。oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。

構成	説明
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。

構成	説明
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして JDBC を指定します。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

構成	説明
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

Oracle Database JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "dbType": {
              "type": "string",
              "enum": [
                "mysql",
                "db2",
                "postgresql",
                "oracle",
                "sqlserver"
              ]
            },
            "dbHost": {
              "type": "string"
```

```
            },
            "dbPort": {
              "type": "string"
            },
            "dbInstance": {
              "type": "string"
            }
          },
          "required": [
            "dbType",
            "dbHost",
            "dbPort",
            "dbInstance"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
```

```
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required": [
          "fieldMappings"
        ]
      }
    },
    "required": [
    ]
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "primaryKey": {
        "type": "string"
      },
      "titleColumn": {
        "type": "string"
      },
      "bodyColumn": {
        "type": "string"
      },
      "sqlQuery": {
        "type": "string",
        "not": {
          "pattern": ";+"
        }
      },
      "timestampColumn": {
        "type": "string"
      },
      "timestampFormat": {
        "type": "string"
      },
      "timezone": {
        "type": "string"
      },
      "changeDetectingColumns": {
        "type": "array",
```

```
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
```

```
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

PostgreSQL テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。データソースのタイプを JDBC に指定し、データベースタイプを postgresql に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。PostgreSQL JSON スキーマ を参照してください。

次の表では、PostgreSQL JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースの接続に必要な設定情報。 • DBType — 使用する Java データベースのタイプ。、、、またはのいずれでもかまいませんmysql。db2 postgresql oracle sqlserver • dbHost - データベースのホスト名。 • dbPort - データベースポート。 • dbInstance - データベースインスタンス。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。データソースのタイプとシークレット ARN を指定します。

構成	説明
ドキュメント	Amazon Kendra データベースコンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。データベースデータソースに特定のコンテンツを含めたり除外したりするのに使用します。
primaryKey	データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
titleColumn	データベーステーブル内の文書タイトル列の名前を指定します。
bodyColumn	データベーステーブル内の文書タイトル列の名前を指定します。
sqlQuery	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
timestampColumn	タイムスタンプを含む列の名前を入力します。Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
timestampFormat	コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプ形式を含む列の名前を入力します。

構成	説明
timezone	クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
changeDetectingColumns	Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
allowedUsersColumns	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
allowedGroupsColumn	コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
sourceURIColumn	インデックスを作成するソース URL を含む列の名前を入力します。
isSslEnabled	SELECT や JOIN 操作などの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
type	データソースのタイプ。データソースタイプとして `JDBC` を指定します。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下を選択することができます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
secretArn	データベースに接続するためのユーザー名とパスワードが含まれている Secrets Manager シークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "user name": "database user name", "password": " password" }`
version	現在サポートされているテンプレートのバージョン。

PostgreSQL JSON スキーマ

```
{
```

```
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"properties": {
  "connectionConfiguration": {
    "type": "object",
    "properties": {
      "repositoryEndpointMetadata": {
        "type": "object",
        "properties": {
          "dbType": {
            "type": "string",
            "enum": [
              "mysql",
              "db2",
              "postgresql",
              "oracle",
              "sqlserver"
            ]
          },
          "dbHost": {
            "type": "string"
          },
          "dbPort": {
            "type": "string"
          },
          "dbInstance": {
            "type": "string"
          }
        },
        "required": [
          "dbType",
          "dbHost",
          "dbPort",
          "dbInstance"
        ]
      }
    },
    "required": [
      "repositoryEndpointMetadata"
    ]
  },
  "repositoryConfigurations": {
    "type": "object",
    "properties": {
```

```
        "document": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string"
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "primaryKey": {
          "type": "string"
        },
        "titleColumn": {
          "type": "string"
```

```
        },
        "bodyColumn": {
          "type": "string"
        },
        "sqlQuery": {
          "type": "string",
          "not": {
            "pattern": ";+"
          }
        },
        "timestampColumn": {
          "type": "string"
        },
        "timestampFormat": {
          "type": "string"
        },
        "timezone": {
          "type": "string"
        },
        "changeDetectingColumns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "allowedUsersColumn": {
          "type": "string"
        },
        "allowedGroupsColumn": {
          "type": "string"
        },
        "sourceURIColumn": {
          "type": "string"
        },
        "isSslEnabled": {
          "type": "boolean"
        }
      },
      "required": ["primaryKey", "titleColumn", "bodyColumn", "sqlQuery"]
    },
    "type" : {
      "type" : "string",
      "pattern": "JDBC"
    },
```

```
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
      "connectionConfiguration",
      "repositoryConfigurations",
      "syncMode",
      "additionalProperties",
      "secretArn",
      "type"
  ]
}
```

Salesforce テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として Salesforce ホスト URL を指定します。また、データソースのタイプを SALESFORCEV2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Salesforce JSON スキーマを参照してください。

次の表では、Salesforce JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
hostUrl	インデックスを作成する Salesforce インスタンスの URL。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• アカウント • contact • キャンペーン • ケース • product • lead • contract • partner • profile • idea • pricebook • タスク • solution • 添付 • ユーザー • ドキュメント • knowledgeArticles • グループ • opportunity • chatter	Salesforce Amazon Kendra エンティティの属性またはフィールド名をインデックスフィールド名にマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。

構成	説明
• customEntity	

secretARN	Salesforce AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。シークレットには、次のキーを持つ JSON 構造を含める必要があります。
	
	{
	"authenticationUrl": " *OAUTH endpoint that Amazon Kendra connects to get an OAUTH token*",
	"consumerKey": " *Application public key generated when you created your Salesforce application* ",
	"consumerSecret": " *Application private key generated when you created your Salesforce application* ",
	"password": " *Password associated with the user logging in to the Salesforce instance* ",
	"securityToken": " *Token associated with the user account logging in to the Salesforce instance* ",
	"username": " *User name of the user logging in to the Salesforce instance*"
	}

additionalProperties	データソース内のコンテンツ用の追加設定オプション。

構成	説明
• accountFilter • contactFilter • caseFilter • campaignFilter • contractFilter • groupFilter • leadFilter • productFilter • opportunityFilter • partnerFilter • pricebookFilter • ideaFilter • profileFilter • taskFilter • solutionFilter • userFilter • chatterFilter • documentFilter • knowledgeArticleFilter • customEntities	フィルタリングするエンティティを指定する文字列のコレクション。

構成	説明
inclusionPatterns • inclusionDocumentFileTypePatterns • inclusionDocumentFileNamePatterns • inclusionAccountFileTypePatterns • inclusionCampaignFileTypePatterns • inclusionDocumentFileNamePatterns • inclusionCampaignFileNamePatterns • inclusionCaseFileTypePatterns • inclusionCaseFileNamePatterns • inclusionContactFileTypePatterns • inclusionContractFileNamePatterns • inclusionLeadFileTypePatterns • inclusionLeadFileNamePatterns • inclusionOpportunityFileTypePatterns • inclusionOpportunityFileNamePatterns • inclusionSolutionFileTypePatterns • inclusionSolutionFileNamePatterns • inclusionTaskFileTypePatterns • inclusionTaskFileNamePatterns • inclusionGroupFileTypePatterns • inclusionGroupFileNamePatterns • inclusionChatterFileTypePatterns • inclusionChatterFileNamePatterns • inclusionCustomEntityFileTypePatterns • inclusionCustomEntityFileNamePatterns	特定のファイルを Salesforce データソースに含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
exclusionPatterns • exclusionDocumentFileTypePatterns • exclusionDocumentFileNamePatterns • exclusionAccountFileTypePatterns • exclusionCampaignFileTypePatterns • exclusionCampaignFileNamePatterns • exclusionCaseFileTypePatterns • exclusionCaseFileNamePatterns • exclusionContactFileTypePatterns • exclusionContractFileNamePatterns • exclusionLeadFileTypePatterns • exclusionLeadFileNamePatterns • exclusionOpportunityFileTypePatterns • exclusionOpportunityFileNamePatterns • exclusionSolutionFileTypePatterns • exclusionSolutionFileNamePatterns • exclusionTaskFileTypePatterns • exclusionTaskFileNamePatterns • exclusionGroupFileTypePatterns • exclusionGroupFileNamePatterns • exclusionChatterFileTypePatterns • exclusionChatterFileNamePatterns • exclusionCustomEntityFileTypePatterns • exclusionCustomEntityFileNamePatterns	特定のファイルを Salesforce データソースから除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
• isCrawlAccount • isCrawlContact • isCrawlCase • isCrawlCampaign • isCrawlProduct • isCrawlLead • isCrawlContract • isCrawlPartner • isCrawlProfile • isCrawlIdea • isCrawlPricebook • isCrawlDocument • crawlSharedDocument • isCrawlGroup • isCrawlOpportunity • isCrawlChatter • isCrawlUser • isCrawlSolution • isCrawlTask • isCrawlAccount添付ファイル • isCrawlContact添付ファイル • isCrawlCase添付ファイル • isCrawlCampaign添付ファイル • isCrawlLead添付ファイル • isCrawlContract添付ファイル • isCrawlGroup添付ファイル • isCrawlOpportunity添付ファイル • isCrawlChatter添付ファイル • isCrawlSolution添付ファイル	`true`Salesforce アカウント内のこれらの種類のファイルをクロールします。

構成	説明

• isCrawlTask添付ファイル
• isCrawlCustomEntityAttachments
• isCrawlKnowledge記事
  • isCrawlDraft
  • isCrawlPublish
  • isCrawlArchived	

type	データソースのタイプ。データソースタイプとして SALESFORCEV2 を指定します。
enableIdentityCrawler	true Amazon Kendraの ID クローラを使用して、特定のドキュメントへのアクセス権を持つユーザやグループの ID /プリンシパル情報を同期すること。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。 • CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
version	現在サポートされているこのテンプレートのバージョン。

Salesforce JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties":
  {
    "connectionConfiguration": {
      "type": "object",
      "properties":
      {
        "repositoryEndpointMetadata":
        {
          "type": "object",
          "properties":
```

```
          {
            "hostUrl":
            {
              "type": "string",
              "pattern": "https:.*"
            }
          },
          "required":
          [
            "hostUrl"
          ]
        }
      },
      "required":
      [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties":
      {
        "account":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
```

```
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE",
                      "LONG"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "contact":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
```

```
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "campaign":
      {
        "type": "object",
```

```
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
```

```
            }
          },
          "required":
          [
            "fieldMappings"
          ]
        },
        "case":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
```

```
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "product":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
```

```
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "lead":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
```

```
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required":
          [
            "fieldMappings"
          ]
        },
        "contract":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
```

```
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
```

```
        "partner":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
```

```
            }
          ]
        }
      },
      "required":
      [
        "fieldMappings"
      ]
    },
    "profile":
    {
      "type": "object",
      "properties":
      {
        "fieldMappings":
        {
          "type": "array",
          "items":
          [
            {
              "type": "object",
              "properties":
              {
                "indexFieldName":
                {
                  "type": "string"
                },
                "indexFieldType":
                {
                  "type": "string",
                  "enum":
                  [
                    "STRING",
                    "STRING_LIST",
                    "DATE"
                  ]
                },
                "dataSourceFieldName":
                {
                  "type": "string"
                },
                "dateFieldFormat":
                {
                  "type": "string",
```

```
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "idea":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE",
```

```
                    "LONG"
                  ]
                },
                "dataSourceFieldName":
                {
                  "type": "string"
                },
                "dateFieldFormat":
                {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required":
              [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required":
      [
        "fieldMappings"
      ]
    },
    "pricebook":
    {
      "type": "object",
      "properties":
      {
        "fieldMappings":
        {
          "type": "array",
          "items":
          [
            {
              "type": "object",
              "properties":
              {
                "indexFieldName":
                {
```

```
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "task":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
```

```
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
```

```
          ]
        },
        "solution":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
```

```
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required":
          [
            "fieldMappings"
          ]
        },
        "attachment":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
```

```
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required":
          [
            "fieldMappings"
          ]
        },
        "user":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
```

```
                    "STRING",
                    "STRING_LIST",
                    "DATE"
                  ]
                },
                "dataSourceFieldName":
                {
                  "type": "string"
                },
                "dateFieldFormat":
                {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required":
              [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required":
      [
        "fieldMappings"
      ]
    },
    "document":
    {
      "type": "object",
      "properties":
      {
        "fieldMappings":
        {
          "type": "array",
          "items":
          [
            {
              "type": "object",
              "properties":
              {
```

```
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE",
                        "LONG"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
                  [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required":
          [
            "fieldMappings"
          ]
        },
        "knowledgeArticles":
        {
          "type": "object",
          "properties":
```

```
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
```

```
          "required":
          [
            "fieldMappings"
          ]
        },
        "group":
        {
          "type": "object",
          "properties":
          {
            "fieldMappings":
            {
              "type": "array",
              "items":
              [
                {
                  "type": "object",
                  "properties":
                  {
                    "indexFieldName":
                    {
                      "type": "string"
                    },
                    "indexFieldType":
                    {
                      "type": "string",
                      "enum":
                      [
                        "STRING",
                        "STRING_LIST",
                        "DATE"
                      ]
                    },
                    "dataSourceFieldName":
                    {
                      "type": "string"
                    },
                    "dateFieldFormat":
                    {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required":
```

```
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "opportunity":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE",
                      "LONG"
                    ]
                  },
                  "dataSourceFieldName":
```

```
                {
                  "type": "string"
                },
                "dateFieldFormat":
                {
                  "type": "string",
                  "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                }
              },
              "required":
              [
                "indexFieldName",
                "indexFieldType",
                "dataSourceFieldName"
              ]
            }
          ]
        }
      },
      "required":
      [
        "fieldMappings"
      ]
    },
    "chatter":
    {
      "type": "object",
      "properties":
      {
        "fieldMappings":
        {
          "type": "array",
          "items":
          [
            {
              "type": "object",
              "properties":
              {
                "indexFieldName":
                {
                  "type": "string"
                },
                "indexFieldType":
                {
```

```
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      },
      "customEntity":
      {
        "type": "object",
        "properties":
        {
          "fieldMappings":
          {
            "type": "array",
            "items":
            [
              {
```

```
                "type": "object",
                "properties":
                {
                  "indexFieldName":
                  {
                    "type": "string"
                  },
                  "indexFieldType":
                  {
                    "type": "string",
                    "enum":
                    [
                      "STRING",
                      "STRING_LIST",
                      "DATE"
                    ]
                  },
                  "dataSourceFieldName":
                  {
                    "type": "string"
                  },
                  "dateFieldFormat":
                  {
                    "type": "string",
                    "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                  }
                },
                "required":
                [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        },
        "required":
        [
          "fieldMappings"
        ]
      }
    }
  },
```

```
    "additionalProperties": {
      "type": "object",
      "properties":
      {
        "accountFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "contactFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "caseFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "campaignFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "contractFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "groupFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
```

```
        }
      },
      "leadFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "productFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "opportunityFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "partnerFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "pricebookFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "ideaFilter":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
```

```
        "profileFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "taskFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "solutionFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "userFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "chatterFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "documentFilter":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "knowledgeArticleFilter":{
          "type": "array",
```

```
          "items":
          {
            "type": "string"
          }
        },
        "customEntities":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "isCrawlAccount": {
          "type": "boolean"
        },
        "isCrawlContact": {
          "type": "boolean"
        },
        "isCrawlCase": {
          "type": "boolean"
        },
        "isCrawlCampaign": {
          "type": "boolean"
        },
        "isCrawlProduct": {
          "type": "boolean"
        },
        "isCrawlLead": {
          "type": "boolean"
        },
        "isCrawlContract": {
          "type": "boolean"
        },
        "isCrawlPartner": {
          "type": "boolean"
        },
        "isCrawlProfile": {
          "type": "boolean"
        },
        "isCrawlIdea": {
          "type": "boolean"
        },
        "isCrawlPricebook": {
          "type": "boolean"
```

```
        },
        "isCrawlDocument": {
          "type": "boolean"
        },
        "crawlSharedDocument": {
          "type": "boolean"
        },
        "isCrawlGroup": {
          "type": "boolean"
        },
        "isCrawlOpportunity": {
          "type": "boolean"
        },
        "isCrawlChatter": {
          "type": "boolean"
        },
        "isCrawlUser": {
          "type": "boolean"
        },
        "isCrawlSolution":{
          "type": "boolean"
        },
        "isCrawlTask":{
          "type": "boolean"
        },

        "isCrawlAccountAttachments": {
          "type": "boolean"
        },
        "isCrawlContactAttachments": {
          "type": "boolean"
        },
        "isCrawlCaseAttachments": {
          "type": "boolean"
        },
        "isCrawlCampaignAttachments": {
          "type": "boolean"
        },
        "isCrawlLeadAttachments": {
          "type": "boolean"
        },
        "isCrawlContractAttachments": {
          "type": "boolean"
        },
```

```
        "isCrawlGroupAttachments": {
          "type": "boolean"
        },
        "isCrawlOpportunityAttachments": {
          "type": "boolean"
        },
        "isCrawlChatterAttachments": {
          "type": "boolean"
        },
        "isCrawlSolutionAttachments":{
          "type": "boolean"
        },
        "isCrawlTaskAttachments":{
          "type": "boolean"
        },
        "isCrawlCustomEntityAttachments":{
          "type": "boolean"
        },
        "isCrawlKnowledgeArticles": {
          "type": "object",
          "properties":
          {
            "isCrawlDraft": {
              "type": "boolean"
            },
            "isCrawlPublish": {
              "type": "boolean"
            },
            "isCrawlArchived": {
              "type": "boolean"
            }
          }
        },
        "inclusionDocumentFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionDocumentFileTypePatterns": {
          "type": "array",
          "items":
          {
```

```
            "type": "string"
          }
        },
        "inclusionDocumentFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionDocumentFileNamePatterns": {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionAccountFileTypePatterns": {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionAccountFileTypePatterns": {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionAccountFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionAccountFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
```

```
        },
        "inclusionCampaignFileTypePatterns": {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCampaignFileTypePatterns": {
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionCampaignFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCampaignFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionCaseFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCaseFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionCaseFileNamePatterns":{
```

```
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCaseFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionContactFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionContactFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionContactFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionContactFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionContractFileTypePatterns":{
          "type": "array",
          "items":
```

```
          {
            "type": "string"
          }
        },
        "exclusionContractFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionContractFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionContractFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionLeadFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionLeadFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionLeadFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
```

```
        }
      },
      "exclusionLeadFileNamePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "inclusionOpportunityFileTypePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "exclusionOpportunityFileTypePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "inclusionOpportunityFileNamePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "exclusionOpportunityFileNamePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
      "inclusionSolutionFileTypePatterns":{
        "type": "array",
        "items":
        {
          "type": "string"
        }
      },
```

```
        "exclusionSolutionFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionSolutionFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionSolutionFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionTaskFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionTaskFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionTaskFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionTaskFileNamePatterns":{
          "type": "array",
```

```
          "items":
          {
            "type": "string"
          }
        },
        "inclusionGroupFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionGroupFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionGroupFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionGroupFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionChatterFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionChatterFileTypePatterns":{
          "type": "array",
          "items":
          {
```

```
            "type": "string"
          }
        },
        "inclusionChatterFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionChatterFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionCustomEntityFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCustomEntityFileTypePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "inclusionCustomEntityFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
        },
        "exclusionCustomEntityFileNamePatterns":{
          "type": "array",
          "items":
          {
            "type": "string"
          }
```

```
        }
      },
      "required":
      []
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "type": {
      "type": "string",
      "pattern": "SALESFORCEV2"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FULL_CRAWL",
        "FORCED_FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

ServiceNow テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部として、 ServiceNow ホスト URL、認証タイプ、インスタンスバージョンを指定します。また、データソースのタイプを SERVICENOWV2 に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出し時にとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。ServiceNow JSON スキーマ を参照してください。

次の表では、 ServiceNow JSON スキーマのパラメータについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
hostUrl	ServiceNow ホスト URL。例えば、*your-domain.service-now.com* です。
authType	使用する認証のタイプ (basicAuth または OAuth2)。
servicenowInstanceVersion	ServiceNow 使用しているバージョン。Tokyo、、SandiegoRome、のいずれかを選択できますOthers。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• knowledgeArticle • 添付 • serviceCatalog • インシデント	ServiceNowナレッジ記事、添付ファイル、サービスカタログ、 Amazon Kendra インシデントの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。ServiceNow ServiceNow データソースのフィールド名はカスタムメタデータに存在している必要があります。

構成	説明
その他のプロパティ	データソース内のコンテンツ用の追加設定オプション。
maxFileSizeInMegaBytes	Amazon Kendra がクロールするファイルサイズの制限を MB 単位で指定します。Amazon Kendra は、定義したサイズ制限内のファイルのみをクロールします。デフォルトのファイルサイズは 50 MB です。最大ファイルサイズは 0 MB 以上 50 MB 以下でなければなりません。
• knowledgeArticleFilter • incidentQueryFilter • serviceCatalogQueryフィルター • knowledgeArticleTitleRegExp • serviceCatalogTitleRegExp • incidentTitleRegExp • inclusionFileTypeパターン • exclusionFileTypeパターン • inclusionFileNameパターン • exclusionFileNameパターン • incidentStateType	ServiceNow データソース内の特定のファイルを含めたり除外したりする正規表現パターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。

構成	説明
• isCrawlKnowledge記事 • isCrawlKnowledgeArticleAttachment • includePublicArticlesのみ • isCrawlService[カタログ] • isCrawlServiceCatalogAttachment • isCrawlActiveServiceCatalog • isCrawlInactiveServiceCatalog • isCrawlIncident • isCrawlIncident添付ファイル • isCrawlActive事件 • isCrawlInactive事件 • ACL を適用 ForKnowledgeArticle • ACL を適用 ForServiceCatalog • ACL を適用 ForIncident	`true` ServiceNow ナレッジ記事、サービスカタログ、インシデント、添付ファイルをクロールします。
type	データソースのタイプ。データソースタイプとして `SERVICENOWV2` を指定します。
enableIdentityCrawler	`true` Amazon Kendraの ID クローラーを使用して、特定のドキュメントへのアクセス権を持つユーザーおよびグループの ID /プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。

構成	説明
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。 • FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。 • FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。

secretARN	AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。 ServiceNowシークレットには、次のキーを持つ JSON 構造を含める必要があります。
	`{ "username": " user name", "password": " password" }`
	OAuth2 認証を使用する場合、シークレットには、次のキーを含む JSON 構造を含める必要があります。
	`{ "username": " user name", "password": " password", "clientId": " client id", "clientSecret": " client secret" }`

version	現在サポートされているテンプレートのバージョン。

ServiceNow JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "hostUrl": {
              "type": "string",
              "pattern": "^(?!(^(https?|ftp|file):\/\/))[a-z0-9-]+(.service-
now.com|.servicenowservices.com)$",
              "minLength": 1,
              "maxLength": 2048
            },
            "authType": {
              "type": "string",
              "enum": [
                "basicAuth",
                "OAuth2"
              ]
            },
            "servicenowInstanceVersion": {
              "type": "string",
              "enum": [
                "Tokyo",
                "Sandiego",
                "Rome",
                "Others"
                ]
            }
          },
          "required": [
            "hostUrl",
            "authType",
            "servicenowInstanceVersion"
          ]
        }
      },
      "required": [
```

```
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "knowledgeArticle": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
```

```
            "fieldMappings"
          ]
        },
        "attachment": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "LONG",
                        "DATE",
                        "STRING_LIST"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
```

```
        },
        "serviceCatalog": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "incident": {
          "type": "object",
```

```
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": [
                        "STRING",
                        "DATE",
                        "STRING_LIST"
                      ]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
          "required": [
            "fieldMappings"
          ]
        }
      }
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
```

```
        "maxFileSizeInMegaBytes": {
          "type": "string"
        },
        "isCrawlKnowledgeArticle": {
          "type": "boolean"
        },
        "isCrawlKnowledgeArticleAttachment": {
          "type": "boolean"
        },
        "includePublicArticlesOnly": {
          "type": "boolean"
        },
        "knowledgeArticleFilter": {
          "type": "string"
        },
        "incidentQueryFilter": {
          "type": "string"
        },
        "serviceCatalogQueryFilter": {
          "type": "string"
        },
        "isCrawlServiceCatalog": {
          "type": "boolean"
        },
        "isCrawlServiceCatalogAttachment": {
          "type": "boolean"
        },
        "isCrawlActiveServiceCatalog": {
          "type": "boolean"
        },
        "isCrawlInactiveServiceCatalog": {
          "type": "boolean"
        },
        "isCrawlIncident": {
          "type": "boolean"
        },
        "isCrawlIncidentAttachment": {
          "type": "boolean"
        },
        "isCrawlActiveIncident": {
          "type": "boolean"
        },
        "isCrawlInactiveIncident": {
          "type": "boolean"
```

```
        },
        "applyACLForKnowledgeArticle": {
          "type": "boolean"
        },
        "applyACLForServiceCatalog": {
          "type": "boolean"
        },
        "applyACLForIncident": {
          "type": "boolean"
        },
        "incidentStateType": {
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Open",
              "Open - Unassigned",
              "Resolved",
              "All"
            ]
          }
        },
        "knowledgeArticleTitleRegExp": {
          "type": "string"
        },
        "serviceCatalogTitleRegExp": {
          "type": "string"
        },
        "incidentTitleRegExp": {
          "type": "string"
        },
        "inclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileTypePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionFileNamePatterns": {
```

```
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "exclusionFileNamePatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      },
      "required": []
    },
    "type": {
      "type": "string",
      "pattern": "SERVICENOWV2"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
        "FULL_CRAWL"
      ]
    },
    "secretArn": {
      "type": "string",
      "minLength": 20,
      "maxLength": 2048
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
```

```
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type"
  ]
}
```

Slack テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部としてホスト URL を指定します。また、データソースのタイプを SLACK に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。スラック JSON スキーマを参照してください。

以下の表では、Slack JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
チーム ID	Slack のメインページ URL からコピーした Slack チーム ID。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
すべて	Slack Amazon Kendra コンテンツの属性またはフィールド名をインデックスフィールド名にマップするオブジェクトのリスト。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。

構成	説明
inclusionPatterns	Slackデータソースに特定のコンテンツを含めるための正規表現パターンのリスト。パターンに一致するコンテンツは、インデックスに含まれます。パターンに一致しないコンテンツは、インデックスから除外されます。包含パターンと除外パターンの両方に一致するコンテンツがある場合、除外パターンが優先され、コンテンツはインデックスに含まれません。
exclusionPatterns	データソース内の特定のコンテンツを除外するための正規表現パターンのリスト。Slackパターンに一致するコンテンツは、インデックスから除外されます。パターンに一致しないコンテンツはインデックスに含まれます。包含パターンと除外パターンの両方に一致するコンテンツがある場合、除外パターンが優先され、コンテンツはインデックスに含まれません。
crawlBotMessages	`true`ボットメッセージをクロールします。
除外/アーカイブ済み	`true`アーカイブされたメッセージのクロールを除外します。
会話タイプ	、PUBLIC_CHANNEL 、PRIVATE_CHANNEL のいずれかをインデックスに登録したい会話のタイプ。GROUP_MESSAGE DIRECT_MESSAGE
チャンネルフィルター	またはにインデックスを付けるチャンネルのタイプ。private_channel public_channel
sinceDate	sinceDate Slackコネクタが特定の内容に基づいてコンテンツをクロールするようにパラメーターを構成できます。sinceDate

構成	説明
ルックバック	lookBackSlack前回のコネクタ同期の指定時間前までにコネクタが更新または削除されたコンテンツをクロールするようにパラメータを設定できます。
syncMode	すべてのドキュメントを同期してインデックスを更新するか、新規、変更、 Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。 • データソースがインデックスと同期されるたびに、すべてのコンテンツを再クロールしたりFORCED_FULL_CRAWL 、既存のコンテンツを置き換えたりできます。 • FULL_CRAWL データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールできます。 • CHANGE_LOG データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールする場合に使用します。
type	データソースのタイプ。データソースタイプとして SLACK を指定します。

構成	説明
enableIdentityCrawler	`true` Amazon Kendraの ID クローラーを使用して、特定のドキュメントにアクセスできるユーザーやグループの ID /プリンシパル情報を同期します。ID クローラーがオフになっている場合は、すべてのドキュメントを公開検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、代わりに PutPrincipalMappingAPI を使用してユーザーとグループのアクセス情報をアップロードできます。
secretArn	AWS Secrets Manager への接続に必要なキーと値のペアを含むシークレットの Amazon リソースネーム (ARN)。Slackシークレットには、次のキーを持つ JSON 構造を含める必要があります。 `{ "slackToken": " token" }`
version	このテンプレートの現在サポートされているバージョン。

スラック JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
```

```
            "teamId": {
              "type": "string"
            }
          },
          "required": ["teamId"]
        }
      }
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "All": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "DATE","LONG"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "yyyy-MM-dd'T'HH:mm:ss'Z'"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          },
```

```
          "required": [
            "fieldMappings"
          ]
        }
      },
      "required": [
      ]
    },
    "additionalProperties": {
      "type": "object",
      "properties": {
        "exclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "inclusionPatterns": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "crawlBotMessages": {
          "type": "boolean"
        },
        "excludeArchived": {
          "type": "boolean"
        },
        "conversationType": {
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "PUBLIC_CHANNEL",
              "PRIVATE_CHANNEL",
              "GROUP_MESSAGE",
              "DIRECT_MESSAGE"
            ]
          }
        },
        "channelFilter": {
            "type": "object",
            "properties": {
```

```
            "private_channel": {
              "type": "array",
              "items": {
                "type": "string"
              }
            },
            "public_channel": {
              "type": "array",
              "items": {
                "type": "string"
              }
            }
          }
        },
        "channelIdFilter": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "sinceDate": {
          "anyOf": [
            {
              "type": "string",
              "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z$"
            },
            {
              "type": "string",
              "pattern": ""
            }
          ]
        },
        "lookBack": {
          "type": "string",
          "pattern": "^[0-9]*$"
        }
      },
      "required": [
      ]
    },
    "syncMode": {
      "type": "string",
      "enum": [
        "FORCED_FULL_CRAWL",
```

```
        "FULL_CRAWL",
        "CHANGE_LOG"
      ]
    },
    "type" : {
      "type" : "string",
      "pattern": "SLACK"
    },
    "enableIdentityCrawler": {
      "type": "boolean"
    },
    "secretArn": {
      "type": "string"
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "syncMode",
    "additionalProperties",
    "secretArn",
    "type",
    "enableIdentityCrawler"
  ]
}
```

Zendesk テンプレートスキーマ

データソーススキーマを含む JSON TemplateConfigurationをオブジェクトの一部として含めます。接続設定またはリポジトリエンドポイントの詳細の一部としてホスト URL を指定します。また、データソースのタイプを ZENDESK に指定します。認証情報のシークレット、およびその他の必要な設定を指定します。次に、TEMPLATEType呼び出すときとしてを指定しますCreateDataSource。

このデベロッパーガイドで提供されているテンプレートを使用できます。Zendesk JSON スキーマを参照してください。

次の表では、Zendesk JSON スキーマのパラメーターについて説明しています。

構成	説明
connectionConfiguration	データソースのエンドポイントの設定情報。
repositoryEndpointMetadata	データソースのエンドポイント情報。
hostURL	Zendesk のホスト URL。例えば、https://yoursubdomain.zendesk.com。
repositoryConfigurations	データソースのコンテンツに関する設定情報。例えば、特定のタイプのコンテンツやフィールドマッピングの設定などです。
• ticket • ticketComment • ticketCommentAttachment • article • articleComment • articleAttachment • communityTopic • communityPostComment	Zendesk チケットの属性またはフィールド名を Amazon Kendra インデックスフィールド名にマッピングするオブジェクトのリスト。詳細については、データソースフィールドのマッピングを参照してください。
secretARN	AWS Secrets Manager Zendeskへの接続に必要なキーと値のペアを含むシークレットのAmazonリソースネーム（ARN）。シークレットには、ホスト URL、クライアント ID、クライアントシークレット、ユーザー名、パスワードのキーを含む JSON 構造が含まれている必要があります。
additionalProperties	データソース内のコンテンツ用の追加設定オプション。
organizationNameFilter	特定の [組織] 内に存在するチケットのインデックスを作成できます。

構成	説明
sinceDate	Zendesk コネクタが特定の `sinceDate` に基づいてコンテンツをクロールするように `sinceDate` パラメータを設定できます。
inclusionPatterns	Zendesk データソースにある特定のファイルを含めるための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスに含まれます。パターンに一致しないファイルは、インデックスから除外されます。ファイルが包含パターンと除外パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
exclusionPatterns	Zendesk データソースにある特定のファイルを除外するための正規表現のパターンのリスト。パターンに一致するファイルは、インデックスから除外されます。パターンに一致しないファイルは、インデックスに含まれます。ファイルが除外パターンと包含パターンの両方に一致する場合、除外パターンが優先され、そのファイルはインデックスに含まれません。
• isCrawlTicket • isCrawlTicketコメント • isCrawlTicketCommentAttachment • isCrawlArticle • isCrawlArticle[コメント] • isCrawlArticle添付ファイル • isCrawlCommunityトピック • isCrawlCommunity投稿 • isCrawlCommunityPostComment	「`true`」を入力すると、この種のコンテンツをクロールできます。
type	データソースタイプとして `ZENDESK` を指定します。

構成	説明
useChangeLog	「true」を入力すると、Zendeskの変更ログを使用して、インデックス内の更新が必要なドキュメントを特定できます。変更ログのサイズによっては、Zendesk でドキュメントをスキャンする方が速い場合があります。Zendesk データソースをインデックスに初めて同期する場合は、すべてのドキュメントがスキャンされます。

Zendesk JSON スキーマ

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "connectionConfiguration": {
      "type": "object",
      "properties": {
        "repositoryEndpointMetadata": {
          "type": "object",
          "properties": {
            "hostUrl": {
              "type": "string",
              "pattern": "https:.*"
            }
          },
          "required": [
            "hostUrl"
          ]
        }
      },
      "required": [
        "repositoryEndpointMetadata"
      ]
    },
    "repositoryConfigurations": {
      "type": "object",
      "properties": {
        "ticket": {
```

```
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
                      },
                      "dateFieldFormat": {
                        "type": "string",
                        "pattern": "dd-MM-yyyy HH:mm:ss"
                      }
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"

                    ]
                  }
                ]
              }
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "ticketComment": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
```

```
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"

                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "ticketCommentAttachment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
```

```
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
                      },
                      "dateFieldFormat": {
                        "type": "string",
                        "pattern": "dd-MM-yyyy HH:mm:ss"
                      }
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
                    ]
                  }
                ]
              }
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "article": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
```

```
                    "type": "string",
                    "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
                  },
                  "dateFieldFormat": {
                    "type": "string",
                    "pattern": "dd-MM-yyyy HH:mm:ss"
                  }
                },
                "required": [
                  "indexFieldName",
                  "indexFieldType",
                  "dataSourceFieldName"
                ]
              }
            ]
          }
        }
      },
      "required": [
        "fieldMappings"
      ]
    },
    "communityPostComment": {
      "type": "object",
      "properties": {
        "fieldMappings": {
          "type": "array",
          "items": {
            "anyOf": [
              {
                "type": "object",
                "properties": {
                  "indexFieldName": {
                    "type": "string"
                  },
                  "indexFieldType": {
                    "type": "string",
                    "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                  },
                  "dataSourceFieldName": {
                    "type": "string"
```

```
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
                  },
                  "required": [
                    "indexFieldName",
                    "indexFieldType",
                    "dataSourceFieldName"
                  ]
                }
              ]
            }
          }
        },
        "required": [
          "fieldMappings"
        ]
      },
      "articleComment": {
        "type": "object",
        "properties": {
          "fieldMappings": {
            "type": "array",
            "items": {
              "anyOf": [
                {
                  "type": "object",
                  "properties": {
                    "indexFieldName": {
                      "type": "string"
                    },
                    "indexFieldType": {
                      "type": "string",
                      "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                    },
                    "dataSourceFieldName": {
                      "type": "string"
                    },
                    "dateFieldFormat": {
                      "type": "string",
                      "pattern": "dd-MM-yyyy HH:mm:ss"
                    }
```

```
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
                    ]
                  }
                ]
              }
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "articleAttachment": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
                      },
                      "dateFieldFormat": {
                        "type": "string",
                        "pattern": "dd-MM-yyyy HH:mm:ss"
                      }
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
```

```
                    ]
                  }
                ]
              }
            }
          },
          "required": [
            "fieldMappings"
          ]
        },
        "communityTopic": {
          "type": "object",
          "properties": {
            "fieldMappings": {
              "type": "array",
              "items": {
                "anyOf": [
                  {
                    "type": "object",
                    "properties": {
                      "indexFieldName": {
                        "type": "string"
                      },
                      "indexFieldType": {
                        "type": "string",
                        "enum": ["STRING", "STRING_LIST", "LONG", "DATE"]
                      },
                      "dataSourceFieldName": {
                        "type": "string"
                      },
                      "dateFieldFormat": {
                        "type": "string",
                        "pattern": "dd-MM-yyyy HH:mm:ss"
                      }
                    },
                    "required": [
                      "indexFieldName",
                      "indexFieldType",
                      "dataSourceFieldName"
                    ]
                  }
                ]
              }
            }
```

```
        },
        "required": [
          "fieldMappings"
        ]
      }
    }
  },
  "secretArn": {
    "type": "string",
    "minLength": 20,
    "maxLength": 2048
  },
  "additionalProperties": {
    "type": "object",
    "properties": {
      "organizationNameFilter": {
        "type": "array"
      },
      "sinceDate": {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2} [0-9]{2}:[0-9]{2}:[0-9]{2}$"
      },
      "inclusionPatterns": {
        "type": "array"
      },
      "exclusionPatterns": {
        "type": "array"
      },
      "isCrawTicket": {
        "type": "string"
      },
      "isCrawTicketComment": {
        "type": "string"
      },
      "isCrawTicketCommentAttachment": {
        "type": "string"
      },
      "isCrawlArticle": {
        "type": "string"
      },
      "isCrawlArticleAttachment": {
        "type": "string"
      },
      "isCrawlArticleComment": {
```

```
          "type": "string"
        },
        "isCrawlCommunityTopic": {
          "type": "string"
        },
        "isCrawlCommunityPost": {
          "type": "string"
        },
        "isCrawlCommunityPostComment": {
          "type": "string"
        }
      }
    },
    "type": {
      "type": "string",
      "pattern": "ZENDESK"
    },
    "useChangeLog": {
      "type": "string",
      "enum": ["true", "false"]
    }
  },
  "version": {
    "type": "string",
    "anyOf": [
      {
        "pattern": "1.0.0"
      }
    ]
  },
  "additionalProperties": false,
  "required": [
    "connectionConfiguration",
    "repositoryConfigurations",
    "additionalProperties",
    "useChangeLog",
    "secretArn",
    "type"
  ]
}
```

Adobe Experience Manager

Adobe Experience Manager は、ウェブサイトまたはモバイルアプリケーションのコンテンツの作成に使用されるコンテンツ管理システムです。を使用して Amazon Kendra 、ページAdobe Experience Managerとコンテンツアセットに接続し、インデックスを作成できます。

Amazon Kendra はAdobe Experience Manager、 (AEM) を Cloud Service オーサリングインスタンスとして、Adobe Experience Managerオンプレミスのオーサリングおよびパブリッシュインスタンスとしてサポートしています。

Amazon Kendra コンソールまたは TemplateConfiguration API を使用して、Adobe Experience Managerデータソース Amazon Kendra に接続できます。

Amazon Kendra Adobe Experience Manager データソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

Adobe Experience Manager データソースコネクタは以下の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期
- OAuth 2.0 と基本的な認証
- 仮想プライベートクラウド (VPC)

前提条件

を使用してAdobe Experience Managerデータソース Amazon Kendra のインデックスを作成する前に、 Adobe Experience Managerおよび AWS アカウントでこれらの変更を行ってください。

Adobe Experience Manager で以下を確認してください。

- 管理者権限を持つアカウント、または管理者ユーザーへのアクセス。
- Adobe Experience Manager ホスト URL をコピー済み。

> **Note**
>
> (オンプレミス/サーバー) Amazon Kendra に含まれるエンドポイント情報が、データソース設定の詳細で指定されたエンドポイント情報 AWS Secrets Manager と同じかどうかを確認します。混乱する代理問題は、ユーザーがアクションを実行するアクセス許可がないにもかかわらず、 Amazon Kendra をプロキシとして使用して設定された秘密にアクセスし、アクションを実行するセキュリティの問題です。後でエンドポイント情報を変更する場合は、新しいシークレットを作成してこの情報を同期する必要があります。

- 管理者ユーザー名とパスワードの基本認証情報を記録済み。
- オプション: Adobe Experience Manager (AEM) でクラウドサービスまたは AEM オンプレミスとして OAuth 2.0 認証情報を生成しました。AEM オンプレミスを使用する場合、認証情報にはクライアント ID、クライアントシークレット、プライベートキーが含まれます。AEM をクラウドサービスとして使用する場合、認証情報にはクライアント ID、クライアントシークレット、プライベートキー、組織 ID、テクニカルアカウント ID、および Adobe Identity Management System (IMS) ホストが含まれます。AEM をクラウドサービスとして使用するための認証情報を生成する方法については、「Adobe Experience Manager ドキュメント」を参照してください。AEM オンプレミスでは、Adobe Granite OAuth 2.0 サーバー実装 (com.adobe.granite.oauth.server) が AEM の OAuth 2.0 サーバー機能をサポートしています。
- 各ドキュメントが Adobe Experience Manager および同じインデックスに使用する予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

で AWS アカウント、以下があることを確認します。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。
- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- Adobe Experience Manager の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合はシークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、Adobe Experience Manager データソースをに接続するときに、コンソールを使用して新しい IAM ロールと Secrets Manager シークレットを作成できます Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Adobe Experience Manager データソース Amazon Kendra に接続するには、 がAdobe Experience Managerデータ Amazon Kendra にアクセスできるように、データソースの必要な詳細を入力する必要があります。に をまだ設定していない場合は Amazon Kendra、Adobe Experience Manager「」を参照してください前提条件。

Console

Amazon Kendra に接続するには Adobe Experience Manager

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加ページで、Adobe Experience Manager コネクタ を選択し、コネクタ を追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 - インデックスのドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。

   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] - [AEM オンプレミス] または [クラウドサービスとしての AEM] を選択します。

   Adobe Experience Manager ホスト URL を入力します。例えば、AEM オンプレミスを使用する場合は、ホスト名とポートを含めます。https://hostname:port。または、AEM をクラウドサービスとして使用する場合は、作成者 URL を使用できます。https://author-xxxxxx-xxxxxxx.adobeaemcloud.com。

   b. [SSL 証明書の場所] - Amazon S3 バケットに保存されている SSL 証明書へのパスを入力します。これを使用して、安全な SSL 接続で AEM オンプレミスに接続します。

   c. 承認 — ACL があり、それをアクセスコントロールに使用する場合は、ドキュメントのアクセスコントロールリスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

d. [認証] - [基本認証] または [OAuth 2.0 認証] を選択します。次に、既存の AWS Secrets Manager シークレットを選択するか、新しいシークレットを作成してAdobe Experience Manager認証情報を保存します。新しいシークレットを作成する場合は、 AWS Secrets Manager シークレットウィンドウが開きます。

   [基本認証] を選択した場合は、シークレットの名前、Adobe Experience Manager サイトユーザー名、パスワードを入力します。ユーザーは管理者アクセス許可を持っているか、管理者ユーザーである必要があります。

   [OAuth 2.0 認証] を選択し、AEM オンプレミスを使用する場合は、シークレットの名前、クライアント ID、クライアントシークレット、およびプライベートキーを入力します。AEM をクラウドサービスとして使用する場合は、シークレット、クライアント ID、クライアントシークレット、プライベートキー、組織 ID、テクニカルアカウント ID、および Adobe Identity Management System (IMS) ホストの名前を入力します。

   [保存] を選択します。

e. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

f. ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

g. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

h. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] - 特定のコンテンツタイプ、ページコンポーネント、ルートパスのクロールに制限を設定し、正規表現パターンを使用してコンテンツをフィルタリングします。

i. [コンテンツタイプ] - ページまたはアセットのみ、あるいはその両方をクロールするかを選択します。

ii. (オプション) [その他の設定] で、次のオプションフィールドを設定します。

- [ページコンポーネント] - ページコンポーネントの特定の名前。ページコンポーネントは、Adobe Experience Manager テンプレートエディタと連携するように設計された拡張可能なページコンポーネントで、ページヘッダー/フッターコンポーネントと構造コンポーネントをテンプレートエディタで組み立てることができます。
- [コンテンツフラグメントバリエーション] - コンテンツフラグメントバリエーションの具体的な名前。コンテンツフラグメントを使用すると、Adobe Experience Manager でページに依存しないコンテンツをデザイン、作成、キュレーション、公開できます。これにより、複数の場所や複数のチャネルですぐに使用できるコンテンツを準備できます。
- [ルートパス] - 特定のコンテンツへのルートパス。
- [正規表現パターン] - 特定のページやアセットを含めるまたは除外する正規表現パターン。

b. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを Amazon Kendra 初めて と同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

c. [タイムゾーン ID] - EM オンプレミスを使用していて、サーバーのタイムゾーンが Amazon Kendra AEM コネクタまたはインデックスのタイムゾーンと異なる場合は、AEM コネクタまたはインデックスに合わせてサーバーのタイムゾーンを指定できます。AEM オンプレミスのデフォルトのタイムゾーンは、 Amazon Kendra AEM コネク

タまたはインデックスのタイムゾーンです。クラウドサービスとしての AEM のデフォルトのタイムゾーンはグリニッジ標準時です。

d. [同期実行スケジュール] - [頻度] で、 Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. インデックスにマッピングする Amazon Kendra 生成されたデフォルトのデータソースフィールドから選択します。カスタムデータソースフィールドを追加するには、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra に接続するには Adobe Experience Manager

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — JSON スキーマを使用するAEMときにデータソースタイプを TemplateConfiguration として指定します。また、 CreateDataSource API を呼び出すTEMPLATEときにデータソースを として指定します。
- AEM ホスト URL - Adobe Experience Manager ホスト URL を指定します。例えば、AEM オンプレミスを使用する場合は、ホスト名とポートを含めます。https://hostname:port。または、AEM をクラウドサービスとして使用する場合は、作成者 URL を使用できます。https://author-xxxxxx-xxxxxxx.adobeaemcloud.com。
- 同期モード - がすべてのドキュメントを同期するか、新規、変更、削除されたドキュメントのみを同期するかを指定して、インデックス Amazon Kendra を更新します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。

  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- 認証タイプ - 使用する認証タイプを指定します (Basic または OAuth2)。
- AEM タイプ - 使用する Adobe Experience Manager のタイプを指定します (CLOUD または ON_PREMISE)。
- シークレットの Amazon リソースネーム (ARN) - AEM オンプレミスまたは Cloud のいずれかで基本認証を使用する場合は、ユーザー名とパスワードの認証情報を保存するシークレットを指定します。 AWS Secrets Manager シークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "aemUrl": "Adobe Experience Manager On-Premise host URL",
    "username": "user name with admin permissions",
    "password": "password with admin permissions"
}
```

AEM オンプレミスに OAuth 2.0 認証を使用する場合、シークレットは次のキーを含む JSON 構造に保存されます。

```
{
    "aemUrl": "Adobe Experience Manager host URL",
    "clientId": "client ID",
    "clientSecret": "client secret",
    "privateKey": "private key"
}
```

AEM の OAuth 2.0 認証をクラウドサービスとして使用する場合、シークレットは次のキーを含む JSON 構造に保存されます。

```
{
    "clientId": "client ID",
    "clientSecret": "client secret",
    "privateKey": "private key",
    "orgId": "organization ID",
    "technicalAccountId": "technical account ID",
    "imsHost": "Adobe Identity Management System (IMS) host"
}
```

- role IAM — を呼び出しCreateDataSourceて、 Secrets Manager シークレットにアクセスするためのアクセス許可を IAM ロールに付与し、Adobe Experience Manager コネクタと に必要なパブリック APIs を呼び出すRoleArnタイミングを指定します Amazon Kendra。詳細については、「IAM roles for Adobe Experience Manager data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- タイムゾーン ID — AEM オンプレミスを使用し、サーバーのタイムゾーンが Amazon Kendra AEM コネクタまたはインデックスのタイムゾーンと異なる場合は、AEM コネクタまたはインデックスに合わせてサーバーのタイムゾーンを指定できます。

  AEM オンプレミスのデフォルトのタイムゾーンは、 Amazon Kendra AEM コネクタまたはインデックスのタイムゾーンです。クラウドサービスとしての AEM のデフォルトのタイムゾーンはグリニッジ標準時です。

  サポートされているタイムゾーン ID の詳細については、「Adobe Experience Manager JSON schema」を参照してください。
- 包含フィルターと除外フィルター - 特定のページやアセットを含めるか除外するかを指定します。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、 Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできま

す。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

- フィールドマッピング - Adobe Experience Manager のデータソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> がドキュメント Amazon Kendra を検索するには、ドキュメント本文フィールドまたはドキュメントと同等のドキュメント本文が必要です。データソース内のドキュメント本文フィールド名をインデックスフィールド名 にマッピングする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「Adobe Experience Manager template schema」を参照してください。

Alfresco

Alfresco は、お客様のコンテンツの保存と管理を支援するコンテンツ管理サービスです。を使用して Amazon Kendra 、Alfrescoドキュメントライブラリ、Wiki、ブログのインデックスを作成できます。

Amazon Kendra は、AlfrescoオンプレミスとAlfrescoクラウド (Platform as a Service) をサポートしています。

Amazon Kendra コンソールまたは TemplateConfiguration API を使用して、AlfrescoデータソースAmazon Kendra に接続できます。

Amazon Kendra Alfresco データソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順

- 詳細はこちら

サポートされている機能

Amazon Kendra Alfresco データソースコネクタは以下の機能をサポートしています。

- フィールドマッピング
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期
- 仮想プライベートクラウド (VPC)
- ユーザーコンテキストフィルタリング
- OAuth 2.0 と基本的な認証

前提条件

Amazon Kendra を使用して Alfresco データソースのインデックスを作成する前に、 Alfrescoおよびでこれらの変更を行ってください AWS アカウント。

Alfresco で以下を確認してください。

- Alfresco リポジトリ URL とウェブアプリケーション URL をコピー済み。特定の Alfresco サイトのみのインデックスを作成する場合は、サイト ID もコピーしてください。
- 少なくとも読み取りアクセス許可のあるユーザー名とパスワードを含む Alfresco 認証情報を記録しました。OAuth 2.0 認証を使用する場合は、ユーザーを Alfresco 管理者グループに追加する必要があります。
- オプション: Alfresco で OAuth 2.0 認証情報を生成しました。認証情報には、クライアント ID、クライアントシークレット、およびトークン URL が含まれます。クライアントを Alfresco オンプレミス用に設定する方法について詳しくは、Alfresco のドキュメントを参照してください。Alfresco クラウド (PaaS) を使用している場合は、Hyland サポートに連絡して Alfresco OAuth 2.0 認証を受ける必要があります。
- 各ドキュメントが Alfresco および同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

で AWS アカウント、以下があることを確認します。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。
- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- Alfresco 認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合はシークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、Alfresco データソースを に接続するときに、コンソールを使用して新しい IAM ロールと Secrets Manager シークレットを作成できます Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Alfresco データソース Amazon Kendra に接続するには、 がデータ Amazon Kendra にアクセスできるように Alfresco データソースの必要な詳細を入力する必要があります。の Alfresco をまだ設定していない場合は Amazon Kendra、「」を参照してください前提条件。

Console

Amazon Kendra に接続するには Alfresco

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加ページで Alfresco コネクタ を選択し、コネクタ を追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 - インデックスのドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. Alfresco タイプ - Alfresco オンプレミスを使用するか Alfresco クラウド (Platform as a Service) を使用するかを選択します。

   b. Alfresco リポジトリ URL - Alfresco リポジトリ URL を入力します。例えば、Alfresco クラウド (PaaS) を使用している場合、リポジトリ URL は、https://company.alfrescocloud.com になる可能性があります。または、Alfresco オンプレミスを使用している場合は、リポジトリ URL は https://company-alfresco-instance.company-domain.suffix:port になる可能性があります。

   c. Alfresco ユーザーアプリケーション。URL - Alfresco ユーザーインターフェイスの URL を入力します。リポジトリ URL は Alfresco 管理者から取得できます。例えば、ユーザーインターフェイス URL は https://example.com とすることができます。

   d. SSL 証明書の場所 — Amazon S3 バケットに保存されている SSL 証明書へのパスを入力します。これを使用して、安全な SSL 接続で Alfresco オンプレミスに接続します。

   e. 承認 — ACL があり、それをアクセスコントロールに使用する場合は、ドキュメントのアクセスコントロールリスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザー

とグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

f. [認証] - [基本認証] または [OAuth 2.0 認証] を選択します。次に、既存の Secrets Manager シークレットを選択するか、新しいシークレットを作成して Alfresco 認証情報を保存します。新しいシークレットを作成する場合は、 AWS Secrets Manager シークレットウィンドウが開きます。

[基本認証] を選択した場合は、シークレットの名前、Alfresco サイトユーザー名、パスワードを入力します。

[OAuth 2.0 認証]を選択した場合は、シークレットの名前、クライアント ID、クライアントシークレット、トークン URL を入力します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

i. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

j. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] - 特定のコンテンツのクロールに制限を設定し、正規表現パターンを使用してコンテンツをフィルタリングします。

b. i. [コンテンツ] - Alfresco の「アスペクト」とマークされたコンテンツ、特定の Alfresco サイト内のコンテンツ、またはすべての Alfresco サイトにわたるコンテンツをクロールするかどうかを選択します。

ii. (オプション) [その他の設定] - 以下の設定を設定します。

- [コメントを含める] - Alfresco ドキュメントライブラリとブログにコメントを含めるかどうかを選択します。
- [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。

c. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを Amazon Kendra 初めて と同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. 同期実行スケジュール で、頻度 - Amazon Kendra データソースと同期する頻度を選択します。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. インデックスにマッピングする、 Amazon Kendra 生成されたデフォルトのデータソースフィールドから選択します。

b. カスタムデータソースフィールドを追加するには、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra に接続するには Alfresco

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — JSON スキーマを使用するALFRESCOときに、データソースタイプを TemplateConfiguration として指定します。また、 CreateDataSource API を呼び出すTEMPLATEときにデータソースを として指定します。
- Alfresco サイト ID - Alfresco サイト ID を指定します。
- Alfresco リポジトリ URL - Alfresco リポジトリ URL を指定します。リポジトリ URL は Alfresco 管理者から取得できます。例えば、Alfresco クラウド (PaaS) を使用している場合、リポジトリ URL は、https://company.alfrescocloud.com になる可能性があります。または、Alfresco オンプレミスを使用している場合は、リポジトリ URL は https://company-alfresco-instance.company-domain.suffix:port になる可能性があります。
- Alfresco ウェブアプリケーション URL - Alfresco ユーザーインターフェイス URL を指定します。リポジトリ URL は Alfresco 管理者から取得できます。例えば、ユーザーインターフェイス URL は https://example.com とすることができます。
- 認証タイプ - 使用する認証タイプを指定します (OAuth2 または Basic)。
- Alfresco タイプ - 使用する PAAS のタイプを指定します。Alfresco (クラウド/Platform as a Service) または ON_PREM (オンプレミス) のいずれかを使用するかを指定します。
- シークレットの Amazon リソースネーム (ARN) - 基本認証を使用する場合は、ユーザー名とパスワードの認証情報を保存するシークレットを指定します。 AWS Secrets Manager シークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password"
}
```

OAuth 2.0 認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "clientId": "client ID",
    "clientSecret": "client secret",
    "tokenUrl": "token URL"
}
```

- role IAM — を呼び出しCreateDataSourceて、シー Secrets Manager クレットにアクセスするためのアクセス許可を IAM ロールに付与し、Alfresco コネクタと に必要なパブリック APIs を呼び出すRoleArnタイミングを指定します Amazon Kendra。詳細については、「IAM roles for Alfresco data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- コンテンツタイプ - クロールするコンテンツのタイプ。Alfresco の「アスペクト」でマークされたコンテンツ、特定のAlfresco サイト内のコンテンツ、またはすべての Alfresco サイトにわたるコンテンツ。特定の「アスペクト」コンテンツを一覧表示することもできます。
- 包含フィルターと除外フィルター - 特定のファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- 同期モード - がすべてのドキュメントを同期するか、新規、変更、削除されたドキュメントのみを同期するかを指定して、インデックス Amazon Kendra を更新します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、 Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできま

す。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

- フィールドマッピング - データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> がドキュメント Amazon Kendra を検索するには、ドキュメント本文フィールドまたはドキュメントと同等のドキュメント本文が必要です。データソース内のドキュメント本文フィールド名をインデックスフィールド名 にマッピングする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「Alfresco template schema」を参照してください。

詳細はこちら

Amazon Kendra と Alfresco データソースの統合の詳細については、以下を参照してください。

- を使用してAlfrescoコンテンツをインテリジェントに検索する Amazon Kendra

Aurora (MySQL)

Aurora はクラウド向けに構築されたリレーショナルデータベース管理システム (RDBMS) です。Aurora ユーザーであれば、 Amazon Kendra を使用してデータソースのインデックスを作成できます。Aurora (MySQL) Amazon Kendra Aurora (MySQL)データソースコネクタは Aurora MySQL 3 Aurora とサーバーレス MySQL 8.0 をサポートします。

Amazon Kendra コンソールと API Amazon Kendra Aurora (MySQL) を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Aurora (MySQL)データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Aurora (MySQL)を使用してデータソースのインデックスを作成する前に、Aurora (MySQL) AWS とアカウントでこれらの変更を行ってください。

Aurora (MySQL) で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。 Amazon RDS この情報はコンソールで確認できます。
- 各ドキュメントが Aurora (MySQL) および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

に AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Aurora (MySQL) の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Aurora (MySQL)データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Aurora (MySQL)Aurora (MySQL) Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Aurora (MySQL) Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Aurora (MySQL)

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。

2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Aurora (MySQL)コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] には、次の情報を入力します。

   b. [ホスト] – データベースのホスト URL を入力します (例: `http://`*`instance URL`*`.`*`region`*`.rds.amazonaws.com`)。

   c. [ポート] – データベースポートを入力します (例: 5432)。

   d. [インスタンス] - データベースインスタンスを入力します。

   e. [認証] には、次の情報を入力します。

      - AWS Secrets Manager secret — 既存のシークレットを選択するか、 Secrets Manager Aurora (MySQL)認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraAurora (MySQL)-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

f. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

g. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

h. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。SQL クエリは 32 KB 未満で、セミコロン (;) を含めないでください。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Aurora (MySQL)

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを mySql として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Aurora (MySQL) シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
```

```
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。Aurora (MySQL) Amazon Kendra詳細については、「IAM roles for Aurora (MySQL) data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Aurora (MySQL) データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Aurora (MySQL) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Aurora (PostgreSQL)

Aurora はクラウド向けに構築されたリレーショナルデータベース管理システム (RDBMS) です。Aurora ユーザーであれば、 Amazon Kendra を使用してデータソースのインデックスを作成できます。Aurora (PostgreSQL) Amazon Kendra Aurora (PostgreSQL)データソースコネクタは Aurora PostgreSQL 1 をサポートしています。

Amazon Kendra コンソールと API Amazon Kendra Aurora (PostgreSQL) を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Aurora (PostgreSQL)データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Aurora (PostgreSQL)を使用してデータソースのインデックスを作成する前に、Aurora (PostgreSQL) AWS とアカウントでこれらの変更を行ってください。

Aurora (PostgreSQL) で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが Aurora (PostgreSQL) および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Aurora (PostgreSQL) の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Aurora (PostgreSQL)データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Aurora (PostgreSQL)Aurora (PostgreSQL) Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Aurora (PostgreSQL) Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Aurora (PostgreSQL)

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Aurora (PostgreSQL)コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [ソース] には、次の情報を入力します。
   b. [ホスト] – データベースのホスト URL を入力します (例: `http://instance URL.region.rds.amazonaws.com`)。
   c. [ポート] – データベースポートを入力します (例: `5432`)。
   d. [インスタンス] – データベースインスタンスを入力します (例: `postgres`)。
   e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。
   f. [認証] には、次の情報を入力します。
      - AWS Secrets Manager secret — Aurora (PostgreSQL) 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraAurora (PostgreSQL)-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。SQL クエリは 32 KB 未満で、セミコロン (;) を含めないでください。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Aurora (PostgreSQL)

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを postgresql として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Aurora (PostgreSQL)シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
```

```
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。Aurora (PostgreSQL) Amazon Kendra詳細については、「IAM roles for Aurora (PostgreSQL) data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Aurora (PostgreSQL) データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Aurora (PostgreSQL) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Amazon FSx (ウィンドウズ)

Amazon FSx (Windows) は、共有ストレージ機能を提供するフルマネージド型のクラウドベースのファイルサーバーシステムです。 Amazon FSx (Windows) ユーザーの場合は、 Amazon Kendra を使用して Amazon FSx (Windows) データソースのインデックスを作成できます。

> **Note**
>
> Amazon Kendra アップグレードされた Amazon FSx (Windows) コネクタをサポートするようになりました。
> コンソールは自動的にアップグレードされました。コンソールに新しいコネクタを作成すると、アップグレードされたアーキテクチャが使用されます。API を使用する場合は、TemplateConfigurationオブジェクトではなくオブジェクトを使用してコネクタを設定する必要があります。`FSxConfiguration`
> 古いコンソールと API アーキテクチャを使用して設定されたコネクタは、引き続き設定どおりに機能します。ただし、編集や更新はできません。コネクタ構成を編集または更新する場合は、新しいコネクタを作成する必要があります。

コネクタワークフローをアップグレードされたバージョンに移行することをお勧めします。古いアーキテクチャを使用して構成されたコネクタSupport は、2024 年 6 月までに終了する予定です。

Amazon Kendra コンソールまたは TemplateConfigurationAPI を使用して Amazon FSx (Windows) Amazon Kendra データソースに接続できます。

Amazon Kendra Amazon FSx (Windows) データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Amazon FSx (Windows) データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーアクセス制御
- ユーザー ID クローリング
- 包含フィルターと除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Amazon FSx (Windows) データソースのインデックスを作成する前に、Amazon FSx (Windows) との詳細を確認してください。 AWS アカウント

Amazon FSx (Windows) の場合は、以下の点を確認してください。

- Amazon FSx (Windows) に読み取り権限とマウント権限を設定してください。

- ファイルシステム ID を書き留めました。ファイルシステム ID は Amazon FSx (Windows) コンソールのファイルシステムダッシュボードで確認できます。
- Amazon FSx (Windows) Amazon VPC ファイルシステムが置かれている場所を使用して仮想プライベートクラウドを設定しました。
- Active Directoryユーザーアカウントの Amazon FSx (Windows) 認証情報を書き留めました。これには、Active Directory ユーザー名、DNS ドメイン名 (user@corp.example.com など)、およびパスワードが含まれます。

> **Note**
>
> コネクタが機能するために必要な認証情報のみを使用してください。ドメイン管理者などの特権認証情報は使用しないでください。

- 各ドキュメントが Amazon FSx (Windows) と、同じインデックスに使用する予定の他のデータソース間で一意であることを確認した。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon FSx (Windows) AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールまたはシークレットがない場合は、 Amazon FSx (Windows) データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon FSx (Windows) Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Amazon FSx (Windows) データソースの必要な詳細情報を入力する必要があります。 Amazon FSx (Windows) をまだ設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Amazon FSx (Windows) Amazon Kendra ファイルシステムに接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Amazon FSx (Windows) コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. Amazon FSx (Windows) ファイルシステム ID — (Windows) から取得した既存のファイルシステム ID Amazon FSx をドロップダウンから選択します。または Amazon FSx (Windows) ファイルシステムを作成します。ファイルシステム ID は Amazon FSx (Windows) コンソールのファイルシステムダッシュボードで確認できます。

b. 承認 — ACL があり、それをアクセス制御に使用したい場合は、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

c. 認証 — AWS Secrets Manager 既存のシークレットを選択するか、ファイルシステムの認証情報を保存する新しいシークレットを作成します。新しいシークレットを作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。

ユーザー名とパスワードの認証情報を保存するシークレットを指定します。ユーザー名には DNS ドメイン名を含める必要があります。例えば、user@corp.example.com と入力します。

シークレットを保存して追加します。

d. Virtual Private Cloud (VPC) — Amazon FSx (Windows) Amazon VPC が置かれている場所を選択する必要があります。VPC サブネットとセキュリティグループを含めます。「の設定」を参照してください。 Amazon VPC

e. IAM ロール — IAM 既存のロールを選択するか、 IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. 同期スコープ、正規表現パターン-特定のファイルを含めたり除外したりする正規表現パターンを追加します。

   b. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

      - [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
      - [新規、変更済み、または削除されたコンテンツを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

   c. 同期実行スケジュール — [頻度] では、データソースのコンテンツを同期してインデックスを更新する頻度を選択します。

   d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. Amazon Kendra 生成されたファイルのデフォルトフィールドの中から、インデックスにマップしたいものを選択します。カスタムデータソースフィールドを追加するには、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon FSx (Windows) Amazon Kendra ファイルシステムに接続するには

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON FSX スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- ファイルシステム ID — Amazon FSx (Windows) ファイルシステムの識別子。ファイルシステム ID は Amazon FSx (Windows) コンソールのファイルシステムダッシュボードで確認できます。
- ファイルシステムタイプ - ファイルシステムのタイプを WINDOWS として指定します。
- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。

  > **Note**
  >
  > Amazon FSx (Windows) Amazon VPC が置かれている場所を選択する必要があります。VPC サブネットとセキュリティグループを含めます。

- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

- シークレットアマゾンリソースネーム (ARN) — (Windows) Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム Amazon FSx (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "user@corp.example.com",
    "password": "password"
}
```

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、 Amazon FSx (Windows) コネクタとに必要なパブリック API `RoleArn` を呼び出す際に、呼び出しのタイミングを指定します。 Amazon Kendra詳細については、「Amazon FSx (Windows)IAM データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- 包含フィルターと除外フィルター - 特定のファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- アクセス制御リスト (ACL) — ACL があり、それをアクセス制御に使用したい場合に、ドキュメントの ACL 情報をクロールするかどうかを指定します。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

> **Note**
>
> ユーザーに対してユーザーコンテキストフィルタリングをテストするには、クエリを発行するときにユーザー名の一部として DNS ドメイン名を含める必要がありま

す。Active Directory ドメインの管理用のアクセス許可が必要です。グループ名でユーザーコンテキストフィルタリングをテストすることもできます。

- フィールドマッピング — Amazon FSx (Windows) Amazon Kendra データソースフィールドをインデックスフィールドにマップすることを選択します。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「Amazon FSx (Windows) テンプレートスキーマ」を参照してください。

詳細はこちら

Amazon FSx (Windows) Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Amazon FSx (Windows) for Amazon Kendra 用コネクタを使用して Windows ファイルシステム上の非構造化データを安全に検索します。Windows File Server

Amazon FSx (NetApp ONTAP)

Amazon FSx (NetApp ONTAP) は、共有ストレージ機能を提供するフルマネージド型のクラウドベースのファイルサーバシステムです。 Amazon FSx (NetApp ONTAP) ユーザーの場合は、 Amazon Kendra を使用して Amazon FSx (NetApp ONTAP) データソースのインデックスを作成できます。

Amazon Kendra コンソールまたは Amazon Kendra API を使用して Amazon FSx (NetApp ONTAP) データソースに接続できます。TemplateConfiguration

Amazon Kendra Amazon FSx (NetApp ONTAP) データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

サポートされている機能

Amazon Kendra Amazon FSx (NetApp ONTAP) データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーアクセス制御
- 包含フィルターと除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Amazon FSx (NetApp ONTAP) データソースのインデックスを作成する前に、(ONTAP) との詳細を確認してください。 Amazon FSx NetApp AWS アカウント

Amazon FSx (NetApp ONTAP) の場合は、次のものが揃っていることを確認してください。

- 読み取り権限とマウント権限で Amazon FSx (NetApp ONTAP) を設定します。
- ファイルシステム ID を書き留めました。ファイルシステム ID は、 Amazon FSx (NetApp ONTAP) コンソールのファイルシステムダッシュボードで確認できます。
- ファイルシステムで使用されているストレージ仮想マシン（ SVM ）ID を記録しました。SVM ID は、 Amazon FSx （ NetApp ONTAP ）コンソールのファイルシステムダッシュボードでファイルシステム ID を選択し、次に [ストレージ仮想マシン] を選択すると確認できます。
- Amazon FSx (NetApp ONTAP) Amazon VPC ファイルシステムが置かれている場所を使用して仮想プライベートクラウドを設定しました。
- ユーザアカウントの Amazon FSx (NetApp ONTAP) 認証情報を書き留めました。Active Directory これには、Active Directory ユーザ名、DNS ドメイン名 (user@corp.example.com など)、およびパスワードが含まれます。(NetApp ONTAP) ファイルシステムにネットワークファイルシステム

(NFS) プロトコルを使用する場合、認証資格情報には左の ID、右 ID、および事前共有キーが含まれます。 Amazon FSx

> **Note**
>
> コネクタが機能するために必要な認証情報のみを使用してください。ドメイン管理者のような特権認証情報は使用しないでください。

- Amazon FSx （ NetApp ONTAP ）内でも、同じインデックスに使用する予定の他のデータソースでも、各ドキュメントが一意であることを確認。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon FSx （ NetApp ONTAP ） AWS Secrets Manager 認証クレデンシャルをシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 Amazon FSx (NetApp ONTAP) データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成

できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon FSx (NetApp ONTAP) Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Amazon FSx (NetApp ONTAP) データソースの必要な詳細情報を入力する必要があります。 Amazon FSx (NetApp ONTAP) をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Amazon FSx (NetApp ONTAP) Amazon Kendra ファイルシステムに接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Amazon FSx (NetApp ONTAP) コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-インデックス用のドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. ソース — ファイルシステム情報を入力します。

- ファイルシステムプロトコル — Amazon FSx (NetApp ONTAP) ファイルシステムのプロトコルを選択します。Linux では、共通インターネットファイルシステム (CIFS) プロトコルか、ネットワークファイルシステム (NFS) プロトコルのいずれかを選択できます。
- Amazon FSx (NetApp ONTAP) ファイルシステム ID — (ONTAP) から取得した既存のファイルシステム ID をドロップダウンから選択します。 Amazon FSx NetApp または、Amazon FSx (ONTAP) NetApp ファイルシステムを作成します。ファイルシステム ID は、 Amazon FSx (NetApp ONTAP) コンソールのファイルシステムダッシュボードで確認できます。
- SVM IDAmazon FSx (NetApp ONTAP) NetApp ONTAP のみ) — (ONTAP) のストレージ仮想マシン (SVM) ID を入力します。 Amazon FSx NetApp NetApp ONTAPSVM ID は、 Amazon FSx (NetApp ONTAP) コンソールのファイルシステムダッシュボードに移動し、ファイルシステム ID を選択して [ストレージ仮想マシン] を選択すると確認できます。

b. 承認 — ACL があり、それをアクセス制御に使用したい場合は、ドキュメントのアクセス制御リスト（ ACL ）情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

c. 認証 — AWS Secrets Manager 既存のシークレットを選択するか、ファイルシステムの認証情報を保存する新しいシークレットを作成します。新しいシークレットを作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。

ユーザー名とパスワードの認証情報を保存するシークレットを指定します。ユーザー名には DNS ドメイン名を含める必要があります。例えば、user@corp.example.com と入力します。

Amazon FSx （ NetApp ONTAP ）ファイルシステムに NFS プロトコルを使用する場合は、左 ID、右 ID、事前共有キーの認証情報を保存するシークレットを指定します。

シークレットを保存して追加します。

d. Virtual Private Cloud (VPC)： Amazon FSx （ NetApp ONTAP ） Amazon VPC が存在する場所を選択する必要があります。VPC サブネットとセキュリティグループを含めます。「の設定」を参照してください。 Amazon VPC

e. IAM ロール — IAM 既存のロールを選択するか、 IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. 同期スコープ、正規表現パターン-特定のファイルを含めたり除外したりする正規表現パターンを追加します。

   b. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

      - [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
      - [新規、変更済み、または削除されたコンテンツを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

   c. 同期実行スケジュール — [頻度] では、データソースのコンテンツを同期してインデックスを更新する頻度を選択します。

   d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. Amazon Kendra 生成されたファイルのデフォルトフィールドの中から、インデックスにマップしたいものを選択します。カスタムデータソースフィールドを追加するには、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon FSx (NetApp ONTAP) Amazon Kendra ファイルシステムに接続するには

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON FSXONTAP スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- ファイルシステム ID — Amazon FSx (NetApp ONTAP) ファイルシステムの識別子。ファイルシステム ID は、 Amazon FSx (NetApp ONTAP) コンソールのファイルシステムダッシュボードで確認できます。
- SVM ID — ファイルシステムで使用されるストレージ仮想マシン（ SVM ）ID。SVM ID は、Amazon FSx （ NetApp ONTAP ）コンソールのファイルシステムダッシュボードでファイルシステム ID を選択し、次に [ストレージ仮想マシン] を選択すると確認できます。
- プロトコルタイプ — Linux の共通インターネットファイルシステム（ CIFS ）プロトコルを使用するか、ネットワークファイルシステム（ NFS ）プロトコルを使用するかを指定します。
- ファイルシステムタイプ-ファイルシステムのタイプをどちらかに指定します。FSXONTAP
- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。

  > **Note**
  >
  > Amazon FSx (NetApp ONTAP) Amazon VPC が置かれている場所を選択する必要があります。VPC サブネットとセキュリティグループを含めます。

- シークレットアマゾンリソースネーム（ ARN ）—お使いの（ NetApp ONTAP ） Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム Amazon FSx （ ARN ）を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "user@corp.example.com",
    "password": "password"
}
```

Amazon FSx (NetApp ONTAP) ファイルシステムに NFS プロトコルを使用する場合、シークレットは次のキーを含む JSON 構造で保存されます。

```
{
    "leftId": "left ID",
    "rightId": "right ID",
    "preSharedKey": "pre-shared key"
}
```

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、 Amazon FSx (NetApp ONTAP) コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「Amazon FSx (NetApp ONTAP)IAM データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- 同期モード-すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。
  - `FORCED_FULL_CRAWL` は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - `FULL_CRAWL` は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- 包含フィルターと除外フィルター - 特定のファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- アクセス制御リスト (ACL) — ACL があり、それをアクセス制御に使用したい場合に、ドキュメントの ACL 情報をクロールするかどうかを指定します。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュ

メントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

> **Note**
>
> ユーザーに対してユーザーコンテキストフィルタリングをテストするには、クエリを発行するときにユーザー名の一部として DNS ドメイン名を含める必要があります。Active Directory ドメインの管理用のアクセス許可が必要です。グループ名でユーザーコンテキストフィルタリングをテストすることもできます。

- フィールドマッピング — Amazon FSx (NetApp ONTAP) データソースフィールドをインデックスフィールドにマップすることを選択します。 Amazon Kendra 詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、Amazon FSx (NetApp ONTAP) テンプレートスキーマを参照してください。

Amazon RDS/Aurora

データベースデータソースを使用して、データベースに保存されているドキュメントにインデックスを作成することができます。データベースの接続情報を入力すると、 Amazon Kendra ドキュメントを接続してインデックスを作成します。

Amazon Kendra 以下のデータベースをサポートします。

- Amazon Aurora MySQL
- Amazon Aurora PostgreSQL
- Amazon RDS MySQL 用
- Amazon RDS PostgreSQL 用

> **Note**
>
> サーバーレス Aurora データベースはサポートされていません。

> **Important**
>
> この Amazon RDS/Aurora コネクタは、2023 年末までに廃止される予定です。
> Amazon Kendra 新しいデータベースデータソースコネクタをサポートするようになりました。エクスペリエンスを向上させるために、ユースケースに応じて次の新しいコネクタから選択することをお勧めします。
>
> - Aurora (MySQL)
> - Aurora (PostgreSQL)
> - Amazon RDS (MySQL)
> - Amazon RDS (Microsoft SQL サーバー)
> - Amazon RDS (オラクル)
> - Amazon RDS (PostgreSQL)
> - IBM DB2
> - Microsoft SQL Server
> - MySQL
> - Oracle Database
> - PostgreSQL

Amazon Kendra コンソールと API Amazon Kendra を使用してデータベースデータソースに接続できます。DatabaseConfiguration

Amazon Kendra データベースデータソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順

サポートされている機能

Amazon Kendra データベースデータソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用してデータベースデータソースのインデックスを作成する前に、 AWS データベースとアカウントにこれらの変更を加えてください。

データベースで以下を確認してください。

- データベースのユーザー名とパスワードの基本認証情報を記録しました。
- ホスト名、ポート番号、ホストアドレス、データベース名、ドキュメントデータが含まれているデータテーブルの名前をコピーしました。PostgreSQL の場合、データテーブルはパブリックテーブルまたはパブリックスキーマである必要があります。

> **Note**
>
> ホストとポートは、 Amazon Kendra インターネット上のデータベースサーバーの場所を示します。データベース名とテーブル名は、 Amazon Kendra データベースサーバー上のドキュメントデータの保存場所を示します。

- ドキュメントデータを含むデータテーブル内の列の名前をコピーしました。ドキュメント ID、ドキュメント本文、ドキュメントが変更されたかどうかを検出する列 (最終更新列など)、カスタムインデックスフィールドにマッピングされるオプションのデータテーブル列を含める必要があります。Amazon Kendra の予約済みフィールド名をテーブルの列にマッピングできます。
- MySQL に使用するのか、 Amazon RDS 別のタイプに使用するのかなど、データベースエンジンのタイプ情報をコピーしました。
- 各ドキュメントがデータベースおよび同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- データベースの認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、データベースデータソースをに接続するときに、IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データベースデータソースに接続するには、 Amazon Kendra データにアクセスできるようにデータベースデータソースに関する必要な詳細情報を入力する必要があります。のデータベースをまだ設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Amazon Kendra データベースに接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。

2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [データベースコネクター] を選択し、[コネクターの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [エンドポイント] - DNS ホスト名、IPv4 アドレス、または IPv6 アドレス。

   b. [ポート] - ポート番号。

   c. [データベース] - データベース名。

   d. [テーブル名] - テーブル名。

   e. [認証のタイプ] で [既存] または [新規] を選択してデータベース認証情報を保存します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      - [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-database-' がシークレット名に自動的に追加されます。

B. [ユーザー名] と [パスワード] - データベースアカウントから認証情報の値を入力します。

C. [認証を保存] を選択します。

f. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

> **Note**
>
> プライベートサブネットを使用する必要があります。RDS インスタンスが VPC のパブリックサブネットにある場合は、パブリックサブネット内の NAT ゲートウェイへのアウトバウンドアクセス権を持つプライベートサブネットを作成します。VPC 設定で指定するサブネットは、米国西部 (オレゴン)、米国東部 (バージニア北部)、欧州 (アイルランド) のいずれかに存在する必要があります。

g. IAM role — 既存のロールを選択するか、 IAM 新しいロールを作成して、 IAM リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

h. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. ユースケースに基づいて [Aurora MySQL]、[MySQL]、[Aurora PostgreSQL]、[PostgreSQL] の中から選択してください。

b. [SQL 識別子を二重引用符で囲む] - 選択して SQL 識別子を二重引用符で囲んでください。例えば、“columnName”。

c. ACL 列と変更検出列 — Amazon Kendra 変更検出に使用する列 (最終更新列など) とアクセス制御リストを設定します。

d. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. Amazon Kendra デフォルトフィールドマッピング — Amazon Kendra 生成されたデフォルトデータソースフィールドの中から、インデックスにマップしたいフィールドを選択します。document_id および document_body の [データベース列] 値を追加する必要があります。

   b. [カスタムフィールドマッピング] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra データベースに接続するには

次の DatabaseConfigurationAPI を指定する必要があります。

- ColumnConfiguration—インデックスがデータベースからドキュメント情報を取得する場所に関する情報。詳細については、「ColumnConfiguration」を参照してください。DocumentDataColumnName (ドキュメント本文または本文)、DocumentIdColumnName、および ChangeDetectingColumn (最終更新列など) フィールドを指定する必要があります。DocumentIdColumnName フィールドにマッピングされた列は整数の列である必要があります。次の例は、データベースデータソースの単純な列構成を示しています。

```
"ColumnConfiguration": {
    "ChangeDetectingColumns": [
        "LastUpdateDate",
        "LastUpdateTime"
    ],
    "DocumentDataColumnName": "TextColumn",
    "DocumentIdColumnName": "IdentifierColumn",
    "DocoumentTitleColumnName": "TitleColumn",
    "FieldMappings": [
        {
            "DataSourceFieldName": "AbstractColumn",
```

```
            "IndexFieldName": "Abstract"
        }
    ]
}
```

- ConnectionConfiguration—データベースへの接続に必要な構成情報。詳細については、「ConnectionConfiguration」を参照してください。
- DatabaseEngineType—データベースを実行するデータベースエンジンのタイプ。DatabaseHostのフィールドは、データベースの Amazon Relational Database Service (Amazon RDS) ConnectionConfiguration インスタンスエンドポイントである必要があります。クラスターエンドポイントを使用しないでください。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager データベースアカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password"
}
```

次の例は、シークレット ARN を含むデータベース設定を示します。

```
"DatabaseConfiguration": {
"ConnectionConfiguration": {
"DatabaseHost": "host.subdomain.domain.tld",
        "DatabaseName": "DocumentDatabase",
        "DatabasePort": 3306,
        "SecretArn": "arn:aws:secretmanager:region:account ID:secret/secret name",
        "TableName": "DocumentTable"
    }
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、データベースコネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for database data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - データソース設定の一部として `VpcConfiguration` を指定します。「VPC を使用するための Amazon Kendra の設定」を参照してください。

> **Note**
>
> プライベートサブネットのみを使用する必要があります。RDS インスタンスが VPC のパブリックサブネットにある場合は、パブリックサブネット内の NAT ゲートウェイへのアウトバウンドアクセス権を持つプライベートサブネットを作成します。VPC 設定で指定するサブネットは、米国西部 (オレゴン)、米国東部 (バージニア北部)、欧州 (アイルランド) のいずれかに存在する必要があります。

- フィールドマッピング - 選択すると、データベースデータソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、 Amazon Kendra ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

Amazon RDS (Microsoft SQL サーバー)

SQL Serverは、Microsoft が開発したデータベース管理システムです。 Amazon RDS for SQL Server を使用すると、SQL Server デプロイメントをクラウドに簡単にセットアップ、運用、スケーリングできます。 Amazon RDS (Microsoft SQL Server) ユーザーであれば、 Amazon RDS (Microsoft SQL Server) Amazon Kendra データソースのインデックスを作成するために使用できます。 Amazon Kendra JDBC データソースコネクタは、Microsoft SQL サーバー 2019 をサポートしています。

Amazon Kendra コンソールと TemplateConfigurationAPI Amazon Kendra を使用して Amazon RDS (Microsoft SQL Server) データソースに接続できます。

Amazon Kendra Amazon RDS (Microsoft SQL Server) データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Amazon RDS (Microsoft SQL Server) データソースのインデックスを作成する前に、 Amazon RDS (Microsoft SQL Server) AWS とアカウントにこれらの変更を加えてください。

Amazon RDS (Microsoft SQL Server) で、次のものが揃っていることを確認します。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが Amazon RDS (Microsoft SQL Server)、および同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon RDS (Microsoft SQL Server) AWS Secrets Manager の認証資格情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールまたはシークレットがない場合は、 Amazon RDS (Microsoft SQL Server) データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシーク

レットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon RDS (Microsoft SQL Server) Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように自分の Amazon RDS (Microsoft SQL Server) 認証情報の詳細を入力する必要があります。まだ Amazon RDS (Microsoft SQL Server) を設定していない場合は、Amazon Kendra を参照してください前提条件。

Console

Amazon RDS (Microsoft SQL サーバー) Amazon Kendra に接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで、[Amazon RDS (Microsoft SQL Server) コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] - データベースのホスト名を入力します。

c. [ポート] - データベースのポートを入力します。

d. [インスタンス] - データベースインスタンスを入力します。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Amazon RDS (Microsoft SQL Server) Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

  A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

     I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Amazon RDS (Microsoft SQL Server)-' がシークレット名に自動的に追加されます。

     II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

  B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。

> **Note**
>
> テーブル名の名前に特殊文字 (英数字以外) が含まれている場合は、テーブル名を角括弧で囲む必要があります。たとえば、`[] ## [*] ######`。my-database-table

- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- 本文列 — データベーステーブル内の文書本文列の名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。

- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon RDS (Microsoft SQL サーバー) Amazon Kendra に接続するには

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON `JDBC` スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI `TEMPLATE` を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを `sqlserver` として指定する必要があります。

- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。

  > **Note**
  >
  > テーブル名の名前に特殊文字 (英数字以外) が含まれている場合は、テーブル名を角括弧で囲む必要があります。たとえば、*`[] ## [*] ######`*。my-database-table

- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレットアマゾンリソースネーム (ARN) — (Microsoft SQL Server) Secrets Manager アカウントで作成した認証認証情報を含むシークレットの Amazon リソースネーム Amazon RDS (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

  ```
  {
      "user name": "database user name",
      "password": "password"
  }
  ```

  > **Note**
  >
  > 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、 Amazon RDS (Microsoft SQL Server) Amazon Kendraコネクタとに必要な

パブリック API `RoleArn` を呼び出すタイミングを指定します。詳細については、「Amazon RDS (Microsoft SQL Server)IAM データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用の ACL がある場合、ドキュメントのアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング — Amazon RDS (Microsoft SQL Server) Amazon Kendra データソースフィールドをインデックスフィールドにマップすることを選択します。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Amazon RDS (Microsoft SQL サーバー) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。

- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Amazon RDS (MySQL)

Amazon RDS (Amazon Relational Database Service) は、クラウドでのリレーショナルデータベースのセットアップ、運用、スケーリングを容易にするウェブサービスです。 AWS Amazon RDS ユーザーであれば、 Amazon Kendra Amazon RDS (MySQL)を使用してデータソースのインデックスを作成できます。 Amazon Kendra データソースコネクタは Amazon RDS MySql 5.6、5.7、8.0 をサポートします。

Amazon Kendra コンソールと API Amazon Kendra Amazon RDS (MySQL) を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Amazon RDS (MySQL)データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Amazon RDS (MySQL)を使用してデータソースのインデックスを作成する前に、Amazon RDS (MySQL) AWS とアカウントでこれらの変更を行ってください。

Amazon RDS (MySQL) で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。 Amazon RDS この情報はコンソールで確認できます。
- 各ドキュメントが Amazon RDS (MySQL) および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

に AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon RDS (MySQL) の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Amazon RDS (MySQL)データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Amazon RDS (MySQL)Amazon RDS (MySQL) Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Amazon RDS (MySQL) Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Amazon RDS (MySQL)

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Amazon RDS (MySQL)コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] – データベースのホスト URL を入力します (例: `http://`*`instance URL`*`.`*`region`*`.rds.amazonaws.com`)。

c. [ポート] – データベースポートを入力します (例: `5432`)。

d. [インスタンス] – データベースインスタンスを入力します (例: `postgres`)。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Amazon RDS (MySQL) 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

  A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

     I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraAmazon RDS (MySQL)-' がシークレット名に自動的に追加されます。

     II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

  B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。SQL クエリは 32 KB 未満で、セミコロン (;) を含めないでください。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。

- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Amazon RDS (MySQL)

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを mySql として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Amazon RDS (MySQL)シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指

定します。Amazon RDS (MySQL) Amazon Kendra詳細については、「IAM roles for Amazon RDS (MySQL) data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- フィールドマッピング - 選択すると、Amazon RDS (MySQL) データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、 Amazon Kendra ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Amazon RDS (MySQL) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。

- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Amazon RDS (Oracle)

Amazon RDS (Amazon Relational Database Service) は、クラウドでのリレーショナルデータベースのセットアップ、運用、スケーリングを容易にするウェブサービスです。 AWS Amazon RDS (Oracle)ユーザーであれば、 Amazon Kendra Amazon RDS (Oracle)を使用してデータソースのインデックスを作成できます。 Amazon Kendra Amazon RDS (Oracle)データソースコネクタは Amazon RDS Oracle データベース 21c、Oracle データベース 19c、Oracle データベース 12c をサポートします。

Amazon Kendra コンソールと API Amazon Kendra Amazon RDS (Oracle) を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Amazon RDS (Oracle)データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期

- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Amazon RDS (Oracle)を使用してデータソースのインデックスを作成する前に、Amazon RDS (Oracle) AWS とアカウントでこれらの変更を行ってください。

Amazon RDS (Oracle) で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが Amazon RDS (Oracle) および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon RDS (Oracle) の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Amazon RDS (Oracle)データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Amazon RDS (Oracle)Amazon RDS (Oracle) Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Amazon RDS (Oracle) Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Amazon RDS (Oracle)

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Amazon RDS (Oracle)コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] - データベースのホスト名を入力します。

c. [ポート] - データベースのポートを入力します。

d. [インスタンス] - データベースインスタンスを入力します。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Amazon RDS (Oracle) 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraAmazon RDS (Oracle)-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。

- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Amazon RDS (Oracle)

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを oracle として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Amazon RDS (Oracle)シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。Amazon RDS (Oracle) Amazon Kendra詳細については、「IAM roles for Amazon RDS (Oracle) data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Amazon RDS (Oracle) データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Amazon RDS (Oracle) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。

- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Amazon RDS (PostgreSQL)

Amazon RDS は、 AWS クラウド内のリレーショナルデータベースのセットアップ、運用、拡張を容易にするウェブサービスです。 Amazon RDS ユーザーであれば、 Amazon Kendra Amazon RDS (PostgreSQL)を使用してデータソースのインデックスを作成できます。 Amazon Kendra Amazon RDS (PostgreSQL)データソースコネクタは PostgreSQL 9.6 をサポートしています。

Amazon Kendra コンソールと API Amazon Kendra Amazon RDS (PostgreSQL) を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Amazon RDS (PostgreSQL)データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Amazon RDS (PostgreSQL)を使用してデータソースのインデックスを作成する前に、Amazon RDS (PostgreSQL) AWS とアカウントでこれらの変更を行ってください。

Amazon RDS (PostgreSQL) で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。 Amazon RDS この情報はコンソールで確認できます。
- 各ドキュメントが Amazon RDS (PostgreSQL) および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

に AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Amazon RDS (PostgreSQL) の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Amazon RDS (PostgreSQL) データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Amazon RDS (PostgreSQL)Amazon RDS (PostgreSQL) Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Amazon RDS (PostgreSQL) Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Amazon RDS (PostgreSQL)

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Amazon RDS (PostgreSQL)コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] – データベースのホスト URL を入力します (例: `http://instance URL.region.rds.amazonaws.com`)。

c. [ポート] – データベースポートを入力します (例: 5432)。

d. [インスタンス] – データベースインスタンスを入力します (例: `postgres`)。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Amazon RDS (PostgreSQL) 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

  A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

     I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraAmazon RDS (PostgreSQL)-' がシークレット名に自動的に追加されます。

     II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

  B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満でなければなりません。SQL クエリは 32 KB 未満で、セミコロン (;) を含めないでください。 Amazon Kendra クエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。

- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Amazon RDS (PostgreSQL)

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを postgresql として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Amazon RDS (PostgreSQL)シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。Amazon RDS (PostgreSQL) Amazon Kendra詳細については、「IAM roles for Amazon RDS (PostgreSQL) data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Amazon RDS (PostgreSQL) データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Amazon RDS (PostgreSQL) テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。

- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Amazon S3

Amazon S3 は、データをオブジェクトとしてバケット内に保存するオブジェクトストレージサービスです。 Amazon Kendra Amazon S3 を使用してドキュメントのバケットリポジトリにインデックスを付けることができます。

> **Warning**
>
> Amazon Kendra S3 Amazon Kendra バケットを操作する権限をプリンシパルに付与するバケットポリシーは使用しません。 IAM 代わりにロールを使用します。誤って任意のプリンシパルにアクセス許可を付与することによるデータセキュリティ上の問題を避けるため、Amazon Kendra バケットポリシーに信頼できるメンバーとして含まれていないことを確認してください。ただしバケットポリシーを追加すれば、異なるアカウント間で Amazon S3 バケットを使用できます。詳細については、「Amazon S3 アカウントで使用するポリシー」(S3 IAM ロールタブの [データソースのIAM ロール]) を参照してください。S3 IAM データソースのロールについては、「ロール」を参照してくださいIAM 。

> **Note**
>
> Amazon Kendra Amazon S3 アップグレードされたコネクタをサポートするようになりました。
> コンソールは自動的にアップグレードされました。コンソールに新しいコネクタを作成すると、アップグレードされたアーキテクチャが使用されます。API を使用する場合は、TemplateConfigurationオブジェクトではなくオブジェクトを使用してコネクタを設定する必要があります。`S3DataSourceConfiguration`
> 古いコンソールと API アーキテクチャを使用して設定されたコネクタは、引き続き設定どおりに機能します。ただし、編集や更新はできません。コネクタ構成を編集または更新する場合は、新しいコネクタを作成する必要があります。

コネクタワークフローをアップグレードされたバージョンに移行することをお勧めします。古いアーキテクチャを使用して構成されたコネクタSupport は、2024 年 6 月までに終了する予定です。

Amazon Kendra コンソールまたは TemplateConfigurationAPI Amazon S3 を使用してデータソースに接続できます。

> **Note**
>
> データソースの同期ステータスレポートを生成するには、「 Amazon S3 データソースのトラブルシューティング」を参照してください。

Amazon Kendra S3 データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- Amazon S3 データソースの作成
- Amazon S3 ドキュメントメタデータ
- Amazon S3 データソースのアクセス制御
- Amazon VPCAmazon S3 データソースとの併用

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して S3 データソースのインデックスを作成する前に、S3 AWS とアカウントでこれらの変更を行ってください。

S3 で、次のものが揃っていることを確認してください。

- Amazon S3 バケット名の名前をコピーしました。

> **Note**
>
> Amazon Kendra バケットはインデックスと同じリージョンにある必要があり、インデックスにはドキュメントを含むバケットにアクセスする権限が必要です。

- 各ドキュメントが S3 および同じインデックスに使用する予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていない必要があります。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

AWS アカウントには、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めておきます。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

IAM 既存のロールがない場合は、S3 IAM データソースをに接続するときにコンソールを使用して新しいロールを作成できます。 Amazon Kendra API を使用している場合は、 IAM 既存のロールの ARN とインデックス ID を指定する必要があります。

接続手順

S3 Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように S3 データソースの必要な詳細情報を入力する必要があります。S3 をまだ設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Amazon Kendra に接続するには Amazon S3

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [S3 コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 — インデックスのドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次のオプション情報を入力します。

   a. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

   > **Note**
   >
   > IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

   b. Virtual Private Cloud (VPC) — パブリックインターネットからアクセスできない場合は、 Amazon VPC Amazon S3 バケットに使用することを選択できます。その場合は、Amazon VPC サブネットとセキュリティグループを追加する必要があります。

> **Important**
>
> 次のものが揃っていることを確認してください。
>
> - 「ゲートウェイエンドポイント」 Amazon S3 Amazon VPC の手順に従ってエンドポイントをに追加しました。 Amazon S3
> - Amazon Kendra サポートされているアベイラビリティーゾーンのプライベートサブネットを選択しました。詳細については、「Amazon Kendra を使用するように設定する Amazon VPC」を参照してください。
> - Amazon Kendra が Amazon S3 エンドポイントにアクセスできるようセキュリティグループが設定されていること。詳細については、「Amazon KendraAmazon VPC使用するための設定」を参照してください。

c. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] の [データソースの場所] - Amazon S3 データが保存されているバケットへのパス。[S3 をブラウズ] を選択し、バケットを選択します。

b. (オプション) [メタデータファイルのプレフィックスフォルダの場所] - メタデータが保存されているフォルダへのパス。[S3 をブラウズ] を選択してメタデータフォルダを探します。

c. (オプション) [アクセスコントロールリスト設定ファイルの場所] - S3 データソースに保存されているファイルのアクセス設定を指定する、JSON 構造を含むファイルの場所へのパス。[S3 をブラウズ] を選択して ACL ファイルを探します。

d. (オプション) [復号キーを選択] - 復号キーを使用する場合は選択します。 AWS KMS 既存のキーを使用することもできます。

e. (オプション) [追加の設定] の [パターン] で、ドキュメントをインデックスに含めるか、除外するかパターンを追加します。すべてのパスは、データソースの場所の S3 バケットに相対的です。最大 100 のパターンを追加できます。

f. [同期モード] では、[完全同期モード] か、[新規、変更、削除済みコンテンツの同期] のいずれかを選択して、データソースのコンテンツが変更されたときにインデックスを更新する方法を指定します。Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

g. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

h. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次のオプション情報を入力します。

a. S3 フィールドマッピング — Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングするフィールドを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra 接続するには: Amazon S3

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- BucketName— ドキュメントを含むバケットの名前。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、S3 コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「S3 データソースのIAM ロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター — 特定のファイル名、ファイルタイプ、ファイルパスを含めるか除外するかを指定します。グロブパターン (ワイルドカードパターンを展開して、特定のパターンに一致するパス名のリストを作成できるパターン) を使用します。例については、AWS CLI コマンドリファレンスの「除外フィルターと包含フィルターの使用」を参照してください。
- ドキュメントメタデータの設定 - ドキュメントのアクセスコントロール情報、ソース URI、ドキュメントの作成者、カスタム属性などの情報が含まれているドキュメントメタデータファイルを追加します。各メタデータファイルには、1 つのドキュメントに関するメタデータが含まれています。
- フィールドマッピング - 選択すると、S3 データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

  > **Note**
  >
  > ドキュメントを検索するには、 Amazon Kendra ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Amazon S3 テンプレートスキーマ」を参照してください。

詳細はこちら

S3 Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- VPC Amazon Kendra をサポートする S3 コネクタを使用して回答を正確に検索する

Amazon S3 データソースの作成

以下の例は、 Amazon S3 データソースの作成を示しています。この例では、インデックスと、 IAM そのインデックスからデータを読み取る権限を持つロールが既に作成されていることを前提としてい

ます。 IAM ロールの詳細については、「IAM アクセスロール」を参照してください。インデックスの作成の詳細については、「インデックスの作成」を参照してください。

CLI

```
aws kendra create-data-source \
 --index-id index ID \
 --name example-data-source \
 --type S3 \
 --configuration '{"S3Configuration":{"BucketName":"bucket name"}}'
 --role-arn 'arn:aws:iam::account id:role:/role name
```

Python

次の Python コードスニペットは、 Amazon S3 データソースを作成します。完全な例については、開始方法 (AWS SDK for Python (Boto3)) を参照してください。

```
print("Create an Amazon S3 data source.")

    # Provide a name for the data source
    name = "getting-started-data-source"
    # Provide an optional description for the data source
    description = "Getting started data source."
    # Provide the IAM role ARN required for data sources
    role_arn = "arn:aws:iam::${accountID}:role/${roleName}"
    # Provide the data soource connection information
    s3_bucket_name = "S3-bucket-name"
    type = "S3"
    # Configure the data source
    configuration = {"S3DataSourceConfiguration":
        {
            "BucketName": s3_bucket_name
        }
    }

    data_source_response = kendra.create_data_source(
        Configuration = configuration,
        Name = name,
        Description = description,
        RoleArn = role_arn,
        Type = type,
        IndexId = index_id
```

```
    )
```

データソースの作成には時間がかかることがあります。DescribeDataSourceAPI を使用して進行状況を監視できます。データソースのステータスが ACTIVE の場合、データソースを使用する準備ができています。

次の例は、データソースのステータスの取得を示しています。

CLI

```
aws kendra describe-data-source \
 --index-id index ID \
 --id data source ID
```

Python

次の Python コードのスニペットでは、 S3 データソースに関する情報を取得します。完全な例については、開始方法 (AWS SDK for Python (Boto3)) を参照してください。

```
print("Wait for Amazon Kendra to create the data source.")

    while True:
        data_source_description = kendra.describe_data_source(
            Id = "data-source-id",
            IndexId = "index-id"
        )
        status = data_source_description["Status"]
        print(" Creating data source. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break
```

このデータソースにはスケジュールがないため、自動的に実行されません。StartDataSourceSyncJobデータソースにインデックスを付けるには、を呼び出してインデックスをデータソースと同期させます。

次の例は、データソースの同期を示しています。

CLI

```
aws kendra start-data-source-sync-job \
 --index-id index ID \
 --id data source ID
```

Python

次の Python コードのスニペットで Amazon S3 データソースを同期します。完全な例については、開始方法 (AWS SDK for Python (Boto3)) を参照してください。

```
print("Synchronize the data source.")

    sync_response = kendra.start_data_source_sync_job(
        Id = "data-source-id",
        IndexId = "index-id"
    )
```

Amazon S3 ドキュメントメタデータ

メタデータファイルを使用して、メタデータ、つまりドキュメントに関する追加情報を Amazon S3 バケット内のドキュメントに追加できます。各メタデータファイルは、インデックス作成されたドキュメントに関連付けられます。

メタデータファイルは、インデックス作成されたファイルと同じバケットに保存する必要があります。 Amazon S3 データソースを作成するときに、S3PrefixDocumentsMetadataConfigurationコンソールまたはパラメーターのフィールドを使用して、メタデータファイルのバケット内の場所を指定できます。 Amazon S3 プレフィックスを指定しない場合、メタデータファイルはインデックス作成されたドキュメントと同じ場所に保存する必要があります。

Amazon S3 メタデータファイルにプレフィックスを指定すると、それらはインデックス付きドキュメントと同じディレクトリ構造になります。 Amazon Kendra 指定されたディレクトリでのみメタデータを検索します。メタデータが読み込まれない場合は、ディレクトリの場所がメタデータの場所と一致していることをチェックします。

次の例は、インデックス作成されたドキュメントの場所がメタデータファイルの場所にどのようにマッピングされるかを示しています。 Amazon S3 Amazon S3 ドキュメントのキーはメタデータのプレフィックスに追加され、.metadata.jsonその後にサフィックスが付けられてメタデータファイルのパスになることに注意してください。 Amazon S3 Amazon S3 Amazon S3

.metadata.jsonメタデータのプレフィックスとサフィックスを組み合わせたキーは、合計で 1024 文字以下でなければなりません。 Amazon S3 キーをプレフィックスとサフィックスを組み合わせる場合は、文字数が増えることを考慮して、キーを 1000 文字未満にすることをお勧めします。

```
Bucket name:
     s3://bucketName
Document path:
     documents
Metadata path:
     none
File mapping
     s3://bucketName/documents/file.txt ->
        s3://bucketName/documents/file.txt.metadata.json
```

```
Bucket name:
     s3://bucketName
Document path:
     documents/legal
Metadata path:
     metadata
File mapping
     s3://bucketName/documents/legal/file.txt ->
        s3://bucketName/metadata/documents/legal/file.txt.metadata.json
```

ドキュメントのメタデータは JSON ファイルで定義されます。ファイルは、BOM マーカーの無い UTF-8 テキストファイルである必要があります。JSON ファイルのファイル名は <document>.<extension>.metadata.json である必要があります。この例では、「document」はメタデータが適用されるドキュメントの名前、「extension」はドキュメントのファイル拡張子です。ドキュメント ID は <document>.<extension>.metadata.json 内で一意である必要があります。

JSON ファイルの内容はこのテンプレートに従います。すべての属性やフィールドはオプションなので、すべての属性を含める必要はありません。含める属性ごとに値を指定する必要があります。値を空にすることはできません。を指定しない場合_source_uri、 Amazon Kendra 検索結果で返されるリンクは、 Amazon S3 ドキュメントを含むバケットを指します。DocumentIds3_document_idはフィールドにマップされ、S3 内のドキュメントへの絶対パスです。

```
{
    "DocumentId": "S3 document ID, the S3 path to doc",
```

```
    "Attributes": {
        "_category": "document category",
        "_created_at": "ISO 8601 encoded string",
        "_last_updated_at": "ISO 8601 encoded string",
        "_source_uri": "document URI",
        "_version": "file version",
        "_view_count": number of times document has been viewed,
        "custom attribute key": "custom attribute value",
        additional custom attributes
    },
    "AccessControlList": [
         {
             "Name": "user name",
             "Type": "GROUP | USER",
             "Access": "ALLOW | DENY"
         }
    ],
    "Title": "document title",
    "ContentType": "For example HTML | PDF. For supported content types, see Types of
 documents."
}
```

_created_at および _last_updated_at メタデータフィールドは ISO 8601 でエンコードされた日付です。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。

クエリをフィルタリングしたり、クエリ応答をグループ化したりするために使用するドキュメントに関する Attributes フィールドに追加情報を追加できます。詳細については、「カスタムドキュメントフィールドの作成」を参照してください。

AccessControlList フィールドを使用して、クエリからのレスポンスをフィルタリングできます。これにより、特定のユーザーおよびグループのみがドキュメントにアクセスできます。詳細については、「ユーザーコンテキストでのフィルタリング」を参照してください。

Amazon S3 データソースのアクセス制御

設定ファイルを使用して、 Amazon S3 データソース内のドキュメントへのアクセスを制御できます。ファイルはコンソールで指定するか、CreateDataSourceまたは UpdateDataSourceAPI AccessControlListConfiguration を呼び出すときのパラメータとして指定します。

設定ファイルには、S3 プレフィックスを識別し、プレフィクスのアクセス設定を一覧表示する JSON 構造が含まれています。プレフィックスには、パス、または個別のファイルを使用できま

す。プレフィックスがパスの場合、アクセス設定はそのパス内のすべてのファイルに適用されます。JSON 設定ファイルには S3 プレフィックスの最大数とデフォルトの最大ファイルサイズがあります。詳細については、「のクォータ Amazon Kendra」を参照してください。

アクセス設定でユーザーとグループの両方を指定できます。インデックスをクエリするときは、ユーザー情報とグループ情報を指定します。詳細については、「ユーザー属性でフィルタリングする」を参照してください。

設定ファイルの JSON 構造は次のような形式になります。

```
[
    {
        "keyPrefix": "s3://BUCKETNAME/prefix1/",
        "aclEntries": [
            {
                "Name": "user1",
                "Type": "USER",
                "Access": "ALLOW"
            },
            {
                "Name": "group1",
                "Type": "GROUP",
                "Access": "DENY"
            }
        ]
    },
    {
        "keyPrefix": "s3://prefix2",
        "aclEntries": [
            {
                "Name": "user2",
                "Type": "USER",
                "Access": "ALLOW"
            },
            {
                "Name": "user1",
                "Type": "USER",
                "Access": "DENY"
            },
            {
                "Name": "group1",
                "Type": "GROUP",
                "Access": "DENY"
```

```
            }
        ]
    }
]
```

Amazon VPCAmazon S3 データソースとの併用

このトピックでは、Amazon VPC 経由で Amazon S3 コネクタを使用して Amazon S3 step-by-step バケットに接続する方法を示す例を紹介します。この例では、既存の S3 バケットから始めることを前提としています。この例をテストするには、S3 バケットにいくつかのドキュメントだけをアップロードすることをお勧めします。

Amazon Kendra Amazon S3 バケットにはを介して接続できます Amazon VPC。そのためには、Amazon VPC Amazon VPC Amazon S3 データソースコネクタの作成時にサブネットとセキュリティグループを指定する必要があります。

> **Important**
>
> Amazon Kendra Amazon S3 Amazon S3 コネクタがバケットにアクセスできるように、仮想プライベートクラウド (VPC) Amazon S3 にエンドポイントを割り当てていることを確認してください。

Amazon Kendra Amazon S3 バケットからドキュメントを同期するには Amazon VPC、次の手順を完了する必要があります。

- Amazon S3 のエンドポイントを設定します Amazon VPC。 Amazon S3 エンドポイントの設定方法の詳細については、AWS PrivateLink ガイドの「Gateway エンドポイント」を参照してください。 Amazon S3
- (オプション) Amazon S3 バケットポリシーをチェックして、割り当てた仮想プライベートクラウド (VPC) Amazon S3 からバケットにアクセスできることを確認しました。 Amazon Kendra詳細については、Amazon S3 ユーザーガイドの「バケットポリシーによる VPC エンドポイントからのアクセスの制御」を参照してください。

ステップ

- ステップ 1: を設定する Amazon VPC
- (オプション) ステップ 2: Amazon S3 バケットポリシーを設定する

- ステップ 3: Amazon S3 テストデータソースコネクタを作成する

ステップ 1: を設定する Amazon VPC

Amazon S3 Amazon Kendra 後で使用するためのゲートウェイエンドポイントとセキュリティグループを含むプライベートサブネットを含む VPC ネットワークを作成します。

プライベートサブネット、S3 エンドポイント、セキュリティグループで VPC を設定するには

1. AWS Management Console にサインインし、 Amazon VPC https://console.aws.amazon.com/vpc/でコンソールを開きます。
2. プライベートサブネットと S3 Amazon Kendra エンドポイントを使用して VPC を作成します。

   ナビゲーションペインから [Your VPC] を選択し、[VPC の作成] を選択します。

   a. [Resources to create] (作成するリソース) で、[VPC and more] (VPC など) を選択します。
   b. [名前タグ] で [自動生成] を有効にして、と入力します。**kendra-s3-example**
   c. IPv4/IPv6 CIDR ブロックの場合は、デフォルト値のままにします。
   d. [アベイラビリティーゾーン (AZ) の数] では、番号 1 を選択します。
   e. [Customize AZ] を選択し、1 番目のアベイラビリティーゾーンリストからアベイラビリティーゾーンを選択します。

      Amazon Kendra 特定のアベイラビリティーゾーンのセットのみをサポートします。
   f. [パブリックサブネットの数] では、番号 0 を選択します。
   g. [プライベートサブネットの数] では、番号 1 を選択します。
   h. [NAT ゲートウェイ] には、[なし] を選択します。
   i. VPC エンドポイントの場合は、ゲートウェイを選択しますAmazon S3 。 。
   j. 残りの値はデフォルト設定のままにします。
   k. [VPC の作成] を選択します。

      VPC 作成ワークフローが完了するまでお待ちください。次に、[VPC を表示] を選択して、先ほど作成した VPC を確認します。

   これで、パブリックインターネットにアクセスできないプライベートサブネットを持つ VPC ネットワークが作成されました。
3. Amazon S3 エンドポイントの VPC エンドポイント ID をコピーします。

a. ナビゲーションペインで、[Endpoints] (エンドポイント) を選択します。
b. エンドポイントリストで、VPC と一緒に作成した Amazon S3 `kendra-s3-example-vpce-s3` エンドポイントを探します。
c. VPC エンドポイント ID をメモしておきます。

これで、サブネット経由で Amazon S3 バケットにアクセスするための Amazon S3 ゲートウェイエンドポイントが作成されました。

4. Amazon Kendra 使用するセキュリティグループを作成します。

a. ナビゲーションペインから [セキュリティグループ] を選択し、[セキュリティグループの作成] を選択します。
b. [Security group name] (セキュリティグループ名) に **`s3-data-source-security-group`** と入力します。
c. Amazon VPCリストから VPC を選択します。
d. インバウンドルールとアウトバウンドルールはデフォルトのままにします。
e. [Create Security Group] を選択します。

これで VPC セキュリティグループが作成されました。

コネクタの設定プロセス中に、作成したサブネットとセキュリティグループを Amazon Kendra Amazon S3 データソースコネクタに割り当てます。

(オプション) ステップ 2: Amazon S3 バケットポリシーを設定する

このオプションステップでは、Amazon S3 バケットに、割り当てた VPC からのみアクセスできるように Amazon S3 バケットポリシーを設定する方法を学習します。 Amazon Kendra

Amazon Kendra IAM ロールを使用して Amazon S3 バケットにアクセスするため、Amazon S3 バケットポリシーを設定する必要はありません。ただし、パブリックインターネットからのアクセスを制限する既存のポリシーがある Amazon S3 Amazon S3 バケットを使用してコネクタを設定する場合は、バケットポリシーを作成すると便利な場合があります。

Amazon S3 バケットポリシーを設定するには

1. https://console.aws.amazon.com/s3/ で Amazon S3 コンソールを開きます。
2. ナビゲーションペインから [Buckets] を選択します。

3. 同期したい Amazon S3 バケットの名前を選択します Amazon Kendra。
4. [権限] タブを選択し、[バケットポリシー] までスクロールして [編集] をクリックします。
5. 作成した VPC エンドポイントからのアクセスのみを許可するようにバケットポリシーを追加または変更します。

   以下は、バケットポリシーの例です。*bucket-name*と*vpce-id*を、自分の Amazon S3 バケット名と、先にメモしておいた Amazon S3 エンドポイント ID に置き換えます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::bucket-name/*",
      "Condition": {
        "StringNotEquals": {
          "aws:SourceVpce": "vpce-id"
        }
      }
    }
  ]
}
```

6. [変更を保存] を選択します。

S3 バケットには、作成した特定の VPC からのみアクセスできるようになりました。

ステップ 3: Amazon S3 テストデータソースコネクタを作成する

Amazon VPC 構成をテストするには、 Amazon S3 コネクタを作成します。次に、で説明されている手順に従って作成した VPC で設定します。Amazon S3

Amazon VPC 設定値には、この例で作成した値を選択します。

- Amazon VPC(VPC) — `kendra-s3-example-vpc`
- サブネット — `kendra-s3-example-subnet-private1-[availability zone]`
- セキュリティグループ — `s3-data-source-security-group`

コネクタの作成が完了するまでお待ちください。 Amazon S3 コネクタが作成されたら、[Sync now] を選択して同期を開始します。

Amazon S3 バケット内のドキュメントの数によっては、同期が完了するまでに数分から数時間かかる場合があります。この例をテストするには、S3 バケットにいくつかのドキュメントだけをアップロードすることをお勧めします。設定が正しければ、最終的に Sync ステータスが Completed と表示されるはずです。

エラーが発生した場合は、「Amazon VPC 接続のトラブルシューティング」を参照してください。

Amazon Kendra ウェブクローラー

Amazon Kendra ウェブクローラーを使用して、ウェブページをクロールおよびインデックス作成できます。

クロールできるのは、公開ウェブサイト、または、安全な通信プロトコルである Hypertext Transfer Protocol Secure (HTTPS) を使用する社内ウェブサイトのみです。ウェブサイトをクロールするときにエラーが発生した場合は、ウェブサイトのクロールがブロックされている可能性があります。内部ウェブサイトをクロールするには、ウェブプロキシを設定できます。ウェブプロキシは公開されている必要があります。認証を使用してウェブサイトにアクセスし、クロールすることもできます。

インデックス作成するウェブサイトを選択するときは、Amazon 利用規定ポリシーおよびその他の Amazon 規約のすべてに準拠している必要があります。独自のウェブページ、またはインデックス作成の権限を持つウェブページのインデックス作成には、 Amazon Kendra Web Crawler のみを使用する必要があることに注意してください。 Amazon Kendra ウェブクローラーによるウェブサイトのインデックス作成を停止する方法については、「」を参照してくださいAmazon Kendra Web Crawler 用の *robots.txt* ファイルの設定。

> **Note**
>
> Amazon Kendra ウェブクローラーを悪用して、所有していないウェブサイトやウェブページを積極的にクロールすることは、許容される使用とは見なされません。

Amazon Kendra には 2 つのバージョンのweb crawlerコネクタがあります。各バージョンでサポートされる機能は次のとおりです。

Amazon Kendra ウェブクローラーコネクタ v1.0/WebCrawlerConfigurationAPI

- ウェブプロキシ

- 包含/除外フィルター

Amazon Kendra ウェブクローラーコネクタ v2.0/TemplateConfigurationAPI

- フィールドマッピング
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期
- ウェブプロキシ
- ウェブサイトの基本認証、NTLM/Kerberos 認証、SAML 認証、フォーム認証
- 仮想プライベートクラウド (VPC)

> **Important**
>
> ウェブクローラー v2.0 コネクタの作成は、 ではサポートされていません AWS CloudFormation。 AWS CloudFormation サポートが必要な場合は、Web Crawler v1.0 コネクタを使用します。

Amazon Kendra ウェブクローラーデータソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

- Amazon Kendra ウェブクローラーコネクタ v1.0
- Amazon Kendra ウェブクローラーコネクタ v2.0
- Amazon Kendra Web Crawler 用の robots.txt ファイルの設定

Amazon Kendra ウェブクローラーコネクタ v1.0

Amazon Kendra ウェブクローラーを使用して、ウェブページをクロールおよびインデックス作成できます。

クロールできるのは、公開ウェブサイトと、安全な通信プロトコルである Hypertext Transfer Protocol Secure (HTTPS) を使用するウェブサイトのみです。ウェブサイトをクロールするときにエラーが発生した場合は、ウェブサイトのクロールがブロックされている可能性があります。内部ウェ

ブサイトをクロールするには、ウェブプロキシを設定できます。ウェブプロキシは公開されている必要があります。

インデックス作成するウェブサイトを選択するときは、Amazon 利用規定ポリシーおよびその他の Amazon 規約のすべてに準拠している必要があります。独自のウェブページ、またはインデックス作成の権限を持つウェブページのインデックス作成には、 Amazon Kendra Web Crawler のみを使用する必要があることに注意してください。 Amazon Kendra ウェブクローラーによるウェブサイトのインデックス作成を停止する方法については、「」を参照してくださいAmazon Kendra Web Crawler 用の *robots.txt* ファイルの設定。

> **Note**
>
> Amazon Kendra ウェブクローラーを悪用して、所有していないウェブサイトやウェブページを積極的にクロールすることは、許容できる使用とは見なされません。

Amazon Kendra ウェブクローラーデータソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

- ウェブプロキシ
- 包含/除外フィルター

前提条件

Amazon Kendra を使用してウェブサイトのインデックスを作成する前に、ウェブサイトと AWS アカウントの詳細を確認してください。

ウェブサイトについて、以下を確認してください。

- インデックス作成するウェブサイトのシードまたはサイトマップ URL をコピーしました。

- 基本認証を必要とするウェブサイトの場合: ユーザー名とパスワードを書き留め、ウェブサイトのホスト名とポート番号をコピーしました。
- オプション:ウェブプロキシを使用して、クロールする内部ウェブサイトに接続する場合に、ウェブサイトのホスト名とポート番号をコピーしました。ウェブプロキシは公開されている必要があります。 Amazon Kendra では、基本認証によってバックアップされたウェブプロキシサーバーへの接続がサポートされています。認証なしで接続することもできます。
- インデックスを作成する各ドキュメントが一意であり、同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

AWS アカウントで、以下があることを確認します。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。
- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- 認証が必要なウェブサイト、または 認証でウェブプロキシを使用する場合は、 が認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、web crawlerデータソースを に接続するときに、コンソールを使用して新しい IAM ロールと Secrets Manager シークレットを作成できます

Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

web crawler データソース Amazon Kendra に接続するには、 がweb crawlerデータ Amazon Kendra にアクセスできるように、データソースの必要な詳細を入力する必要があります。をまだ設定していない場合は、web crawler Amazon Kendra 「」を参照してください前提条件。

Console

Amazon Kendra に接続するには web crawler

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加ページで、ウェブクローラーコネクタ を選択し、コネクタ を追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語 - インデックスのドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] では、ユースケースに応じて [ソース URL] と [ソースサイトマップ] を選択し、それぞれの値を入力します。

   ソース URL は 10 個まで、サイトマップは 3 個まで追加できます。

   > **Note**
   >
   > サイトマップをクロールする場合は、ベース URL またはルート URL がサイトマップページに記載されている URL と同じであることを確認してください。例えば、サイトマップ URL が https://example.com/sitemap-page.html の場合、このサイトマップページに記載されている URL にもベース URL "https://example.com/" を使用する必要があります。

b. (オプション) [ウェブプロキシ] - 次の情報を入力します。

   i. [ホスト名] - ウェブプロキシを必要とするホスト名。

   ii. [ポート番号] - ホスト URL トランスポートプロトコルが使用するポート。ポート番号は 0 ~ 65535 の数字である必要があります。

   iii. ウェブプロキシ認証情報の場合 - ウェブプロキシ接続で認証が必要な場合は、既存のシークレットを選択するか、認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

   iv. [AWS Secrets Manager Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

      A. [シークレット名] - シークレットの名前。シークレット名に、プレフィックス「AmazonKendra-WebCrawler-」が自動的に追加されます。

      B. [ユーザー名] と [パスワード] - ウェブサイトの基本認証情報を入力します。

      C. [保存] を選択します。

c. (オプション) [認証済みのホスト] - 選択すると、認証付きのホストをさらに追加できます。

d. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [クロール範囲] - クロールするウェブページの種類を選択します。

   b. クロール深度 — クロール Amazon Kendra する必要があるシード URL からレベルの数を選択します。

   c. [クロールの詳細設定] および [追加設定] - 次の情報を入力します。

      i. [最大ファイルサイズ] - クロールするウェブページまたは添付ファイルの最大サイズ。最小 0.000001 MB (1 バイト)。最大 50 MB。

      ii. 1 ページあたりの最大リンク数 - 1 ページあたりにクロールされるリンクの最大数。リンクは表示順にクロールされます。1 ページあたり最小 1 リンク。1 ページあたり最大 1000 リンク。

      iii. 最大スロットリング - ホスト名ごとにクロールされる URL の、1 分あたりの最大数。ホスト名ごとに 1 分あたり最小 1 URL。ホスト名ごとに 1 分あたり最大 300 URL。

      iv. [正規表現パターン] - 特定の URL を含めるまたは除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

   d. 同期実行スケジュール で、頻度 - Amazon Kendra データソースと同期する頻度を選択します。

   e. [次へ] を選択します。

8. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra に接続するには web crawler

WebCrawlerConfiguration API を使用して以下を指定する必要があります。

- URL - SeedUrlConfiguration と SiteMapsConfiguration を使用して、ウェブサイトのシード URL または開始ポイント URL、または、クロールするウェブサイトのサイトマップ URL を指定します。

  > **Note**
  >
  > サイトマップをクロールする場合は、ベース URL またはルート URL がサイトマップページに記載されている URL と同じであることを確認してください。例えば、サイトマップ URL が https://example.com/sitemap-page.html の場合、このサイトマップページに記載されている URL にもベース URL "https://example.com/" を使用する必要があります。

- シークレットの Amazon リソースネーム (ARN) - ウェブサイトが基本認証を使用する場合は、ホスト名、ポート番号、および、ユーザー名とパスワードの基本認証情報を保存するシークレットを指定します。AuthenticationConfiguration API を使用してシークレット ARN を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

  ```
  {
      "username": "user name",
      "password": "password"
  }
  ```

  AWS Secrets Manager シークレットを使用してウェブプロキシ認証情報を指定することもできます。ProxyConfiguration API を使用して、ウェブサイトのホスト名とポート番号、およびウェブプロキシ認証情報を保存するシークレットを指定します。

- role IAM — を呼び出しCreateDataSourceて、 Secrets Manager シークレットにアクセスするためのアクセス許可を IAM ロールに付与し、ウェブクローラーコネクタと に必要なパブリック APIs を呼び出すRoleArnタイミングを指定します Amazon Kendra。詳細については、「IAM roles for web crawler data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- クロールモード –ウェブサイトのホスト名のみをクロールするか、サブドメインを含むホスト名をクロールするか、ウェブページのリンク先となる他のドメインもクロールするかを選択します。

- 深さ、またはシードレベルからクロールするレベルの数。例えば、シード URL ページは深度 1 で、このページ上でクロールされるハイパーリンクはすべて深度 2 です。
- クロールする単一ウェブページの URL の最大数。
- クロールするウェブページの最大サイズ (MB 単位)。
- 1 分あたりウェブサイトホストごとにクロールされる URL の最大数。
- 内部ウェブサイトに接続してクロールするウェブプロキシのホストとポート番号。例えば、https://a.example.com/page1.html のホスト名は 「a.example.com」で、ポート番号は HTTPS の標準ポートである 443 です。ウェブサイトホストへの接続にウェブプロキシ認証情報が必要な場合は、認証情報を保存する AWS Secrets Manager を作成できます。
- ユーザー認証を必要とするウェブサイトにアクセスしてクロールするための認証情報。
- カスタムドキュメントエンリッチメントツールを使用して、HTML メタタグをフィールドとして抽出できます。詳細については、取り込みプロセス中のドキュメントのメタデータのカスタマイズを参照してください。HTML メタタグの抽出例については、「CDE サンプル」を参照してください。
- 包含フィルターと除外フィルター - 特定の URL を含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

詳細はこちら

web crawler データソース Amazon Kendra との統合の詳細については、以下を参照してください。

- Amazon Kendraの Web Crawler を使用してナレッジ検出を再検討する

Amazon Kendra ウェブクローラーコネクタ v2.0

Amazon Kendra ウェブクローラーを使用して、ウェブページをクロールおよびインデックス作成できます。

クロールできるのは、公開ウェブサイト、または、安全な通信プロトコルである Hypertext Transfer Protocol Secure (HTTPS) を使用する社内ウェブサイトのみです。ウェブサイトをクロールするときにエラーが発生した場合は、ウェブサイトのクロールがブロックされている可能性があります。内部ウェブサイトをクロールするには、ウェブプロキシを設定できます。ウェブプロキシは公開されている必要があります。認証を使用してウェブサイトにアクセスし、クロールすることもできます。

Amazon Kendra Web Crawler v2.0 は Selenium ウェブクローラーパッケージと Chromium ドライバーを使用します。 は、継続的インテグレーション (CI) を使用して Selenium と Chromium ドライバーのバージョン Amazon Kendra を自動的に更新します。

インデックス作成するウェブサイトを選択するときは、Amazon 利用規定ポリシーおよびその他の Amazon 規約のすべてに準拠している必要があります。独自のウェブページ、またはインデックス作成の権限を持つウェブページのインデックス作成には、 Amazon Kendra Web Crawler のみを使用する必要があることに注意してください。 Amazon Kendra ウェブクローラーによるウェブサイトのインデックス作成を停止する方法については、「」を参照してくださいAmazon Kendra Web Crawler 用の `robots.txt` ファイルの設定。 Amazon Kendra 所有していないウェブサイトやウェブページを積極的にクロールするために Web Crawler を悪用しても、許容できる用途とは見なされません。

Amazon Kendra ウェブクローラーデータソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

> **Note**
>
> ウェブクローラーコネクタ v2.0 は、 AWS KMS 暗号化された Amazon S3 バケットからのウェブサイトリストのクロールをサポートしていません。 Amazon S3 マネージドキーによるサーバー側の暗号化のみをサポートします。

> **Important**
>
> ウェブクローラー v2.0 コネクタの作成は、 ではサポートされていません AWS CloudFormation。 AWS CloudFormation サポートが必要な場合は、Web Crawler v1.0 コネクタを使用します。

トピック

- サポートされている機能
- 前提条件

- 接続手順

サポートされている機能

- フィールドマッピング
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期
- ウェブプロキシ
- ウェブサイトの基本認証、NTLM/Kerberos 認証、SAML 認証、フォーム認証
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用してウェブサイトのインデックスを作成する前に、ウェブサイトと AWS アカウントの詳細を確認してください。

ウェブサイトについて、以下を確認してください。

- インデックス作成するウェブサイトのシードまたはサイトマップ URL をコピーしました。URL をテキストファイルに保存して、それを Amazon S3 バケットにアップロードできます。テキストファイル内の各 URL は、別々の行にフォーマットする必要があります。サイトマップを Amazon S3 バケットに保存する場合は、サイトマップ XML をコピーし、XML ファイルに保存したことを確認してください。複数のサイトマップ XML ファイルを 1 つの ZIP ファイルにまとめることもできます。

  > **Note**
  >
  > (オンプレミス/サーバー) Amazon Kendra に含まれるエンドポイント情報が、データソース設定の詳細で指定されたエンドポイント情報 AWS Secrets Manager と同じかどうかを確認します。混乱する代理問題は、ユーザーがアクションを実行するアクセス許可がないにもかかわらず、 Amazon Kendra をプロキシとして使用して設定された秘密にアクセスし、アクションを実行するセキュリティの問題です。後でエンドポイント情報を変更する場合は、新しいシークレットを作成してこの情報を同期する必要があります。

- 基本認証、NTLM 認証、または Kerberos 認証を必要とするウェブサイトの場合:
  - ユーザー名とパスワードを含むウェブサイトの認証情報を書き留めました。

> **Note**
>
> Amazon Kendra Web Crawler v2.0 は、パスワードハッシュを含む NTLM 認証プロトコルと、パスワード暗号化を含む Kerberos 認証プロトコルをサポートしています。

- SAML 認証またはログインフォーム認証を必要とするウェブサイトの場合:
  - ユーザー名とパスワードを含むウェブサイトの認証情報を書き留めました。
  - ユーザー名フィールド (SAML を使用する場合は、加えてユーザー名ボタン)、パスワードフィールド、ボタンの XPath (XML Path Language) をコピーし、ログインページ URL をコピーしました。要素の XPath は、ウェブブラウザのデベロッパーツールを使用して確認できます。XPath は通常、次の形式に従います。`//tagname[@Attribute='Value']`。

> **Note**
>
> Amazon Kendra Web Crawler v2.0 は、ヘッドレス Chrome ブラウザと フォームからの情報を使用して、OAuth 2.0 で保護された URL によるアクセスを認証および承認します。

- オプション: ウェブプロキシを使用して、クロールする内部ウェブサイトに接続する場合に、ウェブプロキシサーバーのホスト名とポート番号をコピーしました。ウェブプロキシは公開されている必要があります。 は、基本認証によってバックアップされたウェブプロキシサーバーへの接続 Amazon Kendra をサポートしているため、認証なしで接続できます。
- オプション: VPC を使用して、クロールする内部ウェブサイトに接続する場合は、仮想プライベートクラウド (VPC) のサブネット ID をコピーしました。詳細については、「 の設定 Amazon VPC」を参照してください。
- インデックスを作成する各ドキュメントが一意であり、同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

AWS アカウントには、以下があることを確認してください。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。

- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- 認証が必要なウェブサイト、または 認証でウェブプロキシを使用する場合は、 が認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、web crawlerデータソースを に接続するときに、コンソールを使用して新しい IAM ロールと Secrets Manager シークレットを作成できます Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

web crawler データソース Amazon Kendra に接続するには、 がweb crawlerデータ Amazon Kendra にアクセスできるように、データソースの必要な詳細を入力する必要があります。をまだ設定していない場合は、web crawler Amazon Kendra 「」を参照してください前提条件。

Console

Amazon Kendra に接続するには web crawler

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加 ページで、ウェブクローラーコネクタ を選択し、コネクタ を追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 - 言語を選択して、ドキュメントをフィルタリングしてインデックスを作成します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。

   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] - [ソース URL]、[ソースサイトマップ]、[ソース URL ファイル]、[ソースサイトマップファイル] のいずれかを選択します。最大 100 個のシード URLs のリストを含むテキストファイルを使用する場合は、ファイルが保存されている Amazon S3 バケットへのパスを指定します。サイトマップ XML ファイルを使用する場合は、ファイルが保存されている Amazon S3 バケットへのパスを指定します。複数のサイトマップ XML ファイルを 1 つの ZIP ファイルにまとめることもできます。それ以外の場合は、最大 10 個のシードまたは開始ポイント URL と、最大 3 つのサイトマップ URL を手動で入力できます。

   > **Note**
   >
   > サイトマップをクロールする場合は、ベース URL またはルート URL がサイトマップページに記載されている URL と同じであることを確認してください。例えば、サイトマップ URL が https://example.com/sitemap-page.html の場

合、このサイトマップページに記載されている URL にもベース URL "https://example.com/" を使用する必要があります。

ウェブサイトが、そのウェブサイトにアクセスするために認証を必要とする場合は、基本認証、NTLM/Kerberos 認証、SAML 認証、またはフォーム認証のいずれかを選択できます。それ以外の場合は、認証なしのオプションを選択します。

> **Note**
>
> 後でデータソースを編集して、認証を含むシード URL をサイトマップに変更する場合は、新しいデータソースを作成する必要があります。 Amazon Kendra は認証用の Secrets Manager のシークレット内のシード URL エンドポイント情報を使用してデータソースを設定するため、サイトマップに変更してもデータソースを再設定することはできません。

- AWS Secrets Manager シークレット — ウェブサイトにアクセスするために同じ認証が必要な場合は、既存のシークレットを選択するか、新しいシー Secrets Manager クレットを作成してウェブサイトの認証情報を保存します。新しいシークレットを作成する場合は、 AWS Secrets Manager シークレットウィンドウが開きます。

  [基本] または [NTML/Kerberos] を選択した場合は、シークレットの名前と、ユーザー名、パスワードを入力します。NTLM 認証プロトコルにはパスワードハッシュが含まれ、Kerberos 認証プロトコルにはパスワード暗号化が含まれます。

  [SAML] または [フォーム] を選択した場合は、シークレットの名前と、ユーザー名、パスワードを入力します。ユーザー名フィールドには XPath を使用します (SAML を使用する場合はユーザー名ボタンに XPath を使用します)。パスワードフィールドとボタン、およびログインページの URL には XPath を使用します。要素の XPath (XML パス言語) は、ウェブブラウザのデベロッパーツールを使用して確認できます。XPath は通常、次の形式に従います。`//tagname[@Attribute='Value']`。

b. (オプション) [ウェブプロキシ - 内部ウェブサイトへの接続に使用するプロキシサーバーのホスト名とポート番号を入力します。例えば、https://a.example.com/page1.html のホスト名は 「a.example.com」で、ポート番号は HTTPS の標準ポートである 443 です。

ウェブサイトホストに接続するためにウェブプロキシ認証情報が必要な場合は、認証情報 AWS Secrets Manager を保存する を作成できます。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] - ドメイン、ファイルサイズ、リンクなど、ウェブページのクローリングに制限を設定し、正規表現パターンを使用して URL をフィルタリングします。

i. (オプション) [クロールドメイン範囲] - ウェブサイトのドメインのみをクロールするか、サブドメインのあるドメインをクロールするか、ウェブページのリンク先となる他のドメインもクロールするかを選択します。デフォルトでは、 はクロールするウェブサイトのドメイン Amazon Kendra のみをクロールします。

ii. (オプション) [追加設定] - 以下の設定を設定します。

- [クロールの深さ] - 深さ、またはシードレベルからクロールするレベルの数。例えば、シード URL ページは深度 1 で、このページ上でクロールされるハイパーリンクはすべて深度 2 です。
- [最大ファイルサイズ] - クロールするウェブページまたは添付ファイルの最大サイズ (MB 単位)。
- 1 ページあたりの最大リンク数 - クロールする単一ウェブページの URL の最大数。
- [クローリング速度の最大スロットリング] - ウェブサイトホストごとにクロールされる URL の 1 分あたりの最大数。
- [ファイル] - ウェブページのリンク先のファイルをクロールすることを選択します。

- [URL のクロールおよびインデックス作成 - 正規表現パターンを追加して、特定の URL のクローリングと、その URL ウェブページのハイパーリンクのインデックス作成を含めるか除外します。

b. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

c. [同期実行スケジュール] - [頻度] で、 Amazon Kendra がデータソースと同期する頻度。

d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. インデックスにマッピングするウェブページとファイルの Amazon Kendra 生成されたデフォルトフィールドから選択します。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra に接続するには web crawler

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — JSON スキーマを使用するWEBCRAWLERV2ときに、データソースタイプを TemplateConfiguration として指定します。また、 CreateDataSource API を呼び出すTEMPLATEときにデータソースを として指定します。
- URL - ウェブサイトのシードまたは開始ポイント URL、またはクロールするウェブサイトのサイトマップ URL を指定します。シード URLs のリストを保存する Amazon S3 バケットへのパスを指定できます。シード URL のテキストファイル内の各 URL は、別々の行にフォーマットする必要があります。サイトマップ XML ファイルを保存する Amazon S3 バケットへのパスを

指定することもできます。複数のサイトマップファイルを 1 つの ZIP ファイルにまとめ、その ZIP ファイルを Amazon S3 バケットに保存できます。

> **Note**
>
> サイトマップをクロールする場合は、ベース URL またはルート URL がサイトマップページに記載されている URL と同じであることを確認してください。例えば、サイトマップ URL が https://example.com/sitemap-page.html の場合、このサイトマップページに記載されている URL にもベース URL "https://example.com/" を使用する必要があります。

- 同期モード — がすべてのドキュメントを同期するか、新規、変更、削除されたドキュメントのみを同期するかを指定して、インデックス Amazon Kendra を更新します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- 認証 — ウェブサイトで同じ認証を必要とする場合は、BasicAuth、NTLM_Kerberos、SAML、または Form 認証のいずれかを指定します。ウェブサイトが認証を必要としない場合は、NoAuthentication を指定してください。
- シークレットの Amazon リソースネーム (ARN) - ウェブサイトで基本認証、NTLM、または Kerberos 認証が必要な場合は、ユーザー名とパスワードの認証情報を保存するシークレットを指定します。 AWS Secrets Manager シークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "seedUrlsHash": "Hash representation of all seed URLs",
    "userName": "user name",
    "password": "password"
}
```

ウェブサイトで SAML 認証が必要な場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
```

```
    "seedUrlsHash": "Hash representation of all seed URLs",

    "userName": "user name",
    "password": "password",
    "userNameFieldXpath": "XPath for user name field",
    "userNameButtonXpath": "XPath for user name button",
    "passwordFieldXpath": "XPath for password field",
    "passwordButtonXpath": "XPath for password button",
    "loginPageUrl": "Full URL for website login page"
}
```

ウェブサイトでフォーム認証が必要な場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "seedUrlsHash": "Hash representation of all seed URLs",
    "userName": "user name",
    "password": "password",
    "userNameFieldXpath": "XPath for user name field",
    "passwordFieldXpath": "XPath for password field",
    "passwordButtonXpath": "XPath for password button",
    "loginPageUrl": "Full URL for website login page"
}
```

要素の XPath (XML パス言語) は、ウェブブラウザのデベロッパーツールを使用して確認できます。XPath は通常、次の形式に従います。`//tagname[@Attribute='Value']`。

AWS Secrets Manager シークレットを使用してウェブプロキシ認証情報を指定することもできます。

- role IAM — を呼び出し`CreateDataSource`て、シー Secrets Manager クレットにアクセスするためのアクセス許可を IAM ロールに付与し、ウェブクローラーコネクタと に必要なパブリック APIs を呼び出す`RoleArn`タイミングを指定します Amazon Kendra。詳細については、「IAM roles for web crawler data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。

- ドメイン範囲 - サブドメインのみを含むウェブサイトドメインをクロールするか、ウェブページのリンク先となる他のドメインもクロールするかを選択します。デフォルトでは、 はクロールするウェブサイトのドメイン Amazon Kendra のみをクロールします。
- 深さ、またはシードレベルからクロールするレベルの数。例えば、シード URL ページは深度 1 で、このページ上でクロールされるハイパーリンクはすべて深度 2 です。
- クロールする単一ウェブページの URL の最大数。
- クロールするウェブページまたは添付ファイルの最大サイズ (MB 単位)。
- 1 分あたりウェブサイトホストごとにクロールされる URL の最大数。
- 内部ウェブサイトに接続してクロールするウェブプロキシのホストとポート番号。例えば、https://a.example.com/page1.html のホスト名は 「a.example.com」で、ポート番号は HTTPS の標準ポートである 443 です。ウェブサイトホストへの接続にウェブプロキシ認証情報が必要な場合は、認証情報を保存する AWS Secrets Manager を作成できます。
- 包含フィルターと除外フィルター - 特定の URL のクローリングと、その URL ウェブページのハイパーリンクのインデックス作成を含めるか除外するかを指定します。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング - 選択すると、ウェブページとウェブページファイルのフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

設定が必要なその他の重要な JSON キーのリストについては、「Amazon Kendra ウェブクローラーテンプレートスキーマ」を参照してください。

Amazon Kendra Web Crawler 用の **`robots.txt`** ファイルの設定

Amazon Kendra は、 AWS お客様が選択したドキュメントのインデックス作成と検索に使用するインテリジェントな検索サービスです。ウェブ上のドキュメントのインデックスを作成するには、ウェ

ブクローラーを使用できます。 Amazon Kendra これは、特定のウェブサイトのインデックスを作成する前に、どの URL (複数の URL) やその他の運用パラメータ. Amazon Kendra カスタマーが認証を取得する必要があるかを示します。

Amazon Kendra ウェブクローラーは、 Allowや などの標準の robots.txt ディレクティブを尊重しますDisallow。ウェブサイトの robots.txt ファイルを変更して、 Amazon Kendra Web クローラーがウェブサイトをクロールする方法を制御できます。

Amazon Kendra Web クローラーがウェブサイトにアクセスする方法の設定

Allow および Disallowディレクティブを使用して、 Amazon Kendra Web Crawler がウェブサイトのインデックスを作成する方法を制御できます。また、インデックス作成されるウェブページとクロールしないウェブページを制御することもできます。

Amazon Kendra 許可されていないウェブページを除くすべてのウェブページをウェブクローラーでクロールできるようにするには、次のディレクティブを使用します。

```
User-agent: amazon-kendra    # Amazon Kendra Web Crawler
Disallow: /credential-pages/ # disallow access to specific pages
```

Amazon Kendra ウェブクローラーが特定のウェブページのみをクロールできるようにするには、次のディレクティブを使用します。

```
User-agent: amazon-kendra    # Amazon Kendra Web Crawler
Allow: /pages/ # allow access to specific pages
```

Amazon Kendra ウェブクローラーがすべてのウェブサイトのコンテンツをクロールし、他のロボットのクロールを禁止するには、次のディレクティブを使用します。

```
User-agent: amazon-kendra # Amazon Kendra Web Crawler
Allow: / # allow access to all pages
User-agent: * # any (other) robot
Disallow: / # disallow access to any pages
```

Amazon Kendra ウェブクローラーによるウェブサイトのクロールの停止

Disallow ディレクティブを使用して Amazon Kendra 、ウェブクローラーによるウェブサイトのインデックス作成を停止できます。また、クロールされるウェブページとクロールしないウェブページを制御できます。

Amazon Kendra ウェブクローラーによるウェブサイトのクロールを停止するには、次のディレクティブを使用します。

```
User-agent: amazon-kendra # Amazon Kendra Web Crawler
Disallow: / # disallow access to any pages
```

Amazon Kendra ウェブクローラーは、HTML ページのメタタグのロボットnoindexとnofollowディレクティブもサポートしています。これらのディレクティブは、ウェブクローラーによるウェブページのインデックス作成を停止し、ウェブページ上のリンクの追跡を停止します。メタタグをドキュメントのセクションに配置して、ロボットルールのルールを指定します。

例えば、以下のウェブページにはディレクティブロボット noindex および nofollow が含まれています。

```
<html>
<head>
    <meta name="robots" content="noindex, nofollow"/>
    ...
</head>
<body>...</body>
</html>
```

Amazon Kendra Web Crawler に関する質問や懸念がある場合は、AWS サポートチームにお問い合わせください。

Amazon WorkDocs

Amazon WorkDocs は、コンテンツを作成、編集、保存、共有するための安全なコンテンツコラボレーションサービスです。 Amazon Kendra Amazon WorkDocs を使用してデータソースのインデックスを作成できます。

Amazon Kendra コンソールと WorkDocsConfigurationAPI Amazon Kendra Amazon WorkDocs を使用してデータソースに接続できます。

Amazon WorkDocs オレゴン、ノースバージニア、シドニー、シンガポール、アイルランドの各リージョンで利用できます。

Amazon Kendra WorkDocs データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

Amazon Kendra WorkDocs データソースコネクタは次の機能をサポートしています。

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター

前提条件

Amazon Kendra WorkDocs を使用してデータソースのインデックスを作成する前に、 WorkDocs AWS およびアカウントで次の変更を行ってください。

で WorkDocs、次のものが揃っていることを確認してください。

- Amazon WorkDocs Amazon WorkDocs リポジトリのディレクトリ ID (組織 ID) を書き留めました。
- 各ドキュメントが、 WorkDocs 同じインデックスに使用する予定の他のデータソースと重複していないことを確認した。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

AWS アカウントに次のものがあることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めておきます。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

IAM 既存のロールがない場合は、 IAM WorkDocs データソースをに接続するときにコンソールを使用して新しいロールを作成できます。 Amazon Kendra API を使用している場合は、 IAM 既存のロールの ARN とインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 WorkDocs Amazon Kendra データにアクセスできるようにデータソースの必要な詳細情報を入力する必要があります。 WorkDocs をまだ設定していない場合は Amazon Kendra、 WorkDocs を参照してください前提条件。

Console

Amazon Kendra に接続するには Amazon WorkDocs

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [WorkDocs コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. Amazon WorkDocs サイト固有の組織 ID — Amazon WorkDocs インデックスを作成するサイトの ID を選択します。あらかじめサイトを作成しておく必要があります。

   b. IAM ロール — IAM IAM リポジトリの認証情報とインデックスコンテンツにアクセスするための既存のロールを選択するか、新しいロールを作成します。

   > **Note**
   >
   > IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

   c. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. クロールドキュメントコメント - Amazon WorkDocs クロールするエンティティまたはコンテンツタイプ。

   b. [変更ログを使用] - 選択すると、すべてのファイルを同期する代わりにインデックスを更新できます。

   c. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

   d. In Sync の実行スケジュールの頻度- Amazon Kendra データソースと同期する頻度を選択します。

   e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. デフォルトデータソースフィールド — Amazon Kendra 生成されたデフォルトデータソースフィールドの中から、インデックスにマップしたいものを選択します。

   b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra 接続するには Amazon WorkDocs

WorkDocsConfigurationAPI を使用して以下を指定する必要があります。

- Amazon WorkDocs ディレクトリ ID — Amazon WorkDocs ディレクトリの組織 ID を指定します。組織 ID は、[アクティブディレクトリ]、[ディレクトリ] の順で移動して、AWS Directory Service で確認できます。
- IAM ロール — `CreateDataSource` IAM WorkDocs ディレクトリへのアクセス権限をロールに付与したり、コネクタとに必要なパブリック API `RoleArn` を呼び出したりするタイミングを指定します。 WorkDocs Amazon Kendra詳細については、「データソースの IAM ロール」を参照してください。 WorkDocs

オプションで、次の機能を追加することもできます。

- 変更ログ — インデックス内のドキュメントを更新する必要があるかどうかを判断するために、 Amazon Kendra WorkDocs データソースの変更ログメカニズムを使用すべきかどうか。

  > **Note**
  >
  > Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、 Amazon Kendra WorkDocs 変更ログを処理するよりもデータソース内のドキュメントをスキャンするほうが時間がかからない場合があります。 WorkDocsデータソースとインデックスを初めて同期する場合は、すべてのドキュメントがスキャンされます。

- 包含フィルターと除外フィルター - 特定のドキュメントとドキュメントコメントを含めるか除外するかを指定します。各コメントは、個別のドキュメントとしてインデックスが作成されます。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定

した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用のAmazon Kendra ACL がある場合、文書のアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング — WorkDocs データソースフィールドをインデックスフィールドにマップすることを選択します。 Amazon Kendra 詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

詳細はこちら

Amazon Kendra WorkDocs データソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendra Amazon WorkDocs コネクターを使い始めましょう

[Box] (ボックス)

Box は、ファイルホスティング機能を提供するクラウドストレージサービスです。を使用すると Amazon Kendra 、コメント、タスク、Web リンクなど、Box コンテンツ内のコンテンツのインデックスを作成できます。

Amazon Kendra コンソールと BoxConfigurationAPI を使用して Box Amazon Kendra データソースに接続できます。

Amazon Kendra Box データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

Amazon Kendra Box データソースコネクタは次の機能をサポートしています。

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Box データソースのインデックスを作成する前に、Box AWS とアカウントで以下の変更を行ってください。

Box で以下を確認してください。

- Box Enterprise または Box Enterprise Plus アカウント。
- Box Developer Console で Box カスタムアプリを作成し、[サーバー認証 (JWT を使用)] を使用するように設定済み。
- [アプリのアクセスレベル] を [アプリ + エンタープライズアクセス] に設定し、[as-user ヘッダーを使用して API 呼び出しを行えるようにします]。
- 管理者ユーザーを使用して、以下の[アプリケーションのスコープ] を Box アプリに追加しました。
  - Box に保存されているすべてのファイルとフォルダを書き込みます。
  - ユーザーの管理
  - グループの管理
  - エンタープライズプロパティの管理
- クライアント ID、クライアントシークレット、パブリックキー ID、プライベートキー ID、パスフレーズ、認証情報として使用するエンタープライズ ID を含むパブリックキー/プライベートキーペ

アを生成してダウンロードしました。詳細については、「パブリックキーとプライベートキーのキーペア」を参照してください。

- Box Developer Console の設定または Box アプリから Box エンタープライズ ID をコピーしました。例えば、*801234567* です。
- 各ドキュメントが Box および同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次の内容が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Box の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Box データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Box Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Box データソースの必要な詳細情報を入力する必要があります。Box をまだ設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Box Amazon Kendra に接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Box コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引の対象となるドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. Box エンタープライズ ID - Box エンタープライズ ID を入力します。

b. AWS Secrets Manager secret — Box Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

   i. [シークレット名] - シークレットの名前。シークレット名には「AmazonKendra-Box-' というプレフィックスが自動的に追加されます。

   ii. [クライアント ID]、[クライアントシークレット]、[パブリックキー ID]、[プライベートキー ID]、[パスフレーズ] の場合 - Box アカウントで生成し、Box アカウントからダウンロードしたパブリックキー/プライベートキーの値を入力します。

   iii. [保存] を選択します。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [エンティティまたはコンテンツタイプを選択] - クロールする Box エンティティまたはコンテンツタイプ。各コメントは、個別のドキュメントとしてインデックスが作成されます。

   b. [変更ログ] - 選択すると、すべてのファイルを同期する代わりにインデックスを更新できます。

   c. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

   d. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

   e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. [ファイル]、[フォルダー]、[コメント]、[タスク]、[Web リンク] の場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングするフィールドを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Box Amazon Kendra に接続するには

BoxConfigurationAPI を使用して以下を指定する必要があります。

[Box エンタープライズ ID] - Box エンタープライズ ID を入力します。エンタープライズ ID は Box Developer Console の設定で、または Box でアプリを作成するときに確認できます。

- シークレットアマゾンリソースネーム (ARN) — Box Secrets Manager アカウントの認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "clientID": "client-id",
    "clientSecret": "client-secret",
    "publicKeyID": "public-key-id",
    "privateKey": "private-key",
    "passphrase": "pass-phrase"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Box コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Box data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - データソース設定の一部として `VpcConfiguration` を指定します。「VPC を使用するための Amazon Kendra の設定」を参照してください。
- 変更ログ — Box データソース変更ログメカニズムを使用して、インデックス内のドキュメントを更新する必要があるかどうかを判断するかどうか Amazon Kendra 。

> **Note**
>
> Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、変更ログを処理するよりも Box Amazon Kendra データソース内のドキュメントをスキャンするほうが時間がかからない場合があります。Box データソースをインデックスに初めて同期する場合は、すべてのドキュメントがスキャンされます。

- コメント、タスク、Web リンク — これらの種類のコンテンツをクロールするかどうかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- 包含フィルターと除外フィルター — 特定の Box ファイルおよびフォルダを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用のAmazon Kendra ACL がある場合、文書のアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Box データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

詳細はこちら

Box Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendra Box コネクタを使い始める

Confluence

Confluence は、プロジェクト計画、ソフトウェア開発、製品管理の共有、保存、作業を目的とした共同作業管理ツールです。 Amazon Kendra を使用して、Confluence スペース、ページ (ネストされ

たページを含む)、ブログ、インデックスされたページやブログへのコメントや添付ファイルのインデックスを作成できます。

Amazon Kendra Confluence サーバーと Confluence クラウドの両方をサポートしています。

> **Note**
>
> デフォルトでは、Confluence Amazon Kendra アーカイブとパーソナルスペースはインデックスされません。データソースの作成時に、インデックス作成を行うことができます。Amazon Kendra スペースにインデックスを付けたくない場合は、Confluence でそのスペースを非公開に設定してください。

Amazon Kendra コンソール、API、または TemplateConfigurationAPI のいずれかを使用して Confluence Amazon Kendra データソースに接続できます。ConfluenceConfiguration

Amazon Kendra には 2 つのバージョンの Confluence コネクターがあります。各バージョンでサポートされる機能は次のとおりです。

コンフルエンスコネクター V1.0/ API ConfluenceConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- (Confluence サーバーのみ) 仮想プライベートクラウド (VPC)

コンフルエンスコネクター V2.0/ API TemplateConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 仮想プライベートクラウド (VPC)
- すべてのドキュメントを同期、または新規、変更、削除したドキュメントのみを同期
- 包含/除外パターン

> **Note**
>
> Confluence コネクター ConfluenceConfiguration V1.0/API Support は 2023 年に終了する予定です。Confluence コネクター V2.0/ API に移行するか、使用することをお勧めします。TemplateConfiguration

Amazon Kendra Confluence データソースコネクターのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- Confluence コネクタ V1.0
- Confluence コネクタ V2.0

Confluence コネクタ V1.0

Confluence は、プロジェクト計画、ソフトウェア開発、製品管理の共有、保存、作業を目的とした共同作業管理ツールです。 Amazon Kendra を使用して、Confluence のスペース、ページ (ネストされたページを含む)、ブログ、インデックスに登録されたページやブログへのコメントや添付ファイルのインデックスを作成できます。

> **Note**
>
> Confluence コネクター ConfluenceConfiguration V1.0/API Support は 2023 年に終了する予定です。Confluence コネクター V2.0/ API に移行するか、使用することをお勧めします。TemplateConfiguration

Amazon Kendra Confluence データソースコネクターのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Confluence データソースコネクターは以下の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- (Confluence サーバーのみ) 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Confluence データソースのインデックスを作成する前に、Confluence とアカウントでこれらの変更を行ってください。 AWS

Confluence で以下を確認してください。

- Confluence Amazon Kendra インスタンス内のすべてのコンテンツを閲覧する権限を以下の方法で付与しました。
  - Amazon Kendra `confluence-administrators`グループのメンバーにする。
  - 既存のすべてのスペース、ブログ、ページにサイト管理者アクセス許可を付与する。
- Confluence インスタンスの URL をコピーしました。
- SSO (シングルサインオン) ユーザー向け: Confluence データセンターで Confluence 認証方法を設定するときに、ユーザー名とパスワードの [ログインページに表示] をアクティブ化しました。
- Confluence サーバー用
  - Amazon Kendraに接続するための Confluence 管理アカウントのユーザー名とパスワードを含む基本認証情報を記録しました。
  - オプション: Amazon Kendraに接続するための個人アクセストークンが Confluence アカウントで生成されました。詳細については、「個人用アクセストークンの生成に関する Confluence ドキュメント」を参照してください。
- Confluence クラウド用
  - Amazon Kendraに接続するための Confluence 管理アカウントのユーザー名とパスワードを含む基本認証情報を記録しました。
- 各ドキュメントが Confluence および同じインデックスに使用する予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Confluence の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Confluence データソースをに接続するときに、IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Confluence Amazon Kendra データソースに接続するには、データにアクセスできるように Amazon Kendra Confluence 認証情報の詳細を提供する必要があります。まだ Confluence を設定していない場合は、を参照してください。 Amazon Kendra 前提条件

Console

Confluence Amazon Kendra に接続するには

1. AWS 管理コンソールにサインインし、コンソールを開きます。Amazon Kendra

2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースを追加] ページで、[Confluence コネクタ V1.0] を選択し、[データソースを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 — 索引用のドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. ユースケースに基づいて [Confluence クラウド] と [Confluence サーバー] のどちらかを選択します。

   b. [Confluence クラウド]を選択した場合は、次の情報を入力します。

      i. [Confluence URL] - ユーザーの Confluence URL。

      ii. AWS Secrets Manager シークレット — Confluence Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

         - [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Confluence-' がシークレット名に自動的に追加されます。

II. [ユーザー名]と [パスワード] の場合 - Confluence ユーザー名と Confluence API トークンをパスワードとして入力します。

III. [認証を保存] を選択します。

c. [Confluence サーバー]を選択した場合は、次の情報を入力します。

i. [Confluence URL] - お客様の Confluence ユーザー名とパスワード。

ii. (オプション) [ウェブプロキシ] には次の情報を入力します。

A. [ホスト名] - Confluence アカウントのホスト名。

B. [ポート番号] - ホスト URL トランスポートプロトコルが使用するポート。

iii. [基本認証] と [個人用アクセストークン] のどちらかを選択します。

iv. AWS Secrets Manager シークレット — Confluence 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

- [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Confluence-' がシークレット名に自動的に追加されます。

II. [ユーザー名] と [パスワード] には、Confluence アカウントから生成してダウンロードした認証資格の値を入力します。基本認証を使用する場合は、Confluence のユーザー名とパスワードを認証情報として使用してください。個人用アクセストークンを使用する場合は、Confluence アカウントで作成した[個人用アクセストークン] の詳細を入力します。

III. [認証を保存] を選択します。

d. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [個人用スペースを含める] と [アーカイブされたスペースを含む] の場合 - このデータソースに含めるオプションのスペースタイプを選択します。

   b. [追加設定] の場合: 特定のコンテンツを含めるか除外する正規表現パターンを指定します。最大 100 のパターンを追加できます。

   c. 選択したスペース内の添付ファイルをクロールすることもできます。

   d. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

   e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. Space、Page、Blogの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドか、その他の推奨フィールドマッピングから選択してインデックスフィールドを追加します。

   b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Confluence Amazon Kendra に接続するには

API ConfluenceConfigurationを使用して以下を指定する必要があります。

- Confluence バージョン - CLOUD または SERVER として使用している Confluence インスタンスのバージョンを指定します。
- シークレットアマゾンリソースネーム (ARN) — Confluence Secrets Manager アカウントで作成した認証証明書を含むシークレットの Amazon リソースネーム (ARN) を指定します。

  Confluence サーバーを使用している場合は、Confluence のユーザー名とパスワード、または個人アクセストークンのいずれかを認証情報として使用できます。

  Confluence のユーザー名とパスワードを認証情報として使用する場合、以下の認証情報を JSON 構造としてシークレットに保存します。 Secrets Manager

```
{
    "username": "user name",
    "password": "password"
}
```

  個人アクセストークンを使用して Confluence Server に接続する場合 Amazon Kendra、以下の認証情報を JSON 構造としてシークレットに保存します。 Secrets Manager

```
{
    "patToken": "personal access token"
}
```

  Confluence Cloud Amazon Kendra をデータソースとして使用している場合、Confluence ユーザー名と Confluence アカウントで生成された API トークンをパスワードとして使用します。以下の認証情報を JSON 構造としてシークレットに保存します。 Secrets Manager

```
{
    "username": "user name",
    "password": "API token"
}
```

  > **Note**
  >
  > 認証情報とシークレットは、定期的に更新または口ーテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供したり、Confluence コネクターとに必要なパブリック API `RoleArn` を呼び出したりするタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Confluence data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- ウェブプロキシ - ウェブプロキシ経由で Confluence URL インスタンスに接続するかどうか。このオプションは Confluence サーバーに使用できます。
- (Confluence サーバーのみ) [仮想プライベートクラウド (VPC)] - データソース設定の一部として `VpcConfiguration` を指定します。VPC Amazon Kendra を使用するための設定を参照してください。
- 包含フィルターと除外フィルター - 特定のスペース、ブログ投稿、ページ、スペース、および添付ファイルを含めるか除外する正規表現パターンを指定します。添付ファイルのインデックスを作成する場合は、インデックスで指定されたページおよびブログへの添付ファイルのみがインデックス作成されます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング - 選択すると、Confluence データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

詳細はこちら

Confluence Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Confluence Amazon Kendra サーバーコネクターの設定

Confluence コネクタ V2.0

Confluence は、プロジェクト計画、ソフトウェア開発、製品管理の共有、保存、作業を目的とした共同作業管理ツールです。 Amazon Kendra を使用して、Confluence のスペース、ページ (ネストされたページを含む)、ブログ、インデックスに登録されたページやブログへのコメントや添付ファイルのインデックスを作成できます。

Amazon Kendra Confluence データソースコネクターのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順

サポートされている機能

Amazon Kendra Confluence データソースコネクターは以下の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外パターン
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Confluence データソースのインデックスを作成する前に、Confluence とアカウントでこれらの変更を行ってください。 AWS

Confluence で以下を確認してください。

- Confluence インスタンスの URL をコピーしました。例: *https://example.confluence.com*、*https://www.example.confluence.com/*、または *https:// atlassian.net/*。 Amazon Kendraに接続するには、Confluence インスタンス URL が必要です。

  Confluence ################ URL # atlassian.net/ で終わる必要があります。

  > **Note**
  >
  > 以下の URL の形式がサポートされています。
  >
  > - *https://example.confluence.com/xyz*
  > - *https://www.example.confluence.com//wiki/spacekey/xxx*
  > - *https:// atlassian.net/xyz*

  > **Note**
  >
  > (オンプレミス/サーバー) に含まれるエンドポイント情報が、 Amazon Kendra データソース設定の詳細で指定されているエンドポイント情報と同じかどうかを確認します。 AWS Secrets Manager 混乱する代理問題は、ユーザーがアクションを実行するアクセス許可がないにもかかわらず、 Amazon Kendra をプロキシとして使用して設定された秘密にアクセスし、アクションを実行するセキュリティの問題です。後でエンドポイント情報を変更する場合は、新しいシークレットを作成してこの情報を同期する必要があります。

- Confluence Amazon Kendra インスタンスへの接続を許可するユーザー名 (Confluence へのログインに使用される電子メール ID) とパスワード (Confluence サーバーのパスワード) を含む基本認証資格情報を設定しました。Confluence API トークンの作成方法については、「Atlassian アカウントの API トークンの管理」を参照してください。
- オプション:Confluence インスタンスに接続できるように、Confluence アプリキー、Confluence アプリシークレット、Confluence アクセストークン、および Confluence リフレッシュトークンを含む OAuth 2.0 認証情報を設定しました。 Amazon Kendra アクセストークンの有効期限が切れた

場合は、更新トークンを使用してアクセストークンと更新トークンのペアを再生成できます。または、認証プロセスを繰り返すこともできます。アクセストークンの詳細については、「OAuth アクセストークンの管理」を参照してください。

- (Confluence サーバーのみ) オプション:Confluence インスタンスに接続できるように Amazon Kendra 、Confluence トークンを含む個人用アクセストークン (PAT) を設定しました。PAT トークンの作成方法については、「個人用アクセストークンの使用」を参照してください。

には、 AWS アカウント次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Confluence の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Confluence データソースをに接続するときに、IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Confluence Amazon Kendra データソースに接続するには、データにアクセスできるように Amazon Kendra Confluence 認証情報の詳細を提供する必要があります。まだ Confluence を設定していない場合は、を参照してください。 Amazon Kendra 前提条件

Console

Confluence Amazon Kendra に接続するには

1. AWS 管理コンソールにサインインし、コンソールを開きます。Amazon Kendra
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースを追加する] ページで [Confluence コネクタ V2.0] を選択し、[コネクタを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語 — 索引用のドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [ソース] では、Confluence データソースのホスティング方法に基づいて [Confluence クラウド] と [Confluence サーバー] のいずれかを選択します。

b. Confluence の URL - Confluence のホスト URL を入力します。*####### URL ### ##https://example.confluence.com* です。

c. (Confluence サーバーのみ) SSL 証明書の場所-オプション- Confluence サーバーの SSL Amazon S3 証明書ファイルへのパスを入力します。

d. (Confluence Server のみ) [Web プロキシ-オプション] - ウェブプロキシの[ホスト名] (`http://` プロトコルまたは `https://` プロトコルなし) と [ポート番号] (ホスト URL トランスポートプロトコルが使用するポート) を入力します。ポート番号は 0 ~ 65535 の数字である必要があります。

e. (Confluence サーバーのみ) [承認] - [アクセスコントロールリスト (ACL)] を有効にするかどうかを選択します。次に、[ユーザー名] と [E メール] を選択して、アクセスコントロールに使用するフィールドを選択します。

f. ユースケースに応じて、[基本認証]、[OAuth 2.0 認証]、および (Confluence サーバーのみ) [個人用アクセストークン認証] のいずれかを選択します。

g. [AWS Secrets Manager シークレット] - Confluence の認証情報を保存する既存のシークレットを選択するか、新しい Secrets Manager シークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。

   i. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Confluence-' がシークレット名に自動的に追加されます。

   ii. [基本認証] を使用している場合 - Confluence アカウントから生成してダウンロードした [シークレット名]、[ユーザー名]、[パスワード] (Confluence サーバーパスワード) を入力します。

   [OAuth2.0 認証] を使用する場合 - Confluence アカウントで作成した [シークレット名]、[アプリキー]、[アプリシークレット]、[アクセストークン]、[更新トークン] を入力します。

   (Confluence サーバーのみ) [個人用アクセストークン認証] を使用する場合 - Confluence アカウントで作成した [シークレット名]と [Confluence トークン] を入力します。

   iii. [シークレットを保存して追加する] を選択します。

h. [VPC とセキュリティグループの設定 - オプション] で、[仮想プライベートクラウド (VPC)] では VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

i. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

j. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

k. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期スコープ] では、[コンテンツを同期] で、[ページ]、[ページコメント]、[ページ添付ファイル]、[ブログ]、[ブログコメント]、[ブログ添付ファイル]、[個人用スペース]、[アーカイブ済みスペース] のエンティティタイプから同期を選択します。

> **Note**
>
> [ページコメント] と [ページ添付ファイル] は、[ページ] を同期することを選択した場合にのみ削除できます。[ブログコメント] と [ブログ添付ファイル] は、[ブログ] を同期することを選択した場合にのみ削除できます。

> **Important**
>
> [追加設定] で [スペースキー] 正規表現パターンを指定しない場合、すべての [ページ] と [ブログ] がデフォルトでクロールされます。

b. [スペース正規表現パターン] の [追加設定] では、特定のスペースをインデックスに含めるか除外するかを次のように指定します。

- [スペースキー] - 例えば、*my-space-123* と入力します。

> **Note**
>
> [追加設定] で [スペースキー] 正規表現パターンを指定しない場合、すべての [ページ] と [ブログ] がデフォルトでクロールされます。

- URL – *#####. *//MySite/MyDocuments*。
- [ファイルタイプ] 例えば、*.*\.pdf, .*\.txt* です。

- 最大ファイルサイズの場合 — Amazon Kendra がクロールするファイルサイズの制限を MB 単位で指定します。Amazon Kendra は、定義したサイズ制限内のファイルのみをクロールします。デフォルトのファイルサイズは 50 MB です。最大ファイルサイズは 0 MB 以上 50 MB 以下でなければなりません。

- [エンティティタイトル正規表現パターン] の場合 - 特定の [ブログ]、[ページ]、[コメント]、[添付ファイル] をタイトル別に含めたり除外したりする正規表現パターンを指定します。

> **Note**
>
> 特定のページまたはサブページをクロールする場合は、ページタイトルの正規表現パターンを使用して、そのページを含めたり除外したりできます。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更、削除済みコンテンツを同期] - 新規、変更、削除されたコンテンツのみを同期します。

d. [同期実行スケジュール] で、[頻度]-[Amazon Kendra データソースと同期する頻度] を選択します。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. [スペース]、[ページ]、[ブログ]、[コメント]、[添付ファイル]- Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングする項目を選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Confluence Amazon Kendra に接続するには

API を使用してデータソーススキーマの JSON を指定する必要があります。TemplateConfigurationこれには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON CONFLUENCEV2 スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- ホスト URL - Confluence ホストインスタンスのバージョンを指定します。例えば、*https://example.confluence.com* などです。
- 同期モード - Amazon Kendra がすべてのドキュメントを同期してインデックスを更新するか、新しいドキュメント、変更されたドキュメント、削除されたドキュメントのみを同期するかどうかを指定します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。

  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- 認証タイプ - Confluence インスタンスの認証のタイプ (Basic、OAuth2、Personal-token) を指定します。
- (オプション - Confluence サーバーのみ) SSL 証明書の場所 - SSL 証明書の保存に使用した S3bucketName および s3certificateName を指定します。
- シークレットアマゾンリソースネーム (ARN) — Confluence Secrets Manager アカウントで作成した認証証明書を含むシークレットの Amazon リソースネーム (ARN) を指定します。基本アカウント認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "username": "Confluence account user name",
    "password": "Confluence API token"
}
```

OAuth 2.0 認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "confluenceAppKey": "app key for your Confluence account",
    "confluenceAppSecret": "app secret from your Confluence token",
    "confluenceAccessToken": "access token created in Confluence",
    "confluenceRefreshToken": "refresh token created in Confluence"
}
```

(Confluence サーバーのみ) 基本認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "hostUrl": "Confluence Server host URL",
    "username": "Confluence Server user name",
    "password": "Confluence Server password"
}
```

(Confluence サーバーのみ) パーソナルアクセストークン認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "hostUrl": "Confluence Server host URL",
    "patToken": "Confluence token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Confluence コネクターとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Confluence data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 特定のスペース、ページ、ブログ、およびそれらのコメントや添付ファイルを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはその

グループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

- フィールドマッピング - 選択すると、Confluence データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Confluence テンプレートスキーマ」を参照してください。

メモ

- 個人用アクセストークン (PAT) は Confluence クラウドでは使用できません。

カスタムデータソースコネクタ

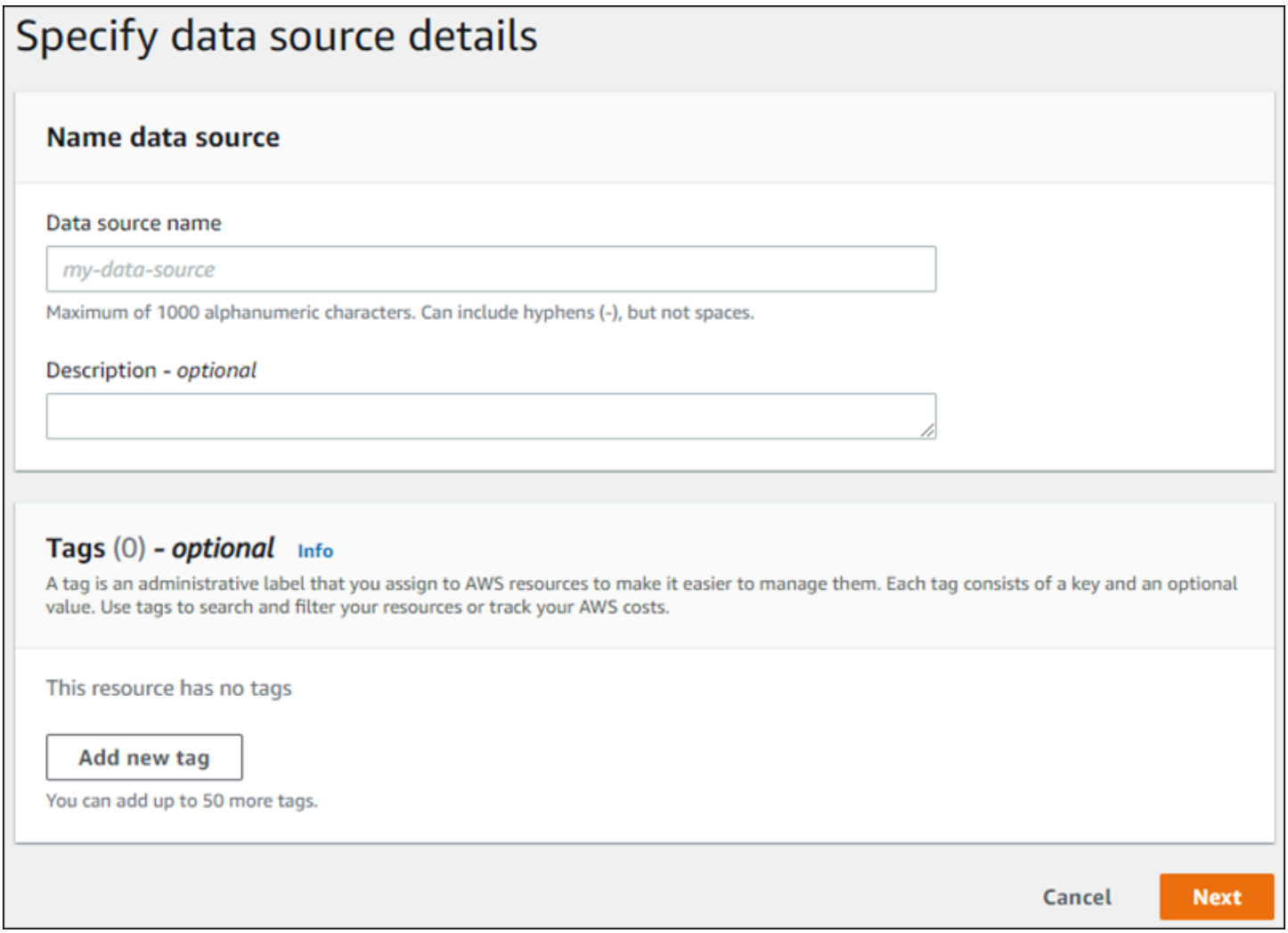
Amazon Kendra がまだデータソースコネクタを提供していないリポジトリがある場合は、カスタムデータソースを使用します。Amazon Kendra のデータソースを使用してリポジトリを同期できない場合でも、Amazon Kendra データソースが提供する同じ実行履歴メトリクスを確認できます。これを使用して、Amazon Kendra データソースとカスタムデータソース間で一貫した同期モニタリングエクスペリエンスを作成します。具体的には、カスタムデータソースを使用して、BatchPutDocumentおよび BatchDeleteDocument APIs を使用して作成したデータソースコネクタの同期メトリクスを確認します。

Amazon Kendra データソースコネクタのトラブルシューティングについては、「データソースのトラブルシューティング」を参照してください。

カスタムデータソースを作成すると、インデックスを作成するドキュメントの選択方法を完全に制御できます。Amazon Kendra はデータソース同期ジョブのモニタリングに使用できるメトリクス情報のみを提供します。データソースインデックスを決定するクローラを作成し、実行する必要があります。

Query 結果のレスポンスに と DocumentTitleDocumentURIを含める_source_uriDocumentAttributeには、ドキュメントオブジェクトを使用してドキュメントのメインタイトルを指定する必要があります。

コンソールまたは CreateDataSource API を使用して、カスタムデータソースの識別子を作成します。コンソールを使用するには、データソースに名前を付け、オプションで説明とリソースタグを指定します。データソースが作成されると、データソース ID が表示されます。この ID をコピーして、データソースをインデックスと同期するときに使用します。



CreateDataSource API を使用して、カスタムデータソースを作成することもできます。この API は、データソースを同期するときに使用する ID を返します。CreateDataSource API を使用してカスタムデータソースを作成する場合、Configuration、RoleArn または Schedule パラメータは設定できません。これらのパラメータを設定すると、Amazon Kendra は ValidationException 例外を返します。

カスタムデータソースを使用するには、Amazon Kendra インデックスの更新を担当するアプリケーションを作成します。アプリケーションは、作成するクローラによって異なります。クローラーはリポジトリ内のドキュメントを読み取り、Amazon Kendra に送信するドキュメントを決定します。アプリケーションでは、以下のステップを実行する必要があります。

1. リポジトリをクロールし、リポジトリ内の追加、更新、または削除されるドキュメントのリストを作成します。
2. StartDataSourceSyncJob API を呼び出して、同期ジョブが開始されていることを通知します。同期しているデータソースを識別するためのデータソース ID を指定します。 Amazon Kendra は、特定の同期ジョブを識別するために実行 ID を返します。
3. BatchDeleteDocument API を呼び出して、インデックスからドキュメントを削除します。同期しているデータソースと、この更新が関連付けられているジョブを識別するために、データソース ID と実行 ID を指定します。
4. StopDataSourceSyncJob API を呼び出して、同期ジョブの終了を通知します。StopDataSourceSyncJob API を呼び出すと、関連付けられた実行 ID は無効になります。
5. インデックスとデータソース識別子を使用して ListDataSourceSyncJobs API を呼び出して、データソースの同期ジョブを一覧表示し、同期ジョブのメトリクスを表示します。

同期ジョブを終了したら、新しい同期ジョブを開始できます。提出されたすべてのドキュメントがインデックスに追加されるまで期間がある場合があります。ListDataSourceSyncJobs API を使用して、同期ジョブのステータスを確認します。同期ジョブに対して返された Status が SYNCING_INDEXING の場合、一部のドキュメントはまだインデックス作成中です。前のジョブのステータスが FAILEDまたは になったら、新しい同期ジョブを開始できますSUCCEEDED。

StopDataSourceSyncJob API を呼び出した後、同期ジョブ識別子は、BatchPutDocument または BatchDeleteDocument API への呼び出には使えません。呼び出しに使用した場合、送信されたすべてのドキュメントは、API からの FailedDocuments レスポンスメッセージに返されます。

必須属性

BatchPutDocument API を使用して Amazon Kendra にドキュメントを送信すると、各ドキュメントにはドキュメントが属するデータソースと同期実行を識別するために 2 つの属性が必要です。カスタムデータソースのドキュメントを Amazon Kendra インデックスに正しくマッピングするには、次の 2 つの属性を指定する必要があります。

- _data_source_id - データソースの識別子。これは、コンソールまたは CreateDataSource API を使用してデータソースを作成したときに返されます。
- _data_source_sync_job_execution_id - 同期実行の識別子。これは、StartDataSourceSyncJob API とのインデックスの同期を開始したときに返されます。

カスタムデータソースを使用してドキュメントのインデックスを作成するために必要な JSON を次に示します。

```
{
    "Documents": [
        {
            "Attributes": [
                {
                    "Key": "_data_source_id",
                    "Value": {
                        "StringValue": "data source identifier"
                    }
                },
                {
                    "Key": "_data_source_sync_job_execution_id",
                    "Value": {
                        "StringValue": "sync job identifier"
                    }
                }
            ],
            "Blob": "document content",
            "ContentType": "content type",
            "Id": "document identifier",
            "Title": "document title"
        }
    ],
    "IndexId": "index identifier",
    "RoleArn": "IAM role ARN"
}
```

BatchDeleteDocument API を使用してインデックスからドキュメントを削除すると、DataSourceSyncJobMetricTarget パラメータで次の 2 つのフィールドを指定する必要があります。

- DataSourceId - データソースの識別子。これは、コンソールまたは CreateDataSource API を使用してデータソースを作成したときに返されます。
- DataSourceSyncJobId - 同期実行の識別子。これは、StartDataSourceSyncJob API とのインデックスの同期を開始したときに返されます。

以下は、BatchDeleteDocument API を使用してインデックスからドキュメントを削除するのに必要な JSON です。

```
{
    "DataSourceSyncJobMetricTarget": {
        "DataSourceId": "data source identifier",
        "DataSourceSyncJobId": "sync job identifier"
    },
    "DocumentIdList": [
        "document identifier"
    ],
    "IndexId": "index identifier"
}
```

メトリクスの表示

同期ジョブが完了したら、 DataSourceSyncJobMetrics API を使用して同期ジョブに関連付けられたメトリクスを取得できます。これを使用して、カスタムデータソースの同期をモニタリングします。

同じドキュメントを複数回提出する場合、BatchPutDocument API、BatchDeleteDocument API のいずれかの一部で、ドキュメントが追加と削除の両方で送信された場合、ドキュメントはメトリクスで一度だけカウントされます。

- DocumentsAdded - インデックスに初めて追加されたこの同期ジョブに関連付けられた BatchPutDocument API で送信されたドキュメントの数。ドキュメントが同期で複数回追加されるように送信された場合、そのドキュメントはメトリクスで 1 回だけカウントされます。
- DocumentsDeleted - インデックスから削除されたこの同期ジョブに関連付けられた BatchDeleteDocument API を使用して送信されたドキュメントの数。ドキュメントが同期で複数回削除されるように送信された場合、そのドキュメントはメトリクスで 1 回だけカウントされます。

- DocumentsFailed - インデックス作成に失敗したこの同期ジョブに関連付けられているドキュメントの数。これらは、Amazon Kendra がインデックス作成のために受け入れましたが、インデックス作成または削除はできなかったドキュメントです。Amazon Kendra によってドキュメントが受け入れられない場合、ドキュメントの識別子は BatchPutDocument および BatchDeleteDocument API の FailedDocuments レスポンスプロパティに返されます。
- DocumentsModified - Amazon Kendra インデックスで変更されたこの同期ジョブに関連付けられた BatchPutDocument API を使用して送信された、変更されたドキュメントの数。

Amazon Kendra は、ドキュメントのインデックス作成中に Amazon CloudWatch メトリクスも発行します。詳細については、「Amazon CloudWatch による Amazon Kendra のモニタリング」を参照してください。

Amazon Kendra はカスタムデータソースの DocumentsScanned メトリクスを返しません。また、Amazon Kendra データソースのメトリクスドキュメントに記載されている CloudWatch メトリクスも出力します。

詳細

Amazon Kendra とカスタムデータソースの統合について詳しくは、以下をご覧ください。

- Amazon Kendra へのカスタムデータソースの追加

カスタムデータソース (Java)

以下のコードは、Java を使用したカスタムデータソースの実装の例を示します。プログラムはまずカスタムデータソースを作成し、次に新しく追加されたドキュメントをカスタムデータソースを持つインデックスに同期します。

次のコードは、カスタムデータソースを作成して使用する方法を示しています。アプリケーションでカスタムデータソースを使用している場合は、インデックスをデータソースと同期するたびに新しいデータソースを作成する必要はありません (1 回限りのプロセス)。インデックス ID とデータソース ID を使用してデータを同期します。

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import csoftware.amazon.awssdk.services.kendra.model.BatchPutDocumentRequest;
import csoftware.amazon.awssdk.services.kendra.model.BatchPutDocumentResponse;
```

```
import software.amazon.awssdk.services.kendra.model.CreateDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.DataSourceType;
import software.amazon.awssdk.services.kendra.model.Document;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsRequest;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsResponse;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobRequest;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobResponse;
import software.amazon.awssdk.services.kendra.model.StopDataSourceSyncJobRequest;
import software.amazon.awssdk.services.kendra.model.StopDataSourceSyncJobResponse;

public class SampleSyncForCustomDataSource {
  public static void main(String[] args) {
    KendraClient kendra = KendraClient.builder().build();

    String myIndexId = "yourIndexId";
    String dataSourceName = "custom data source";
    String dataSourceDescription = "Amazon Kendra custom data source connector"

    // Create custom data source
    CreateDataSourceRequest createDataSourceRequest = CreateDataSourceRequest
        .builder()
        .indexId(myIndexId)
        .name(dataSourceName)
        .description(dataSourceDescription)
        .type(DataSourceType.CUSTOM)
        .build();

    CreateDataSourceResponse createDataSourceResponse =
 kendra.createDataSource(createDataSourceRequest);
    System.out.println(String.format("Response of creating data source: %s",
 createDataSourceResponse));

    // Get the data source ID from createDataSourceResponse
    String dataSourceId = createDataSourceResponse.Id();

    // Wait for the custom data source to become active
    System.out.println(String.format("Waiting for Amazon Kendra to create the data
 source %s", dataSourceId));
    // You can use the DescribeDataSource API to check the status
    DescribeDataSourceRequest describeDataSourceRequest = DescribeDataSourceRequest
        .builder()
        .indexId(myIndexId)
        .id(dataSourceId)
```

```
        .build();

   while (true) {
       DescribeDataSourceResponse describeDataSourceResponse =
kendra.describeDataSource(describeDataSourceRequest);

       DataSourceStatus status = describeDataSourceResponse.status();
       System.out.println(String.format("Creating data source. Status: %s", status));
       if (status != DataSourceStatus.CREATING) {
           break;
       }

       TimeUnit.SECONDS.sleep(60);
   }

   // Start syncing yor data source by calling StartDataSourceSyncJob and providing
your index ID
   // and your custom data source ID
   System.out.println(String.format("Synchronize the data source %s", dataSourceId));
   StartDataSourceSyncJobRequest startDataSourceSyncJobRequest =
StartDataSourceSyncJobRequest
       .builder()
       .indexId(myIndexId)
       .id(dataSourceId)
       .build();
   StartDataSourceSyncJobResponse startDataSourceSyncJobResponse =
kendra.startDataSourceSyncJob(startDataSourceSyncJobRequest);

   // Get the  sync job execution ID from startDataSourceSyncJobResponse
   String executionId = startDataSourceSyncJobResponse.ExecutionId();

   // Add 2 documents uploaded to S3 bucket to your index using the BatchPutDocument
API
   // The added documents should sync with your custom data source
   Document pollyDoc = Document
       .builder()
       .s3Path(
           S3Path.builder()
           .bucket("s3-test-bucket")
           .key("what_is_Amazon_Polly.docx")
           .build())
       .title("What is Amazon Polly?")
       .id("polly_doc_1")
       .build();
```

```
    Document rekognitionDoc = Document
        .builder()
        .s3Path(
            S3Path.builder()
            .bucket("s3-test-bucket")
            .key("what_is_amazon_rekognition.docx")
            .build())
        .title("What is Amazon rekognition?")
        .id("rekognition_doc_1")
        .build();

    BatchPutDocumentRequest batchPutDocumentRequest = BatchPutDocumentRequest
        .builder()
        .indexId(myIndexId)
        .documents(pollyDoc, rekognitionDoc)
        .build();

    BatchPutDocumentResponse result = kendra.batchPutDocument(batchPutDocumentRequest);
    System.out.println(String.format("BatchPutDocument result: %s", result));

    // Wait for the sync job status to succeed
    // If the sync job status is SYNCING_INDEXING, documents are still being indexed
    // If the sync job status is SYNCING, sync job has started
    System.out.println(String.format("Waiting for the data
source to sync with the index %s for execution ID %s", indexId,
startDataSourceSyncJobResponse.executionId()));
    ListDataSourceSyncJobsRequest listDataSourceSyncJobsRequest =
ListDataSourceSyncJobsRequest
        .builder()
        .indexId(myIndexId)
        .id(dataSourceId)
        .build();

    while (true) {
        ListDataSourceSyncJobsResponse listDataSourceSyncJobsResponse =
kendra.listDataSourceSyncJobs(listDataSourceSyncJobsRequest);
        DataSourceSyncJob job = listDataSourceSyncJobsResponse.history().get(0);
        System.out.println(String.format("Syncing data source. Status: %s",
job.status()));

        if (job.status() != DataSourceSyncJobStatus.SYNCING) {
            break;
        }
```

```
        TimeUnit.SECONDS.sleep(60);

    }

    // Once custom data source synced, stop the sync job using the
  StopDataSourceSyncJob API
    StopDataSourceSyncJobResponse stopDataSourceSyncJobResponse =
  kendra.stopDataSourceSyncJob(
        StopDataSourceSyncJobRequest()
            .indexId(myIndexId)
            .id(dataSourceId)
    );
   }
}
```

Dropbox

Dropbox は、クラウドストレージ、ドキュメント整理、ドキュメントテンプレートサービスを提供するファイルホスティングサービスです。Dropbox ユーザーの場合は、Dropbox ファイル、Dropbox Paper、Dropbox Paper テンプレート、 Amazon Kendra 保存されているウェブページへのショートカットのインデックスを作成できます。特定の Dropbox ファイル、Dropbox Paper、Dropbox Paper テンプレート、 Amazon Kendra 保存されているウェブページへのショートカットのインデックスを設定することもできます。

Amazon Kendra Dropbox Business では Dropbox と Dropbox Advanced の両方をサポートしています。

Amazon Kendra コンソールと API を使用して Dropbox Amazon Kendra データソースに接続できます。TemplateConfiguration

Amazon Kendra Dropbox データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Dropbox データソースコネクタは以下の機能をサポートしています。

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Dropbox データソースのインデックスを作成する前に、Dropbox とアカウントにこれらの変更を加えてください。 AWS

Dropbox で以下を確認してください。

- Dropbox Advanced アカウントを作成し、管理者ユーザーを設定しました。
- 独自の [アプリ名] を使用して Dropbox アプリを作成し、[スコープ付きアクセス] を有効化しました。アプリの作成については Dropbox のドキュメントを参照してください。
- Dropbox コンソールで [フル Dropbox] アクセス許可を有効にし、次のアクセス許可を追加しました。
  - files.content.read
  - files.metadata.read
  - sharing.read
  - file_requests.read
  - groups.read
  - team_info.read
  - team_data.content.read
- 基本認証情報として Dropbox アプリキー、Dropbox アプリシークレット、Dropbox アクセストークンを記録しました。
- Dropbox アプリ用の一時的な OAuth 2.0 アクセストークンを生成してコピーしました。このトークンは一時的なもので、4 時間後に有効期限が切れます。OAuth 認証については Dropbox のドキュメントを参照してください。

> **Note**
>
> 4 時間後に有効期限が切れる 1 回限りのアクセストークンに頼るのではなく、有効期限のない Dropbox 更新アクセストークンを作成することをお勧めします。更新アクセストークンは永続的で有効期限がないため、今後もデータソースを同期し続けることができます。

- 推奨：中断することなくデータソースの同期を継続できるように Amazon Kendra 、有効期限のない Dropbox 永久更新トークンを設定しました。更新トークンについては Dropbox のドキュメントを参照してください。
- 各ドキュメントが Dropbox および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Dropbox の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Dropbox IAM Secrets Manager データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Dropbox Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Dropbox データソースに関する必要な詳細情報を入力する必要があります。Dropbox をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Dropbox Amazon Kendra に接続するには

1. AWS Management Console Amazon Kendra にログインしてコンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Dropbox コネクタ] を選択し、[コネクタを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-インデックスの対象となるドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [認証トークンのタイプ] - ユースケースに基づいて、[永久トークン (推奨)] と [アクセストークン (一時使用)] のいずれかを選択します。

   b. AWS Secrets Manager シークレット — 既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成して Dropbox の認証情報を保存します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

         A. [シークレット名] - シークレットの名前。シークレット名には「AmazonKendra-Dropbox-」というプレフィックスが自動的に追加されます。

         B. [アプリキー]、[アプリシークレット]、トークン情報 (永続的または一時的) の場合 - Dropbox アカウントから生成した認証情報の値を入力します。

      ii. [保存] を選択します。

   c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

   d. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

      > **Note**
      >
      > IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

   e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [エンティティまたはコンテンツタイプを選択] - クロールするエンティティまたはコンテンツタイプを選択します。

   b. [ログモードを変更] - すべてのファイルを同期する代わりにインデックスを更新することを選択します。

   c. [正規表現パターン] の [追加設定] - 特定のファイルを含めたり除外する正規表現パターンを追加します。

d. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. ファイル、Dropbox Paper、Dropbox Paper テンプレート- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングする項目を選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Dropbox に接続するには Amazon Kendra

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON DROPBOX スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- 変更ログ — Dropbox のデータソース変更ログメカニズムを使用して、 Amazon Kendra インデックス内のドキュメントを更新する必要があるかどうかを判断すべきかどうか。

> **Note**
>
> Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、変更ログを処理するよりも Dropbox Amazon Kendra データソース内のドキュメントをスキャンするほうが時間がかからない場合があります。Dropbox データソースをインデックスに初めて同期する場合は、すべてのドキュメントがスキャンされます。

- シークレットアマゾンリソースネーム (ARN) — Dropbox Secrets Manager アカウントの認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "appKey": "Dropbox app key",
    "appSecret": "Dropbox app secret",
    "accesstoken": "temporary access token or refresh access token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットへのアクセス権限をロールに付与し、Dropbox コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Dropbox data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 特定のファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト（ACL）をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Dropbox データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

  > **Note**
  >
  > 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Dropbox テンプレートスキーマ」を参照してください。

詳細はこちら

Amazon Kendra と Dropbox データソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendraの Dropbox コネクタを使用して Dropbox コンテンツのインデックスを作成してください

Drupal

Drupal は、ウェブサイトやウェブアプリケーションの作成に使用できるオープンソースのコンテンツ管理システム (CMS) です。を使用して Amazon Kendra 、Drupal で次のインデックスを作成できます。

- コンテンツ - 記事、基本ページ、基本ブロック、ユーザー定義コンテンツタイプ、ユーザー定義ブロックタイプ、カスタムコンテンツタイプ、カスタムブロックタイプ
- コメント - すべてのコンテンツタイプとブロックタイプに対応
- 添付ファイル - すべてのコンテンツタイプとブロックタイプに対応

Amazon Kendra コンソールまたは TemplateConfiguration API を使用して Drupal データソース Amazon Kendra に接続できます。

Amazon Kendra Drupal データソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

Amazon Kendra Drupal データソースコネクタは、次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Drupal データソースのインデックスを作成する前に、Drupal と AWS アカウントでこれらの変更を行ってください。

Drupal で以下を確認してください。

- Drupal (スタンダード) Suite のアカウントと管理者ロールを持つユーザーを作成しました。
- Drupal サイト名をコピーし、ホスト URL を設定しました。例えば、*https://<hostname>/<drupalsitename>*。
- ユーザー名 (Drupal ウェブサイトのログインユーザー名) とパスワード (Drupal ウェブサイトのパスワード) を含む基本認証情報を設定しました。
- 推奨: OAuth 2.0 認証情報トークンを設定しました。このトークンを、接続先の Drupal パスワード付与、クライアント ID、クライアントシークレット、ユーザー名 (Drupal ウェブサイトの口

グインユーザー名)、パスワード (Drupal ウェブサイトのパスワード) とともに使用して Amazon Kendra に接続します。

- 管理者ロールを使用して Drupal アカウントに次のアクセス許可を追加しました。
  - ブロックを管理
  - block_content の表示を管理
  - block_content フィールドを管理
  - block_content の形式表示を管理
  - ビューを管理
  - ユーザーの E メールアドレスを表示
  - 自分の未公開コンテンツを表示
  - ページリビジョンを表示
  - 記事のリビジョンを表示
  - すべてのリビジョンを表示
  - 管理テーマを表示
  - コンテンツへのアクセス
  - コンテンツへのアクセスの概要
  - コメントへのアクセス
  - コンテンツを検索
  - ファイルへのアクセスの概要
  - コンテキストリンクへのアクセス

> **Note**
>
> ユーザー定義のコンテンツタイプまたはユーザー定義のブロックタイプがある場合、またはビューやブロックを Drupal ウェブサイトに追加する場合は、それらに管理者アクセスを提供する必要があります。

で AWS アカウント、以下があることを確認します。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。

- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- Drupal の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、Drupal データソースを に接続するときに、コンソールを使用して新しい IAM ロールと Secrets Manager シークレットを作成できます Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Drupal データソース Amazon Kendra に接続するには、 がデータ Amazon Kendra にアクセスできるように Drupal 認証情報の詳細を入力する必要があります。の Drupal をまだ設定していない場合は、Amazon Kendra 「」を参照してください前提条件。

Console

Drupal Amazon Kendra に接続するには

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加ページで、Drupal connector を選択し、コネクタの追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 - 言語を選択して、ドキュメントをフィルタリングしてインデックスを作成します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] の [ホスト URL] - Drupal サイトのホスト URL。例えば、*https://<hostname>/<drupalsitename>*。

   b. [SSL 証明書の場所] - Amazon S3 バケットに保存されている SSL 証明書へのパスを入力します。

   c. 承認 — ACL があり、それをアクセスコントロールに使用する場合は、ドキュメントのアクセスコントロールリスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

   d. 認証用 - ユースケースに応じて、[基本認証] と [OAuth 2.0 認証] のいずれかを選択します。

   e. AWS Secrets Manager secret — 既存のシークレットを選択するか、新しい Secrets Manager シークレットを作成して Drupal 認証情報を保存します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [基本認証] を選択した場合は、コピーした [シークレット名]、[ユーザー名] (Drupal サイトのユーザー名)、および [パスワード] (Drupal サイトのパスワード) を入力し、[保存してシークレットを追加] を選択します。

B. [OAuth 2.0 認証] を選択した場合は、Drupal アカウントで生成された [シークレット名]、[ユーザー名] (Drupal サイトのユーザー名)、[パスワード] (Drupal サイトのパスワード)、[クライアント ID]、および [クライアントシークレット] を入力して、[シークレットを保存して追加] を選択します。

ii. [保存] を選択します。

f. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

g. ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

h. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] は、次のオプションから選択します。

> **Note**
>
> [記事]、[基本ページ]、[基本ブロック] のクロールを選択すると、それぞれのデフォルトのフィールドが自動的に同期されます。コメント、添付ファイル、カスタムフィールド、その他のカスタムエンティティを同期することもできます。

- [エンティティを選択] の場合。
  - 記事 - [記事]、そのコメント [コメント]、および [添付ファイル]をクロールするかどうかを選択します。
  - [基本ページ] - [基本ページ]、その [コメント]、その [添付ファイル] をクロールするかどうかを選択します。
  - [基本ブロック] - [基本ブロック]、その [コメント]、その [添付ファイル] をクロールするかどうかを選択します。
  - [カスタムコンテンツタイプ] と [カスタムブロック] を追加することもできます。

b. [追加設定 - オプション]。
   - [正規表現パターン] - 特定のエンティティタイトルとファイル名を含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを Amazon Kendra 初めて と同期すると、デフォルトですべてのコンテンツが同期されます。
   - [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
   - [新規、変更済み、または削除されたコンテンツを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. コンテンツ 、コメント 、添付ファイル - インデックスにマッピングする Amazon Kendra デフォルトのデータソースフィールドから選択します。

   b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Drupal Amazon Kendra に接続するには

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — JSON スキーマを使用するDRUPALときに、データソースタイプを TemplateConfiguration として指定します。また、 CreateDataSource API を呼び出すTEMPLATEときにデータソースを として指定します。
- 同期モード - すべてのドキュメントを同期するか、新規、変更、削除されたドキュメントのみを同期するかを指定して、インデックス Amazon Kendra を更新します。以下のいずれかから選択できます。
  - すべてのコンテンツをクロールしてインデックスに同期する FORCED_FULL_CRAWL
  - すべてのコンテンツをクロールし、新規、変更、または削除されたコンテンツのみを同期する FULL_CRAWL
  - 新規、変更、削除したコンテンツのみをクロールして同期する CHANGE_LOG
- シークレット Amazon リソースネーム (ARN) - Drupal アカウントで作成した認証情報を含む Secrets Manager シークレットの Amazon リソースネーム (ARN) を指定します。

  基本認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password"
}
```

  OAuth 2.0 認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
```

```
    "password": "password",
    "clientId": "client id",
    "clientSecret": "client secret"
}
```

> **Note**
>
> > **Note**
> >
> > 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- role IAM — を呼び出しCreateDataSourceて、 Secrets Manager シークレットにアクセスするためのアクセス許可を IAM ロールに付与し、Drupal コネクタと に必要なパブリック APIs を呼び出すRoleArnタイミングを指定します Amazon Kendra。詳細については、「IAM roles for Drupal data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - コンテンツ、コメント、添付ファイルを含めるかどうかを指定できます。また、コンテンツ、コメント、添付ファイルを含めるか除外する正規表現パターンを指定することもできます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定

した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、 Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。
- フィールドマッピング - 選択すると、Drupal データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> がドキュメント Amazon Kendra を検索するには、ドキュメント本文フィールドまたはドキュメントと同等のドキュメント本文が必要です。データソース内のドキュメント本文フィールド名をインデックスフィールド名 にマッピングする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Drupal テンプレートスキーマ」を参照してください。

メモ

- Drupal API には公式なスロットリング制限はありません。
- Java SDK は Drupal では使用できません。
- Drupal データは、ネイティブ JSON API を使用してのみ取得できます。
- どの Drupal [ビュー] にも関連付けられていないコンテンツタイプはクロールできません。
- Drupal [ブロック] からデータをクロールするには、管理者権限が必要です。
- HTTP 動詞を使用してユーザー定義コンテンツタイプを作成するための JSON API はありません。

- [記事]、[基本ページ]、[基本ブロック]、ユーザー定義コンテンツタイプ、ユーザー定義ブロックタイプのドキュメント本文とコメントは HTML 形式で表示されます。HTML コンテンツの形式が正しくない場合、HTML 関連のタグがドキュメント本文とコメントに表示され、 Amazon Kendra 検索結果に表示されます。
- 説明または本文のないコンテンツタイプとブロックタイプは に取り込まれません Amazon Kendra。このようなコンテンツまたはブロックタイプのコメントと添付ファイルのみが Amazon Kendra インデックスに取り込まれます。

GitHub

GitHub は、バージョン管理機能を備えたコードストレージおよび管理サービスを提供するソフトウェア開発用の Web ベースのホスティングサービスです。 Amazon Kendra を使用して、 GitHub Enterprise Cloud (SaaS) と GitHub Enterprise Server (オンプレミス) のリポジトリファイル、イシューとプルリクエスト、イシューとプルリクエストのコメント、イシューとプルリクエストのコメント添付ファイルのインデックスを作成できます。また、特定のファイルを含めるまたは除外することもできます。

> **Note**
>
> Amazon Kendra GitHub アップグレードされたコネクタをサポートするようになりました。コンソールは自動的にアップグレードされました。コンソールに新しいコネクタを作成すると、アップグレードされたアーキテクチャが使用されます。API を使用する場合は、TemplateConfigurationオブジェクトではなくオブジェクトを使用してコネクタを設定する必要があります。`GitHubConfiguration`
> 古いコンソールと API アーキテクチャを使用して設定されたコネクタは、引き続き設定どおりに機能します。ただし、編集や更新はできません。コネクタ構成を編集または更新する場合は、新しいコネクタを作成する必要があります。
> コネクタワークフローをアップグレードされたバージョンに移行することをお勧めします。古いアーキテクチャを使用して構成されたコネクタSupport は、2024 年 6 月までに終了する予定です。

Amazon Kendra コンソールと TemplateConfigurationAPI Amazon Kendra GitHub を使用してデータソースに接続できます。

Amazon Kendra GitHub データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

Amazon Kendra GitHub データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザー ID クロール
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra GitHub を使用してデータソースのインデックスを作成する前に、 GitHub AWS とアカウントでこれらの変更を行ってください。

で GitHub、次のものが揃っていることを確認してください。

- GitHub GitHub 組織の管理者権限を持つユーザーを作成した。
- 認証情報用のクラシック個人アクセストークンを作成しました。GitHub 個人アクセストークンの作成に関するドキュメントを参照してください。
- 推奨: 認証情報用の OAuth トークンを作成済み。API のスロットル制限とコネクタのパフォーマンスを向上させるには、OAuth トークンを使用してください。OAuth GitHub 認証に関するドキュメントを参照してください。
- GitHub GitHub使用しているサービスの種類のホスト URL をメモしておきました。たとえば、GitHub クラウドのホスト URL は *https://api.github.com* で、 GitHub サーバーのホスト URL は *on-prem-host-urlhttps:///api/v3/* とすることができます。
- GitHub接続するエンタープライズクラウド (SaaS) GitHub アカウントまたはエンタープライズサーバー (オンプレミス) アカウントの組織名を書き留めました。 GitHub 組織名は、 GitHub デス

クトップにログインし、プロフィール写真のドロップダウンから [所属組織] を選択すると確認できます。

- オプション (サーバーのみ): SSL 証明書を生成し、 Amazon S3 バケットに保存されている証明書へのパスをコピーしました。安全な SSL GitHub 接続が必要な場合は、これを使用して接続します。OpenSSL を使用して、任意のコンピュータで自己署名 X509 証明書を生成できます。OpenSSL を使用して X509 証明書を作成する例については、「X509 証明書の作成と署名」を参照してください。
- 以下のアクセス許可を追加しました。

  GitHub エンタープライズクラウド (SaaS) 向け

  - repo:status
  - public_repo
  - repo:invite
  - read:org
  - user:email
  - read:user

  GitHub エンタープライズサーバー (オンプレミス) 用

  - repo:status
  - public_repo
  - repo:invite
  - read:org
  - user:email
  - read:user
  - site_admin
- GitHub 同じインデックスに使用する予定の他のデータソースとドキュメントがそれぞれ異なることを確認した。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。

- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- GitHub AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager GitHub データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 GitHub Amazon Kendra データにアクセスできるようにデータソースの必要な詳細情報を入力する必要があります。 GitHub をまだ設定していない場合は Amazon Kendra、 GitHub を参照してください前提条件。

Console

Amazon Kendra に接続するには GitHub

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [GitHub コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. GitHubソース — GitHub GitHubエンタープライズクラウドとエンタープライズサーバーのどちらかを選択します。
   b. GitHub ホスト URL — GitHub ホスト名を入力します。
   c. GitHub 組織名 — GitHub 組織名を入力します。 GitHub 組織情報はアカウントで確認できます。
   d. 承認 — ACL があり、それをアクセス制御に使用したい場合は、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
   e. AWS Secrets Manager secret — Secrets Manager GitHub 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraGitHub-' がシークレット名に自動的に追加されます。

B. GitHubトークンの場合 — アカウントで作成した認証資格値を入力します。GitHub

ii. [保存] を選択します。

f. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

g. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. クロールするリポジトリの選択-[すべてのリポジトリをクロールする] か [リポジトリを選択] を選択します。

[リポジトリを選択] を選択した場合は、[リポジトリ名] にリポジトリの名前を追加し、オプションで [ブランチ名] に特定のブランチの名前を追加します。

b. コンテンツタイプ — 含めたいコンテンツタイプを選択します。

c. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- 新規または変更されたコンテンツの同期-新しいコンテンツと変更されたコンテンツのみを同期します。
- [新規、変更、削除済みコンテンツを同期] - 新規、変更、削除されたコンテンツのみを同期します。

9. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

10. [次へ] を選択します。

11. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. リポジトリ、リポジトリコミット、Issue ドキュメント、Issue コメント、Issue 添付ファイル、Pull Request コメント、Pull Request ドキュメント、Pull Request 添付ファイルの場合 — Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

12. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra GitHub

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON GITHUB スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- GitHubタイプ — **SAAS ON_PREMISE** タイプをまたはとして指定します。
- ホスト URL — GitHub ホスト URL または API エンドポイント URL を指定します。たとえば、 GitHub SaaS/Enterprise Cloud を使用する場合、ホスト URL はになり`https://api.github.com`、 GitHub オンプレミス/エンタープライズサーバーの場合はホスト URL がになります。`https://on-prem-host-url/api/v3/`
- 組織名 — アカウントの組織名を指定します。 GitHub 組織名は、 GitHub デスクトップにログインし、プロフィール写真のドロップダウンから「所属組織」を選択すると表示されます。
- 同期モード-すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 GitHub シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "personalToken": "token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 GitHub Amazon Kendra詳細については、「IAM GitHub データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。

> **Note**
>
> GitHub server を使用する場合は、 Amazon VPC GitHubを使用してサーバーに接続する必要があります。

- リポジトリフィルター — 名前とブランチ名でリポジトリをフィルターします。
- ドキュメント/コンテンツタイプ — リポジトリドキュメント、Issue、Issue コメント、Issue コメント添付ファイル、Pull Request、Pull Request コメント、Pull Request コメント添付ファイルをクロールするかどうかを指定します。
- 包含フィルターと除外フィルター — 特定のファイルやフォルダーを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- アクセス制御リスト (ACL) — ACL があり、それをアクセス制御に使用したい場合に、文書の ACL 情報をクロールするかどうかを指定します。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング — GitHub Amazon Kendra データソースフィールドをインデックスフィールドにマップすることを選択します。ドキュメント、コミット、Issue、Issue 添付ファイル、Issue コメント、プルリクエスト、プルリクエスト添付ファイル、プルリクエストコメントのフィールドを含めることができます。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> Amazon Kendra がドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。データソース内のドキュメント本文フィールド名をインデックスフィールド名 _document_body にマッピングする必要があります。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「GitHub template schema」を参照してください。

詳細はこちら

Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。 GitHub

- GitHub コネクターの力を借りてリポジトリの検索を再考しましょう。 Amazon Kendra GitHub

Gmail

Gmail は Google が開発した E メールクライアントで、添付ファイル付きのメールメッセージを送信できます。Gmail のメッセージは、フォルダやラベルを使用して E メールの受信トレイ内で分類して保存できます。 Amazon Kendra を使用して、メールメッセージと添付ファイルのインデックスを作成できます。また、特定のメールメッセージ、添付ファイル、 Amazon Kendra ラベルをインデックス対象に含めたり除外したりするように設定することもできます。

Amazon Kendra コンソールと Amazon Kendra API を使用して Gmail データソースに接続できます。TemplateConfiguration

Amazon Kendra Gmail データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

を使用して Gmail Amazon Kendra データソースをインデックスに登録する前に、Gmail とアカウントでこれらの変更を行ってください。 AWS

Gmail で以下を確認してください。

- Google Cloud Platform の管理者アカウントを作成し、Google Cloud プロジェクトを作成しました。
- 管理者アカウントで Gmail API と管理者 SDK API を有効にしました。
- サービスアカウントを作成し、Gmail の JSON プライベートキーをダウンロードしました。プライベートキーを作成してアクセスする方法については、Google Cloud のドキュメントの「サービスアカウントキーの作成方法」と「サービスアカウントの認証情報」を参照してください。
- 認証に使用する管理者アカウントの E メール、サービスアカウントの E メール、プライベートキーをコピーしました。
- ユーザーおよびインデックスを作成する共有ディレクトリに、次の OAuth スコープ (管理者ロールを使用) を追加しました。
  - https://www.googleapis.com/auth/admin.directory.user.readonly
  - https://www.googleapis.com/auth/gmail.readonly
- 各ドキュメントが Gmail および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

あなたには AWS アカウント、次のものがあることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Gmail の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Gmail IAM Secrets Manager データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Gmail Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Gmail の認証情報を入力する必要があります。Gmail をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Gmail Amazon Kendra に接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Gmail コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-インデックスの対象となるドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [AWS Secrets Manager シークレット認証] — Gmail Secrets Manager の認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。シークレットを新規作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。

- [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。

B. [クライアント E メール] - Google サービスアカウントからコピーしたクライアント E メール。

C. [管理者アカウント E メール] - 使用する管理者アカウントの E メールです。

D. [プライベートキー] - Google サービスアカウントからコピーしたプライベートキー。

E. [保存] を選択します。

b. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

c. IAM ロール — IAM 既存のロールを選択するか、 IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

d. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] の、[エンティティタイプ] で - [メッセージ添付ファイル] を選択してメッセージ添付ファイルを同期します。メッセージはデフォルトで同期されます。

b. (オプション) [追加設定] で、以下の情報を入力します。

   i. [日付範囲] - 日付範囲を入力して、クロールするメールの開始日と終了日を指定します。

   ii. [E メールドメイン] - ドメインに基づいてメールを含めたり除外します。

   iii. 件名のキーワード - 件名のキーワードに基づいてメールを含めたり除外します。

> **Note**
>
> また、入力した件名のキーワードすべてに一致するドキュメントを含めることもできます。

   iv. [ラベル] - 特定のラベルを含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

   v. [添付ファイル] - 特定の添付ファイルを含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更、削除済みコンテンツを同期] - 新規、変更、削除されたコンテンツのみを同期します。

> **Important**
>
> 完全に削除された Gmail メッセージを更新する API がないため、[新規、変更、削除したコンテンツの同期] は以下のようになります。
>
> - Gmail から完全に削除されたメッセージは Amazon Kendra インデックスから削除されません
> - Gmail のメールラベルの変更は同期されません。

Gmail のデータソースラベルの変更や完全に削除されたメールメッセージを Amazon Kendra インデックスに同期するには、定期的にフルクロールを実行する必要があります。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. メッセージとメッセージ添付ファイルの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングしたいものを選択します。

> **Note**
>
> Amazon Kendra Gmail データソースコネクタは API の制限によりカスタムインデックスフィールドの作成をサポートしていません。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Gmail Amazon Kendra に接続するには

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON `GMAIL` スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI `TEMPLATE` を呼び出すときと同じようにデータソースを指定します。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のいずれかから選択できます。
  - `FORCED_FULL_CRAWL` は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。

- FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。

> **Important**
>
> 完全に削除された Gmail メッセージを更新する API がないため、FULL_CRAWL/[新規、変更、削除したコンテンツの同期] は以下のようになります。
>
> - Gmail から完全に削除されたメッセージは Amazon Kendra インデックスから削除されません
> - Gmail のメールラベルの変更は同期されません。
>
> Gmail Amazon Kendra のデータソースラベルの変更や完全に削除されたメールをインデックスに同期するには、定期的にフルクロールを実行する必要があります。

- シークレットの Amazon リソースネーム (ARN) — Gmail Secrets Manager アカウントの認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "adminAccountEmailId": "service account email",
    "clientEmailId": "user account email",
    "privateKey": "private key"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Gmail コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Gmail data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - メッセージや添付ファイルを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Gmail データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

> **Note**
>
> Amazon Kendra Gmail データソースコネクタは API の制限によりカスタムインデックスフィールドの作成をサポートしていません。

詳細はこちら

Gmail Amazon Kendra データソースとの統合について詳しくは、以下をご覧ください。

- Amazon Kendraの Gmail コネクタを使用して、Google ワークスペース内のメール全体でインテリジェントな検索を実行します。

メモ

- 完全に削除された Gmail メッセージを更新する API がないため、FULL_CRAWL/[新規、変更、削除したコンテンツの同期] は以下のようになります。
  - Gmail から完全に削除されたメールはインデックスから削除されません。 Amazon Kendra
  - Gmail のメールラベルの変更は同期されません。

  Gmail Amazon Kendra のデータソースラベルの変更や完全に削除されたメールメッセージをインデックスに同期するには、定期的にフルクロールを実行する必要があります。
- Amazon Kendra Gmail データソースコネクタは API の制限によりカスタムインデックスフィールドの作成をサポートしていません。

Google ドライブ

Google Drive はクラウドベースのファイルストレージサービスです。 Amazon Kendra を使用して、Google Drive データソースの共有ドライブ、My Drives、共有フォルダに保存されているドキュメントのインデックスを作成できます。Google WorkSpace のドキュメントと、ドキュメントのタイプに記載されているドキュメントの両方にインデックスを作成できます。包含フィルターと除外フィルターを使用して、ファイル名、ファイルタイプ、ファイルパスでコンテンツにインデックスを作成することもできます。

Amazon Kendra コンソール、TemplateConfigurationAPI、または API Amazon Kendra を使用して Google ドライブのデータソースに接続できます。GoogleDriveConfiguration

Amazon Kendra には 2 つのバージョンの Google ドライブコネクタがあります。各バージョンでサポートされる機能は次のとおりです。

Google ドライブコネクタ V1.0/ API GoogleDriveConfiguration

- フィールドマッピング
- ユーザーアクセスコントロール

- 包含/除外フィルター

Google ドライブコネクタ V2.0/API TemplateConfiguration

- フィールドマッピング
- ユーザーアクセスコントロール
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

> **Note**
>
> Google ドライブコネクタ V1.0/Google DriveConfiguration API Support は 2023 年に終了する予定です。Google ドライブコネクタ V2.0/ API に移行するか、使用することをおすすめします。 TemplateConfiguration

Amazon Kendra Google Drive データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- Google Drive コネクタ V1.0
- Google Drive コネクタ V2.0

Google Drive コネクタ V1.0

Google Drive はクラウドベースのファイルストレージサービスです。 Amazon Kendra を使用して、Google ドライブデータソースの共有ドライブ、マイドライブ、Shared With Me フォルダに保存されているドキュメントやコメントのインデックスを作成できます。Google WorkSpace のドキュメントと、ドキュメントのタイプに記載されているドキュメントにインデックスを作成できます。包含フィルターと除外フィルターを使用して、ファイル名、ファイルタイプ、ファイルパスでコンテンツにインデックスを作成することもできます。

> **Note**
>
> Google ドライブコネクタ V1.0/Google DriveConfiguration API Support は 2023 年に終了する予定です。Google ドライブコネクタ V2.0/ API に移行するか、使用することをおすすめします。 TemplateConfiguration

Amazon Kendra Google Drive データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

- フィールドマッピング
- ユーザーアクセス制御
- 包含/除外フィルター

前提条件

Amazon Kendra を使用して Google ドライブのデータソースをインデックスに登録する前に、Google AWS ドライブとアカウントにこれらの変更を加えてください。

Google Drive で以下を確認してください。

- スーパー管理者ロールからアクセスを許可されているか、管理者権限を持つユーザーであるかのどちらかです。スーパー管理者ロールからアクセス許可を付与されている場合は、スーパー管理者ロールは必要ありません。
- [G Suite ドメイン全体の委任を有効にする] を有効にしたサービスアカウントを作成し、そのアカウントを使用してプライベートキーとして JSON キーを作成しました。
- ユーザーアカウント E メールとサービスアカウント E メールをコピーしました。接続したら、ユーザーアカウントのメールを管理者アカウントのメールアドレスとして、 Secrets Manager

サービスアカウントのメールをクライアントのメールアドレスとしてシークレットで入力します。Amazon Kendra

- 管理者 SDK API と Google Drive API がアカウントに追加されました。
- スーパー管理者ロールを使用して、以下のアクセス許可をサービスアカウントに追加しました (またはスーパー管理者ロールを持つユーザーに追加を依頼しました)。
  - https://www.googleapis.com/auth/drive.readonly
  - https://www.googleapis.com/auth/drive.metadata.readonly
  - https://www.googleapis.com/auth/admin.directory.user.readonly
  - https://www.googleapis.com/auth/admin.directory.group.readonly
- 各ドキュメントが Google Drive および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

に AWS アカウント、以下が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Google Drive の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Google Drive データソースをに接続するときに、IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Google Amazon Kendra ドライブのデータソースに接続するには、 Amazon Kendra データにアクセスできるように Google ドライブのデータソースに関する必要な詳細情報を入力する必要があります。Google ドライブをまだ設定していない場合は、 Amazon Kendra を参照してください前提条件。

Console

Google Amazon Kendra ドライブに接続するには

1. AWS 管理コンソールにログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースを追加する] ページで [Google Drive コネクタ V1.0] を選択し、[コネクタを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 — 索引用のドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [認証のタイプ] の場合 - [既存] と [新規] を選択します。既存のシークレットを使用する場合は、[シークレットを選択] を使用してシークレットを選択してください。

b. 新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットオプションが開きます。

- [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。シークレットネームには「AmazonKendra-Google Drive-」というプレフィックスが自動的に追加されます。

B. [管理者アカウントの E メール]、[クライアントの E メール]、[プライベートキー] の場合 - Google Drive アカウントから生成してダウンロードした認証情報の値を入力します。

C. [認証を保存] を選択します。

c. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

d. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [ユーザーアカウントを除外する] - インデックスから除外する Google Drive ユーザー。最大 100 件のユーザーアカウントを追加できます。

b. [共有ドライブを除外する] - インデックスから除外する Google Drive の共有ドライブ。最大 100 件の共有ドライブを追加できます。

c. [ファイルタイプのドライブを除外する] - インデックスから除外する Google Drive のファイルタイプ。MIME タイプの選択を編集することもできます。

d. [追加設定] の場合: 特定のコンテンツを含めるか除外する正規表現パターンを指定します。最大 100 のパターンを追加できます。

e. [頻度] - Amazon Kendra がデータソースと同期する頻度。

f. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. GoogleDrive フィールド名とその他の推奨フィールドマッピング用- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマップするフィールドを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Google Amazon Kendra ドライブに接続するには

GoogleDriveConfigurationAPI を使用して以下を指定する必要があります。

- シークレットの Amazon リソースネーム (ARN) — Google Secrets Manager ドライブアカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "clientAccount": "service account email",
    "adminAccount": "user account email"",
    "privateKey": "private key"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットへのアクセス権限をロールに付与し、Google Drive コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Google Drive data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- 包含フィルターと除外フィルター - デフォルトでは、 Amazon Kendra は Google Drive 内のすべてのドキュメントのインデックスを作成します。共有ドライブ、ユーザーアカウント、ドキュメント MIME タイプ、ファイルに特定のコンテンツを含めるか除外するかを指定できます。ユーザーアカウントを除外すると、そのアカウントが所有する My Drive 内のファイルにはインデックスが作成されません。ユーザーと共有されているファイルは、ファイルの所有者も除外されない限り、インデックスが作成されます。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング - 選択すると、Google Drive データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

  > **Note**
  >
  > ドキュメントを検索するには、 Amazon Kendra ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

詳細はこちら

Google Drive Amazon Kendra データソースとの統合について詳しくは、以下をご覧ください。

- Amazon Kendra Google ドライブコネクタを使い始める

Google Drive コネクタ V2.0

Google Drive はクラウドベースのファイルストレージサービスです。を使用すると Amazon Kendra 、Google ドライブデータソースの共有ドライブ、マイドライブ、共有フォルダーに保存されているドキュメントやコメントにインデックスを付けることができます。Google WorkSpace のドキュメントと、ドキュメントのタイプに記載されているドキュメントにインデックスを作成できます。包含フィルターと除外フィルターを使用して、ファイル名、ファイルタイプ、ファイルパスでコンテンツにインデックスを作成することもできます。

> **Note**
>
> Google ドライブコネクタ V1.0/Google DriveConfiguration API Support は 2023 年に終了する予定です。Google ドライブコネクタ V2.0/ API に移行するか、使用することをおすすめします。 TemplateConfiguration

Amazon Kendra Google Drive データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーアクセス制御
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

を使用して Google Amazon Kendra ドライブのデータソースをインデックスに登録する前に、Google AWS ドライブとアカウントでこれらの変更を行ってください。

Google Drive で以下を確認してください。

- スーパー管理者ロールからアクセスを許可されているか、管理者権限を持つユーザーであるかのどちらかです。スーパー管理者ロールからアクセス許可を付与されている場合は、スーパー管理者ロールは必要ありません。
- 管理者アカウントの E メール、クライアントの E メール (サービスアカウントの E メール)、シークレットキーを含む Google Drive サービスアカウントの接続認証情報を設定しました。サービスアカウントキーの作成と削除については、Google Cloud のドキュメントを参照してください。
- server-to-server認証時に G Suite ドメイン全体の委任を有効化して Google Cloud サービスアカウント (ユーザー ID を引き継ぐ権限が委任されたアカウント) を作成し、そのアカウントを使用して JSON プライベートキーを生成しました。

> **Note**
>
> シークレットキーは、サービスアカウントの作成後に生成する必要があります。

- ユーザーアカウントに管理 SDK API と Google Drive API が追加されました。
- オプション: クライアント ID、クライアントシークレット、更新トークンを含む Google Drive OAuth 2.0 接続認証情報を特定のユーザーの接続認証情報として設定しました。これは個々のアカウントデータをクロールするのに必要です。OAuth 2.0 を使用して API にアクセスする方法については、Google のドキュメントを参照してください。
- スーパー管理者ロールを使用して、以下の OAuth スコープをサービスアカウントに追加しました (またはスーパー管理者ロールを持つユーザーに追加を依頼しました)。これらの API スコープ

は、Google Workspace ドメイン内のすべてのドキュメントとアクセスコントロール (ACL) 情報をクロールするために必要です。

- https://www.googleapis.com/auth/drive.readonly - Google Drive のファイルをすべて表示してダウンロードします
- https://www.googleapis.com/auth/drive.metadata.readonly - Google Drive 内のファイルのメタデータを表示します。
- https://www.googleapis.com/auth/admin.directory.group.readonly - スコープは、グループ、グループエイリアス、およびメンバー情報のみを取得するためのものです。これは ID クローラーに必要です。 Amazon Kendra
- https://www.googleapis.com/auth/admin.directory.user.readonly - スコープは、ユーザーまたはユーザーエイリアスのみを取得するためのものです。 Amazon Kendra ID クローラーにユーザーを一覧表示したり、ACL を設定したりするのに必要です。
- https://www.googleapis.com/auth/cloud-platform - スコープは、大きな Google Drive ファイルのコンテンツを取得するためのアクセストークンを生成するためのものです。
- https://www.googleapis.com/auth/forms.body.readonly - スコープは、Google フォームからデータを取得するためのものです。

Forms API をサポートするには、以下のスコープを追加してください。

- https://www.googleapis.com/auth/forms.body.readonly

- 各ドキュメントが Google Drive および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

あなたには AWS アカウント、次のものがあることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Google Drive の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Google Drive データソースをに接続するときに、IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Google Amazon Kendra ドライブのデータソースに接続するには、 Amazon Kendra データにアクセスできるように Google ドライブのデータソースに関する必要な詳細情報を入力する必要があります。Google ドライブをまだ設定していない場合は、 Amazon Kendra を参照してください前提条件。

Console

Google Amazon Kendra ドライブに接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Google ドライブコネクタ] を選択し、[コネクタの追加] を選択します。

5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-インデックス用のドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. 承認 — ACL があり、それをアクセス制御に使用したい場合は、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

   b. [認証] 用 - ユースケースに基づいて [Google サービスアカウント] と [OAuth 2.0 認証] のいずれかを選択します。

   c. AWS Secrets Manager secret — Google Drive Secrets Manager の認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      i. Google サービスアカウントを選択した場合は、シークレットの名前、サービスアカウント設定の admin ユーザーまたは「サービスアカウントユーザー」のメール ID (admin email)、サービスアカウントのメール ID (クライアントメール)、およびサービスアカウントで作成したプライベートキーを入力します。

         シークレットを保存して追加します。

      ii. OAuth 2.0 認証を選択した場合は、OAuth アカウントで作成したシークレット、クライアント ID、クライアントシークレット、更新トークンの名前を入力します。

         シークレットを保存して追加します。

d. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

e. (Google サービスアカウント認証ユーザーのみ)

ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

f. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

g. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. コンテンツの同期 — クロールするオプションまたはコンテンツを選択します。マイドライブ (個人フォルダ)、共有ドライブ (共有フォルダ)、またはその両方をクロールできます。ファイルコメントを含めることもできます。

b. [追加設定-オプション] では以下のオプション情報を入力することもできます。

i. 対象読者-クロールする文書に特定の対象読者を追加します。

ii. 最大ファイルサイズ-クロールするファイルの最大サイズを MB 単位で設定します。

iii. ユーザーメール — 含めたり除外したりするユーザーメールを追加します。

iv. 共有ドライブ — 含めたり除外したりする共有ドライブ名を追加します。

v. MIME タイプ — 含めたり除外したりする MIME タイプを追加します。

vi. エンティティ正規表現パターン — サポートされているすべてのエンティティの特定の添付ファイルを含めたり除外したりする正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

> **Important**
>
> Google Drive API は、完全に削除されたファイルからのコメントの取得をサポートしていません。ゴミ箱に捨てられたファイルからのコメントは取得可能です。ファイルがゴミ箱に捨てられると、コネクタはインデックスからコメントを削除します。 Amazon Kendra

d. [同期実行スケジュール] の [頻度] で、データソースのコンテンツを同期してインデックスを更新する頻度を選択します。

e. [同期実行履歴] で、 Amazon S3 データソースを同期するときに自動生成されたレポートをに保存することを選択します。これは、データソースを同期する際の問題を追跡するのに便利です。

f. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. ファイル用 — Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマップしたいものを選択します。

> **Note**
>
> Google Drive API はカスタムフィールドの作成をサポートしていません。Google Drive コネクタではカスタムフィールドマッピングは使用できません。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Google Amazon Kendra ドライブに接続するには

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON GOOGLEDRIVEV2 スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- 認証タイプ — サービスアカウント認証と OAuth 2.0 認証のどちらを使用するかを指定します。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

> **Important**
>
> Google Drive API は、完全に削除されたファイルからのコメントの取得をサポートしていません。ゴミ箱に捨てられたファイルからのコメントは取得可能です。ファイルが破棄されると、コネクタはインデックスからコメントを削除します。 Amazon Kendra

- シークレット Amazon リソースネーム (ARN) — Google Secrets Manager ドライブアカウントで作成した認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Google サービスアカウント認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "clientEmail": "user account email",
    "adminAccountEmail": "service account email",
    "privateKey": "private key"
}
```

OAuth 2.0 認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "clientID": "OAuth client ID",
    "clientSecret": "client secret",
    "refreshToken": "refresh token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Google Drive コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Google Drive data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- マイドライブ、共有ドライブ、コメント — これらの種類のコンテンツをクロールするかどうかを指定できます。
- 包含フィルターと除外フィルター-特定のユーザーアカウント、共有ドライブ、MIME タイプを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- アクセス制御リスト (ACL) — ACL があり、それをアクセス制御に使用したい場合に、ドキュメントの ACL 情報をクロールするかどうかを指定します。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- フィールドマッピング - 選択すると、Google Drive データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「Google Drive template schema」を参照してください。

メモ

- Google Drive UI はカスタムフィールドの作成をサポートしていないため、Google Drive コネクタではカスタムフィールドマッピングを使用できません。
- Google Drive API は、完全に削除されたファイルからのコメントの取得をサポートしていません。ただし、ゴミ箱に捨てられたファイルのコメントは取得できます。ファイルがゴミ箱に捨てられると、 Amazon Kendra Amazon Kendra コネクタはインデックスからコメントを削除します。
- Google Drive API は .docx ファイルにあるコメントを返しません。

IBM DB2

IBM DB2 は、IBM によって開発されたリレーショナルデータベース管理システムです。IBM DB2 ユーザーであれば、 Amazon Kendra を使用して IBM DB2 データソースのインデックスを作成できます。 Amazon Kendra IBM DB2データ・ソース・コネクターは DB2 11.5.7 をサポートします。

Amazon Kendra コンソールと API Amazon Kendra IBM DB2 を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra IBM DB2データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順

- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra IBM DB2を使用してデータソースのインデックスを作成する前に、IBM DB2 AWS とアカウントでこれらの変更を行ってください。

IBM DB2 で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが IBM DB2 および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- IBM DB2 の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager IBM DB2データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、IBM DB2IBM DB2 Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、IBM DB2 Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには IBM DB2

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [IBM DB2コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [ソース] には、次の情報を入力します。
   b. [ホスト] - データベースのホスト名を入力します。
   c. [ポート] - データベースのポートを入力します。
   d. [インスタンス] - データベースインスタンスを入力します。
   e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。
   f. [認証] には、次の情報を入力します。
      - AWS Secrets Manager secret — IBM DB2 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraIBM DB2-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra IBM DB2

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを db2 として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。IBM DB2シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。IBM DB2 Amazon Kendra詳細については、「IAM roles for IBM DB2 data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、IBM DB2 データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「IBM DB2 テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Jira

Jira はソフトウェア開発、製品管理、バグ追跡のためのプロジェクト管理ツールです。を使用して Amazon Kendra Jira プロジェクト、課題、コメント、添付ファイル、作業ログ、ステータスのインデックスを作成できます。

Amazon Kendra 現在のところ Jira Cloud のみをサポートしています。

Jira データソースには、Amazon Kendra コンソールまたは API Amazon Kendra を使用して接続できます。JiraConfiguration それぞれがサポートしている機能のリストについては、「サポートされている機能」を参照してください。

Amazon Kendra Jira データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順

- 詳細はこちら

サポートされている機能

Amazon Kendra Jira データソースコネクタは以下の機能をサポートしています。

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Jira データソースのインデックスを作成する前に、Jira とアカウントでこれらの変更を行ってください。 AWS

Jira で以下を確認してください。

- Jira ID (ユーザー名または E メール) と Jira 認証情報 (Jira API トークン) を含む Jira API トークン認証情報を作成しました。API トークンの管理に関する Atlassian のドキュメントを参照してください。
- Jira アカウント設定の Jira アカウント URL を記録しました。例えば、*https://company.atlassian.net/*。
- 各ドキュメントが Jira および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

に AWS アカウント、以下の内容が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Jira の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Jira IAM Secrets Manager データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Jira Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Jira データソースの必要な詳細情報を入力する必要があります。Jira をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Jira Amazon Kendra に接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Jira コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. 既定の言語で-インデックスの対象となるドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [Jira アカウント URL] - Jira アカウント URL を入力します。例えば、*https://company.atlassian.net/*。

   b. AWS Secrets Manager シークレット — Jira Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

         A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Jira-' がシークレット名に自動的に追加されます。

         B. [Jira ID] の場合 - Jira のユーザー名または E メールアドレスを入力します。

         C. [パスワード/トークン] の場合 - Jira アカウントから作成した Jira API トークンを入力します。

      ii. [保存] を選択します。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. IAM role — 既存のロールを選択するか、 IAM 新しいロールを作成して、 IAM リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [インデックスを作成する Jira プロジェクトを選択] - クロールする Jira エンティティまたはコンテンツタイプを選択します。

b. [ステータス]、[追加要素]、[問題のタイプ] - コンテンツを選択してインデックスの範囲を絞り込みます。

c. [変更ログ] - 選択すると、すべてのファイルを同期する代わりにインデックスを更新できます。

d. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

e. [同期実行スケジュール] の [頻度] で、Amazon Kendra がデータソースと同期する頻度を選択します。

f. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. [プロジェクト]、[問題]、[コメント]、[添付ファイル]、[作業ログ] の場合 - インデックスにマッピングする Amazon Kendra 生成されたデフォルトのデータソースフィールドから選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Jira Amazon Kendra に接続するには

JiraConfiguration API を使用して以下を指定する必要があります。

- データソース URL - Jira アカウントの URL を指定します。例えば、*company.atlassian.net*。
- シークレット Amazon リソースネーム (ARN) — Jira Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "jiraId": "Jira user name or email",
    "jiraCredential": "Jira API token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに付与し、Jira コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Jira data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - データソース設定の一部として `VpcConfiguration` を指定します。「VPC を使用するための Amazon Kendra の設定」を参照してください。

- 変更ログ — Jira データソース変更ログメカニズムを使用して、 Amazon Kendra インデックス内のドキュメントを更新する必要があるかどうかを判断すべきかどうか。

  > **Note**
  >
  > Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、変更ログを処理するよりも Jira Amazon Kendra データソース内のドキュメントをスキャンするほうが時間がかからない場合があります。Jira データソースをインデックスに初めて同期する場合は、すべてのドキュメントがスキャンされます。

- 包含フィルターと除外フィルター — 特定のファイルを含めるか除外するかを指定できます。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- コメント、添付ファイル、および作業ログ — 課題の特定のコメント、添付ファイル、および作業ログをクロールするかどうかを指定できます。
- プロジェクト、課題、ステータス — 特定のプロジェクト ID、課題タイプ、ステータスをクロールするかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Jira データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

詳細はこちら

Jira Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Jira Cloud コネクタを使用して Jira プロジェクトをインテリジェントに検索します。 Amazon Kendra

Microsoft Exchange

Microsoft Exchange は、メッセージング、会議、ファイル共有のためのエンタープライズコラボレーションツールです。Microsoft Exchange ユーザーの場合は、 Amazon Kendra を使用して Microsoft Exchange データソースのインデックスを作成できます。

Amazon Kendra コンソールと TemplateConfigurationAPI を使用して Microsoft Exchange Amazon Kendra データソースに接続できます。

Amazon Kendra Microsoft Exchange データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Microsoft Exchange データソースのインデックスを作成する前に、Microsoft Exchange AWS とアカウントでこれらの変更を行ってください。

Microsoft Exchange で以下を確認してください。

- Office 365 で Microsoft Exchange アカウントを作成しました。
- Microsoft 365 のテナント ID を記録しました。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- Azure ポータルで OAuth アプリケーションを作成し、クライアント ID、クライアントシークレット、またはクライアント認証情報を記録しました。詳細については、「Microsoft チュートリアル」と「登録済みアプリの例」を参照してください。

> **Note**
>
> Azure Portal でアプリを作成または登録すると、シークレット ID は実際のシークレット値を表します。シークレットとアプリを作成したら、すぐに実際のシークレット値を書き留めるか保存する必要があります。シークレットにアクセスするには、Azure Portal でアプリケーションの名前を選択し、証明書とシークレットのメニューオプションに移動します。
> クライアント ID にアクセスするには、Azure Portal でアプリケーションの名前を選択し、概要ページに移動します。アプリケーション (クライアント) ID はクライアント ID です。

- コネクタアプリケーションに次のアクセス許可を追加しました。

Microsoft Graph	Office 365 Exchange Online
• Mail.Read (アプリケーション) • メール。 ReadBasic (アプリケーション) • メール。 ReadBasic.All (アプリケーション) • Calendars.Read (アプリケーション) • User.Read.All (アプリケーション) • Contacts.Read (アプリケーション) • Notes.Read.All (アプリケーション) • Directory.Read.All (アプリケーション)	full_access_as_app (アプリケーション)

Microsoft Graph	Office 365 Exchange Online
• ニュース。 AccessAsUser.すべて (委任)	

- 各ドキュメントが Microsoft Exchange および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Microsoft Exchange の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールまたはシークレットがない場合は、Microsoft Exchange データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Microsoft Exchange Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Microsoft Exchange データソースに関する必要な詳細情報を入力する必要があります。Microsoft Exchange をまだ設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Microsoft Exchange Amazon Kendra に接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Microsoft Exchange コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. [既定の言語]-索引の対象となるドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] - Microsoft 365 テナント ID を入力します。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。

b. 承認 — ACL があり、それをアクセス制御に使用したい場合は、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

c. AWS Secrets Manager シークレット-既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成して Microsoft Exchange 認証資格情報を保存します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

   i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

      A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Microsoft エクスチェンジ

      B. クライアント ID の場合 — クライアント ID を入力します。

      C. クライアントシークレットの場合 — Azure ポータルの Microsoft Exchange アカウントで作成した認証資格情報の値を入力します。

   ii. [保存] を選択します。

d. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

e. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [コンテンツを同期] - 同期するコンテンツを選択します。

b. [追加設定] - オプションで、すべてのドキュメントを同期する代わりに、以下のコンテンツのインデックスを作成することができます。

- [エンティティタイプ] - 同期するエンティティを選択します。[カレンダー] と [連絡先] のいずれかを選択できます。OneNotes
- [カレンダークロール] - カレンダーの同期の開始日と終了日を入力します。
- [E メールを含める] - [メール送信者] ドメインと [メール受信者] ドメイン、およびインデックスに含めるまたは除外する [件名] 行を入力します。
- [ドメインの正規表現] - 特定の E メールドメインをインデックスに含めるまたは除外するパターンを追加します。
- [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択できます。

i. 完全同期を選択すると、 Amazon Kendra は、以前の同期ステータスに関係なく、すべてのエンティティのすべてのコンテンツを同期します。

ii. 新規または変更されたコンテンツの同期を選択すると、 Amazon Kendra は、新しいコンテンツまたは変更されたコンテンツのみが同期されます。

iii. 新規、変更、削除済みコンテンツの同期を選択すると、新規、変更、 Amazon Kendra 削除したコンテンツのみが同期されます。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. デフォルトデータソースフィールド- Amazon Kendra 生成されたデフォルトデータソースフィールドの中から、インデックスにマップしたいものを選択します。

> **Note**
>
> Amazon Kendra Microsoft Exchange データソースコネクタは、カスタムフィールドマッピングをサポートしていません。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Microsoft Exchange Amazon Kendra に接続するには

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON MSEXCHANGE スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- テナント ID - テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレットAmazon リソースネーム (ARN) — Microsoft Exchange Secrets Manager アカウントの認証認証情報を含むシークレットのAmazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "clientId": "client ID",
    "clientSecret": "client secret"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Microsoft Exchange Amazon Kendraコネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。詳細については、「IAM roles for Microsoft Exchange data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 特定のページやアセットを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Microsoft Exchange データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名

にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

詳細はこちら

Microsoft Exchange Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendraの Exchange コネクタを使用して Microsoft Exchange コンテンツのインデックスを作成する

Microsoft OneDrive

Microsoft OneDrive は、コンテンツの保存、共有、ホストに使用できるクラウドベースのストレージサービスです。 Amazon Kendra OneDrive を使用してデータソースのインデックスを作成できます。

Amazon Kendra コンソールと OneDriveConfigurationAPI Amazon Kendra OneDrive を使用してデータソースに接続できます。

Amazon Kendra には 2 OneDrive つのバージョンのコネクタがあります。各バージョンでサポートされる機能は次のとおりです。

Microsoft OneDrive コネクター V1.0/API OneDriveConfiguration

- フィールドマッピング
- 包含/除外フィルター

Microsoft OneDrive コネクター V2.0/API TemplateConfiguration

- ユーザーコンテキストフィルタリング
- ユーザー ID クローラー
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

> **Note**
>
> OneDrive コネクター OneDriveConfiguration V1.0/API Support は 2023 年 6 月までに終了する予定です。 OneDrive コネクタ V2.0/ API の使用をお勧めします。 TemplateConfiguration

Amazon Kendra OneDrive データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- Microsoft OneDrive コネクター V1.0
- Microsoft OneDrive コネクター V2.0
- 詳細はこちら

Microsoft OneDrive コネクター V1.0

Microsoft OneDrive は、コンテンツの保存、共有、ホストに使用できるクラウドベースのストレージサービスです。 Amazon Kendra を使用して Microsoft OneDrive データソースのインデックスを作成できます。

> **Note**
>
> OneDrive コネクタ V1.0/Microsoft OneDrive API Support は 2023 年 6 月までに終了する予定です。 OneDrive コネクタ V2.0/ API の使用をお勧めします。 TemplateConfiguration

Amazon Kendra OneDrive データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順

サポートされている機能

- フィールドマッピング

- 包含/除外フィルター

前提条件

Amazon Kendra OneDrive を使用してデータソースのインデックスを作成する前に、 OneDrive AWS およびアカウントで以下の変更を行ってください。

Azure Active Directory (AD) で、以下を確認します。

- Azure Active Directory (AD) アプリケーションを作成しました。
- AD アプリケーション ID を使用して AD サイト上のアプリケーションのシークレットキーを登録しました。シークレットキーには、アプリケーション ID とシークレットキーが含まれている必要があります。
- 組織の AD ドメインをコピーしました。
- Microsoft Graph オプションの AD アプリケーションに次のアプリケーション権限を追加しました。
  - すべてのサイトコレクション内のファイルを読み取る (File.Read.All)
  - すべてのユーザーの完全なプロフィールを読み取る (User.Read.All)
  - ディレクトリデータを読み取る (Directory.Read.All)
  - すべてのグループを読み取る (Group.Read.All)
  - すべてのサイトコレクションの項目を読み取る (Site.Read.All)
- インデックスを作成する必要があるドキュメントを持つユーザーのリストをコピーしました。ユーザー名のリストを指定するか、 Amazon S3に保存されているファイルにユーザー名を指定できます。データソースを作成すると、次のことが行なえます。
  - ユーザーのリストを変更します。
  - Amazon S3 ユーザーのリストからバケットに保存されているリストに変更します。
  - Amazon S3 ユーザーリストのバケットロケーションを変更します。バケットの場所を変更する場合は、 IAM データソースのロールも更新して、バケットにアクセスできるようにする必要があります。

> **Note**
>
> Amazon S3 ユーザー名のリストをバケットに保存する場合、 IAM データソースのポリシーは、バケットへのアクセスと、バケットの暗号化に使用されたキー (ある場合) へのアクセスを提供する必要があります。

- 各ドキュメントが、 OneDrive 同じインデックスに使用する予定の他のデータソース内およびデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- OneDrive AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager OneDrive データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 OneDrive OneDrive Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、OneDrive Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには OneDrive

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [OneDrive コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. OneDrive テナント ID — OneDrive テナント ID をプロトコルなしで入力します。
   b. [認証のタイプ] の場合 - [新規] と [既存] を選択します。
   c. i. [既存] を選択した場合は、[シークレットを選択] で既存のシークレットを選択します。
      ii. [新規] を選択した場合は、[新規の AWS Secrets Manager シークレット] セクションに次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraOneDrive-' がシークレット名に自動的に追加されます。

B. アプリケーション ID とアプリケーションパスワードの場合: OneDrive アカウントの認証資格値を入力し、[認証を保存] を選択します。

d. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. ユースケースに基づいて、[リストファイル] と [名前リスト] のどちらかを選択します。

i. [リストファイル] を選択した場合は、次の情報を入力します。

- [場所を選択する] - Amazon S3 バケットへのパスを入力します。

  ユーザーリストファイルの追加 Amazon S3 — 選択すると、ユーザーリストファイルがバケットに追加されます。 Amazon S3

  [ユーザーローカルグループマッピング] - ローカルグループマッピングを使用してコンテンツをフィルタリングします。

ii. [名前リスト] を選択した場合は、次の情報を入力します。

- ユーザー名 - インデックスを作成するユーザードライブを最大 10 件入力します。10 件以上のユーザーを追加するには、名前を含むファイルを作成します。

  [別のものを追加] - さらにユーザーを追加します。

  [ユーザーローカルグループマッピング] - ローカルグループマッピングを使用してコンテンツをフィルタリングします。

b. [追加設定] の場合 - 特定のファイルを含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. デフォルトのデータソースフィールドとその他の推奨フィールドマッピングの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングするフィールドを選択します。

b. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra OneDrive

OneDriveConfigurationAPI を使用して以下を指定する必要があります。

- テナント ID - 組織の Azure Active Directory ドメインを指定します。
- OneDrive Users — ドキュメントのインデックスを作成するユーザーアカウントのリストを指定します。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 OneDrive シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "OAuth client ID",
    "password": "client secret"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 OneDrive Amazon Kendra詳細については、「IAM OneDrive データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- 包含フィルターと除外フィルター - 特定のドキュメントを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング — OneDrive Amazon Kendra データソースフィールドをインデックスフィールドにマップすることを選択します。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

Microsoft OneDrive コネクター V2.0

Microsoft OneDrive は、コンテンツの保存、共有、ホストに使用できるクラウドベースのストレージサービスです。 Amazon Kendra OneDriveを使用してデータソースのインデックスを作成できます。

Amazon Kendra コンソールと OneDriveConfigurationAPI Amazon Kendra OneDrive を使用してデータソースに接続できます。

> **Note**
>
> OneDrive コネクタ OneDriveConfiguration V1.0/API Support は 2023 年 6 月までに終了する予定です。 OneDrive コネクタ V2.0/ API の使用をお勧めします。 TemplateConfiguration バージョン 2.0 では、ACL と ID クローラー機能が追加されています。

Amazon Kendra OneDrive データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

Amazon Kendra OneDrive データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザー ID クローラー
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra OneDrive を使用してデータソースのインデックスを作成する前に、 OneDrive AWS とアカウントでこれらの変更を行ってください。

で OneDrive、次のものが揃っていることを確認してください。

- Office 365 OneDrive でアカウントを作成しました。
- Microsoft 365 のテナント ID を記録しました。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- Azure ポータルで OAuth アプリケーションを作成し、クライアント ID、クライアントシークレット、またはクライアント認証情報を記録しました。詳細については、「Microsoft チュートリアル」と「登録済みアプリの例」を参照してください。

  > **Note**
  >
  > Azure Portal でアプリを作成または登録すると、シークレット ID は実際のシークレット値を表します。シークレットとアプリを作成したら、すぐに実際のシークレット値を書き留めるか保存する必要があります。シークレットにアクセスするには、Azure Portal でアプリケーションの名前を選択し、証明書とシークレットのメニューオプションに移動します。
  > クライアント ID にアクセスするには、Azure Portal でアプリケーションの名前を選択し、概要ページに移動します。アプリケーション (クライアント) ID はクライアント ID です。

- AD アプリケーション ID を使用して AD サイト上のアプリケーションのシークレットキーを登録しました。シークレットキーには、アプリケーション ID とシークレットキーが含まれている必要があります。
- 組織の AD ドメインをコピーしました。
- Microsoft Graph オプションで、AD アプリケーションに次アクセス許可を追加しました。
  - すべてのサイトコレクション内のファイルを読み取る (File.Read.All)
  - すべてのユーザーの完全なプロフィールを読み取る (User.Read.All)
  - すべてのグループを読み取る (Group.Read.All)
  - すべてのメモを読む (Notes.Read.All)
- インデックスを作成する必要があるドキュメントを持つユーザーのリストをコピーしました。ユーザー名のリストを指定するか、 Amazon S3に保存されているファイルにユーザー名を指定できます。データソースを作成すると、次のことが行なえます。

- ユーザーのリストを変更します。
- Amazon S3 ユーザーのリストからバケットに保存されているリストに変更します。
- Amazon S3 ユーザーリストのバケットロケーションを変更します。バケットの場所を変更する場合は、 IAM データソースのロールも更新して、バケットにアクセスできるようにする必要があります。

> **Note**
>
> Amazon S3 ユーザー名のリストをバケットに保存する場合、 IAM データソースのポリシーは、バケットへのアクセスと、バケットの暗号化に使用されたキー (ある場合) へのアクセスを提供する必要があります。
> OneDrive コネクタは OneDrive ユーザープロパティにある [連絡先情報からの電子メール] を使用します。データをクロールするユーザーの [連絡先情報] ページの E メールフィールドが設定されていることを確認します。新規ユーザーの場合は、このフィールドは空白になる場合があります。

AWS アカウントには、以下の内容が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。
- OneDrive AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager OneDrive データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 OneDrive OneDrive Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。をまだ設定していない場合は Amazon Kendra、 OneDrive を参照してください前提条件。

Console

Amazon Kendra に接続するには OneDrive

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [OneDrive コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. OneDrive テナント ID — OneDrive テナント ID をプロトコルなしで入力します。
   b. 承認 — ACL があり、それをアクセス制御に使用したい場合は、ドキュメントのアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
   c. [認証] - [新規] または [既存] を選択します。

d. i. [既存] を選択した場合は、[シークレットを選択] で既存のシークレットを選択します。

ii. [新規] を選択した場合は、[新規の AWS Secrets Manager シークレット] セクションに次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraOneDrive-' はシークレット名に自動的に追加されます。

B. [クライアント ID] および [クライアントシークレット] の場合 - クライアント ID とクライアントシークレットを入力し、[認証を保存] を選択します。

e. [VPC とセキュリティグループの設定 - オプション] で、[仮想プライベートクラウド (VPC)] では VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

f. ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

g. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

h. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

8. a. Sync scope の場合- OneDrive インデックスを作成するユーザーのデータを選択します。最大 10 のユーザーを手動で追加できます。

b. [追加設定] の場合 - 特定のコンテンツを含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。[完全同期] は、前回の同期ステータスに関係なく、すべてのコンテンツのインデックスを作成します。[新規、変更済み、または削除されたドキュメントを同期] は、新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] で、 Amazon Kendra データソースと同期する頻度を選択します。

e. [次へ] を選択します。

9. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. デフォルトのデータソースフィールドとその他の推奨フィールドマッピングの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングするフィールドを選択します。

b. [次へ] を選択します。

10. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra OneDrive

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON `ONEDRIVEV2` スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI `TEMPLATE` を呼び出すときと同じようにデータソースを指定します。
- テナント ID - Microsoft 365 テナント ID を指定します。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - `FORCED_FULL_CRAWL` は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。

  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 OneDrive

  OAuth 2.0 認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "clientId": "client ID",
    "clientSecret": "client secret"
}
```

  > **Note**
  >
  > 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。 OneDrive Amazon Kendra詳細については、「IAM OneDrive データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター — 特定のファイル、 OneNote セクション、ページを含めるか除外するかを指定できます。 OneNote

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- フィールドマッピング — コネクタの組み込みインデックスフィールドまたは共通インデックスフィールドのみをマップできます。 Amazon Kendra OneDrive API の制限により、OneDrive コネクタではカスタムフィールドマッピングを使用できません。詳細については、データソースフィールドのマッピングを参照してください。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Microsoft OneDrive テンプレートスキーマ」を参照してください。

詳細はこちら

Amazon Kendra OneDrive データソースとの統合について詳しくは、以下を参照してください。

- の更新されたMicrosoft OneDrive コネクタ (V2) を発表します。 Amazon Kendra

Microsoft SharePoint

SharePoint は、Web コンテンツのカスタマイズや、ページ、サイト、ドキュメントライブラリ、リストの作成に使用できる共同的な Web サイト構築サービスです。 Amazon Kendra SharePoint を使用してデータソースのインデックスを作成できます。

Amazon Kendra 現在、 SharePoint SharePointオンラインとサーバー (バージョン 2013、2016、2019、およびサブスクリプションエディション) をサポートしています。

Amazon Kendra コンソール、API、または TemplateConfigurationAPI Amazon Kendra SharePoint のいずれかを使用してデータソースに接続できます。SharePointConfiguration

Amazon Kendra には 2 SharePoint つのバージョンのコネクタがあります。各バージョンでサポートされる機能は次のとおりです。

SharePoint コネクター V1.0/ API SharePointConfiguration

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

SharePoint コネクター V2.0/ API TemplateConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザー ID クローリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

> **Note**
>
> SharePoint コネクター SharePointConfiguration V1.0/API Support は 2023 年に終了する予定です。 SharePoint コネクタ V2.0/ API への移行または使用をお勧めします。 TemplateConfiguration

Amazon Kendra SharePoint データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- SharePoint コネクタ V1.0
- SharePoint コネクター V2.0

SharePoint コネクタ V1.0

SharePoint は、Web コンテンツのカスタマイズや、ページ、サイト、ドキュメントライブラリ、リストの作成に使用できる共同ウェブサイト構築サービスです。 SharePoint ユーザーであれば、Amazon Kendra SharePoint を使用してデータソースのインデックスを作成できます。

> **Note**
>
> SharePoint コネクター SharePointConfiguration V1.0/API Support は 2023 年に終了する予定です。 SharePoint コネクタ V2.0/ API への移行または使用をお勧めします。 TemplateConfiguration

Amazon Kendra SharePoint データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra SharePoint を使用してデータソースのインデックスを作成する前に、 SharePoint AWS およびアカウントで以下の変更を行ってください。

で SharePoint、次のものが揃っていることを確認してください。

- SharePoint インデックスを作成するサイトの URL を書き留めました。
- SharePoint オンラインの場合:
  - サイト管理者権限を持つユーザー名とパスワードを含む基本認証資格情報を記録しました。
  - オプション: ユーザー名、パスワード、クライアント ID、クライアントシークレットを含む OAuth 2.0 認証情報を生成しました。
  - 管理者ユーザーを使用して Azure Portal の [セキュリティデフォルト] を無効にしました。Azure Portal でのセキュリティのデフォルト設定の管理の詳細については、セキュリティのデフォルトを有効または無効にする方法に関する Microsoft のドキュメントを参照してください。
- SharePoint サーバー用:
  - SharePoint サーバーのドメイン名 (アクティブディレクトリ内の NetBIOS 名) を書き留めておきます。これを、 SharePoint 基本認証のユーザー名およびパスワードと共に SharePoint Server に接続します Amazon Kendra。

> **Note**
>
> SharePoint Server を使用していて、ユーザーコンテキストに基づいてフィルタリングできるようにアクセスコントロールリスト (ACL) を電子メール形式に変換する必要がある場合は、LDAP サーバー URL と LDAP 検索ベースを指定します。または、ディレクトリドメインの上書きを使用することもできます。LDAP サーバーの URL は、完全なドメイン名およびポート番号 (例えば、ldap://example.com:389) です。LDAP 検索ベースは、ドメインコントローラの「example」と「com」です。ディレクトリドメインの上書きでは、LDAP

サーバーの URL と LDAP 検索ベースを使用する代わりに、E メールドメインを使用できます。例えば、「username@example.com」の E メールドメインは「example.com」です。この上書きは、ドメインの検証について心配がなく、単に E メールドメインを使用する場合に使用できます。

- SharePoint アカウントに次の権限を追加しました。

  SharePoint リスト用

  - アイテムを開く - サーバー側のファイルハンドラを使用してドキュメントのソースを表示します。
  - アプリケーションページの表示 - フォーム、ビュー、およびアプリケーションページを表示します。リストを一覧表示します。
  - アイテムの表示 - リスト内のアイテムとドキュメントライブラリ内のドキュメントを表示します。
  - バージョンの表示 - リスト項目またはドキュメントの過去のバージョンを表示します。

  SharePoint Web サイト用

  - ディレクトリを参照-Designer と Web DAV インターフェイスを使用して Web サイト内のファイルとフォルダーを列挙します。 SharePoint
  - ユーザー情報の参照 - ウェブサイトのユーザーに関する情報を表示します。
  - アクセス許可の一覧表示 - ウェブサイト、リスト、フォルダ、ドキュメント、またはリスト項目に対する許可を一覧表示します。
  - 開く - ウェブサイト、リスト、またはフォルダを開き、コンテナ内のアイテムにアクセスします。
  - クライアント統合機能の使用-SOAP、WebDAV、クライアントオブジェクトモデル、または SharePoint Designer インターフェイスを使用して Web サイトにアクセスします。
  - リモートインターフェイスの使用 - クライアントアプリケーションを起動する機能を使用します。
  - ページの表示 - ウェブサイトのページを表示します。
- 各ドキュメントが、 SharePoint 同じインデックスに使用する予定の他のデータソースとの間で、それぞれ異なるものであることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- SharePoint AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager SharePoint データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 SharePoint SharePoint Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、SharePoint Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには SharePoint

1. AWS 管理コンソールにサインインし、Amazon Kendra コンソールを開きます。

2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [SharePoint Connector v1.0] を選択し、[データソースの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ホスティング方法] では、[SharePoint オンライン] と [サーバー] を選択します。SharePoint

      i. SharePointオンラインの場合 — リポジトリ固有のサイト URL を入力します。SharePoint

      ii. SharePointサーバー用 — SharePoint バージョンを選択し、 SharePoint リポジトリ固有のサイト URL を入力し、SSL Amazon S3 証明書の場所へのパスを入力します。

   b. (SharePoint サーバーのみ) Web プロキシの場合: SharePoint 内部インスタンスのホスト名とポート番号を入力します。ポート番号は 0 ~ 65535 の数字である必要があります。

   c. [認証] の場合 - ユースケースに基づいて以下のオプションから選択してください。

i. SharePoint オンラインの場合-基本認証と OAuth 2.0 認証のどちらかを選択します。

ii. SharePoint サーバー用-[なし]、[LDAP]、[手動] のいずれかを選択します。

d. AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、 Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。[シークレット名] を入力する必要があります。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。

e. [AWS Secrets Manager シークレットウィンドウを作成] に次のその他の情報を入力します。

i. ユースケースに応じて、 SharePoint以下のクラウド認証オプションから選択してください。

A. 基本認証 — SharePoint アカウントのユーザー名を [ユーザー名] に、SharePoint アカウントのパスワードを [パスワード] に入力します。

B. OAuth 2.0 認証 — アカウントのユーザー名を [ユーザー名] に、 SharePoint アカウントのパスワードを [パスワード] に、 SharePoint SharePoint 自動生成された一意の ID を [クライアント ID] に、 SharePoint 両方が使用する、またはクライアントシークレットとして使用する共有シークレット文字列を入力します。 Amazon Kendra

ii. ユースケースに応じて、 SharePoint以下のサーバー認証オプションから選択してください。

A. なし — SharePoint アカウントのユーザー名を [ユーザー名] に、 SharePoint アカウントのパスワードを [パスワード] に、サーバーのドメイン名を入力します。

B. LDAP SharePoint ***—################## SharePoint############## ####LDAP ########### (ldap: //example.com: 389 ########### #####)#LDAP ##### (#:dc=example#dc=com) #######***

C. 手動 — アカウントのユーザー名を [ユーザー名] に、 SharePoint アカウントのパスワードを [パスワード] に、 SharePoint 電子メールドメインオーバーライド (ディレクトリユーザーまたはグループの電子メールドメイン) を入力します。

iii. [保存] を選択します。

f. [仮想プライベートクラウド (VPC)] - [サブネット] と [VPC セキュリティグループ] も追加する必要があります。

> **Note**
>
> SharePoint サーバーを使用する場合は VPC を使用する必要があります。Amazon VPC SharePoint他のバージョンではオプションです。

g. IAM ロール — IAM 既存のロールを選択するか、 IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

h. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [変更ログを使用] - 選択すると、すべてのファイルを同期する代わりにインデックスを更新できます。

b. [添付ファイルをクロール] - 選択すると添付ファイルがクロールされます。

c. [ローカルグループマッピングを使用] - 選択すると、ドキュメントが適切にフィルター処理されます。

d. [追加設定] - 特定のファイルを含めるか除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

e. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

f. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. Amazon Kendra デフォルトフィールドマッピング — Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマップするフィールドを選択します。

b. [カスタムフィールドマッピング] の場合 - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには: Amazon Kendra SharePoint

SharePointConfigurationAPI を使用して以下を指定する必要があります。

- SharePointバージョン — SharePoint SharePoint 設定時に使用するバージョンを指定します。これは、 SharePoint サーバー 2013、サーバー 2016、 SharePointサーバー 2019、 SharePoint SharePoint またはオンラインのどれを使用するかにかかわらず当てはまります。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager SharePoint アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは JSON 構造で保存されます。

  SharePoint オンライン基本認証では、シークレットに含める必要のある最小限の JSON 構造は次のとおりです。

```
{
    "userName": "user name",
    "password": "password"
}
```

  SharePoint オンライン OAuth 2.0 認証では、シークレットに含める必要のある最小限の JSON 構造は次のとおりです。

```
{
    "userName": "SharePoint account user name"",
    "password": "SharePoint account password",
    "clientId": "SharePoint auto-generated unique client id",
    "clientSecret": "secret string shared by Amazon Kendra and SharePoint to
 authorize communications"
```

```
}
```

SharePoint Server Basic 認証では、シークレットに含める必要のある最小限の JSON 構造は次のとおりです。

```
{
    "userName": "user name",
    "password": "password",
    "domain": "server domain name"
}
```

SharePoint サーバーの LDAP 認証 (アクセスコントロールリスト (ACL) をメール形式に変換してユーザーコンテキストに基づいてフィルタリングする必要がある場合は、LDAP サーバー URL と LDAP 検索ベースをシークレットに含めることができます)。シークレットに含める必要のある最小限の JSON 構造は次のとおりです。

```
{
    "userName": "user name",
    "password": "password",
    "domain": "server domain name"
    "ldapServerUrl": "ldap://example.com:389",
    "ldapSearchBase": "dc=example,dc=com"
}
```

SharePoint サーバー手動認証の場合、シークレットに含める必要のある最小限の JSON 構造は次のとおりです。

```
{
    "userName": "user name",
    "password": "password",
    "domain": "server domain name",
    "emailDomainOverride": "example.com"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシーク

レットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、 SharePoint Amazon Kendraコネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。詳細については、「IAM SharePoint データソースのロール」を参照してください。
- Amazon VPC SharePoint —Server を使用する場合は、`VpcConfiguration`データソース設定の一部として指定してください。VPC Amazon Kendra を使用するための設定を参照してください。

オプションで、次の機能を追加することもできます。

- ウェブプロキシ — SharePoint ウェブプロキシ経由でサイトの URL に接続するかどうか。SharePointこのオプションはサーバーでのみ使用できます。
- インデックシングリスト — SharePoint 添付ファイルの内容をリストアイテムにインデックスを付けるかどうか Amazon Kendra 。
- 変更ログ — インデックス内のドキュメントを更新する必要があるかどうかを判断するために、 Amazon Kendra SharePoint データソースの変更ログメカニズムを使用すべきかどうか。

> **Note**
>
> Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、 Amazon Kendra SharePoint 変更ログを処理するよりもデータソース内のドキュメントをスキャンするほうが時間がかからない場合があります。 SharePointデータソースとインデックスを初めて同期する場合は、すべてのドキュメントがスキャンされます。

- 包含フィルターと除外フィルター - 特定のコンテンツを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定

した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング — SharePoint データソースフィールドをインデックスフィールドにマップすることを選択します。 Amazon Kendra 詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

詳細はこちら

Amazon Kendra SharePointデータソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendra SharePoint オンラインコネクター入門

SharePoint コネクター V2.0

SharePoint は、Web コンテンツのカスタマイズや、ページ、サイト、ドキュメントライブラリ、リストの作成に使用できる共同ウェブサイト構築サービスです。 Amazon Kendra SharePoint を使用してデータソースのインデックスを作成できます。

Amazon Kendra 現在、 SharePoint SharePointクラウドとサーバー (2013、2016、2019、サブスクリプションエディション) をサポートしています。

> **Note**
>
> SharePoint コネクター SharePointConfiguration V1.0/API Support は 2023 年に終了する予定です。 SharePoint コネクタ V2.0/ API への移行または使用をお勧めします。 TemplateConfiguration

Amazon Kendra SharePoint データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

サポートされている機能

Amazon Kendra SharePoint データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザ ID クロール
- 包含/除外パターン
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra SharePoint を使用してデータソースのインデックスを作成する前に、 SharePoint AWS とアカウントでこれらの変更を行ってください。

SharePoint Online では、以下の点を確認してください。

- SharePoint インスタンス URL をコピーしました。入力したホスト URL の形式は *`https://yourdomain.sharepoint.com/sites/mysite`* です。URL は `https` で始まり、`sharepoint.com` を含む必要があります。

- SharePoint インスタンス URL のドメイン名をコピーした。
- SharePointOnline に接続するためのサイト管理者権限を持つユーザー名とパスワードを含む基本認証情報を書き留めました。
- 管理者ユーザーを使用して Azure Portal の [セキュリティデフォルト] を無効にしました。Azure Portal でのセキュリティのデフォルト設定の管理の詳細については、セキュリティのデフォルトを有効または無効にする方法に関する Microsoft のドキュメントを参照してください。
- SharePoint アカウントの多要素認証 (MFA) Amazon Kendra が無効になっているため、コンテンツのクロールがブロックされません。 SharePoint
- Basic 認証以外の認証タイプを使用している場合:インスタンスのテナント ID をコピーしました。SharePoint テナント ID を確認する方法の詳細については、「Find your Microsoft 365 tenant ID」を参照してください。
- OAuth 2.0 認証と OAuth 2.0 更新トークン認証の場合: SharePoint Online への接続に使用するユーザー名とパスワード、Azure AD SharePoint への登録後に生成されたクライアント ID とクライアントシークレットを含む基本認証資格情報を書き留めておきました。
  - ACL を使用していない場合は、次のアクセス許可が追加されました。

Microsoft Graph	SharePoint
• Notes.Read.All (アプリケーション) — すべてのノートブックを読み込む OneNote • Sites.Read.All (アプリケーション) - すべてのサイトコレクションの項目を読み取る	• AllSites.Read (委任) — すべてのサイトコレクションのアイテムを読み取ります

> **Note**
>
> Note.Read.All と Sites.Read.All は、ドキュメントをクロールする場合にのみ必要です。 OneNote

  - ACL を使用している場合は、次のアクセス許可が追加されました。

Microsoft Graph	SharePoint
• Group.Member.Read.All (アプリケーション) - すべてのグループメンバーシップを読み取る • Notes.Read.All (アプリケーション) — すべてのノートブックを読み込む OneNote • サイト。 FullControl.All (委任) — ドキュメントの ACL を取得するために必要です。 • Sites.Read.All (アプリケーション) - すべてのサイトコレクションの項目を読み取る • User.Read.All (アプリケーション) - すべてのユーザーの完全なプロフィールを読み取る	• AllSites.Read (委任) — すべてのサイトコレクションのアイテムを読み取ります。

> **Note**
>
> GroupMember.Read.All と User.Read.All は ID クローラーがアクティブ化されている場合にのみ必要です。

- Azure AD アプリ専用認証の場合:Azure AD への登録後に生成した秘密鍵とクライアント ID。SharePoint X.509 証明書にも注意してください。
  - ACL を使用していない場合は、次のアクセス許可が追加されました。

SharePoint
• Sites.Read.All (アプリケーション) — すべてのサイトコレクションのアイテムとリストにアクセスするのに必要です。

> **Note**
>
> 特定のサイトをクロールする場合、権限をドメイン内の利用可能なすべてのサイトではなく、特定のサイトに制限できます。Sites.Selected (アプリケーション) 権限を設定します。このAPI権限では、Microsoft Graph APIを通じて各サイトへのアクセス権限を明示的に設定する必要があります。詳細については、「Sites.Selected Permissions」に関するマイクロソフトのブログを参照してください。

- ACL を使用している場合は、次のアクセス許可が追加されました。

SharePoint
• サイト。 FullControl.All (アプリケーション) — ドキュメントの ACL を取得するために必要です。

- SharePoint アプリのみの認証の場合: SharePoint アプリのみに権限を付与する際に生成されたクライアント ID とクライアントシークレット、および Azure AD SharePoint にアプリを登録したときに生成されたクライアント ID とクライアントシークレットを記録しました。 SharePoint

> **Note**
>
> SharePoint アプリのみの認証は 2013 バージョンではサポートされていません。 SharePoint

- (オプション) OneNote ドキュメントをクロールして Identity Crawler を使用している場合は、次の権限が追加されました。

Microsoft Graph
• GroupMember.Read.All (アプリケーション) — すべてのグループメンバーシップを読み取ります。 • Notes.Read.All (アプリケーション) — すべてのノートブックを読み込む OneNote

Microsoft Graph
• Sites.Read.All (アプリケーション) - すべてのサイトコレクションの項目を読み取る • User.Read.All (アプリケーション) - すべてのユーザーの完全なプロフィールを読み取る

> **Note**
>
> 基本認証とアプリ限定認証を使用してエンティティをクロールする場合、API 権限は必要ありません。 SharePoint

SharePoint Server には、次のものが揃っていることを確認してください。

- SharePoint インスタンス URL と URL SharePoint のドメイン名をコピーした。入力したホスト URL の形式は *https://yourcompany/sites/mysite* です。URL は `https` で始まる必要があります。

> **Note**
>
> (オンプレミス/サーバー) AWS Secrets Manager に含まれるエンドポイント情報が、Amazon Kendra データソース設定の詳細で指定されているエンドポイント情報と同じかどうかを確認します。混乱する代理問題は、ユーザーがアクションを実行するアクセス許可がないにもかかわらず、 Amazon Kendra をプロキシとして使用して設定された秘密にアクセスし、アクションを実行するセキュリティの問題です。後でエンドポイント情報を変更する場合は、新しいシークレットを作成してこの情報を同期する必要があります。

- SharePoint アカウントの多要素認証 (MFA) Amazon Kendra が無効になっているため、コンテンツのクロールがブロックされません。 SharePoint
- SharePoint アクセス制御にアプリ専用認証を使用している場合:
  - サイトレベルで App Only SharePoint を登録したときに生成されたクライアント ID をコピーしました。クライアント ID の形式は ClientId @ですTenantId。例: *ffa956f3-8f89-44e7-b0e4-49670756342c@888d0b57-69f1-4fb8-957f-e1f0bedf82fe*。
  - App Only SharePoint をサイトレベルで登録したときに生成されたクライアントシークレットをコピーしました。

注:クライアント ID とクライアントシークレットは、 SharePoint サーバーをアプリ専用認証に登録した場合にのみ単一サイト用に生成されるため、アプリ専用認証でサポートされるサイト URL は 1 つだけです。 SharePoint

> **Note**
>
> SharePoint アプリのみの認証は SharePoint 2013 バージョンではサポートされていません。

- [カスタムドメイン付き E メール ID] をアクセス制御に使用する場合:
  - カスタムメールドメインの値 (例: *"amazon.com"*) を記録しました。
- [IDP からのドメイン付き E メール ID] 認証を使用している場合は、以下をコピーしました。
  - LDAP サーバーエンドポイント (プロトコルとポート番号を含む LDAP サーバーのエンドポイント)。例: *ldap://example.com:389*。
  - LDAP 検索ベース (LDAP ユーザーの検索ベース)。例: *CN=Users#DC=sharepoint#DC=com*.
  - LDAP ユーザー名と LDAP パスワード。
- 構成済みの NTLM 認証資格情報、またはユーザー名 (SharePoint アカウントユーザー名) とパスワード (アカウントパスワード) を含む構成済みの Kerberos 認証資格情報のいずれか。SharePoint

には、 AWS アカウント次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- SharePoint AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager SharePoint データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 SharePoint SharePoint Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、SharePoint Amazon Kendra を参照してください前提条件。

Console: SharePoint Online

Amazon Kendra SharePoint オンラインに接続するには

1. AWS 管理コンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [SharePoint Connector V2.0] を選択し、[データソースの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] の [ホスティング方法] で、[オンライン] を選択しますSharePoint。

b. SharePointリポジトリ固有のサイト URL — SharePoint ホスト URL を入力します。入力したホスト URL の形式は *https://yourdomain.sharepoint.com/sites/mysite* です。URL は `https` プロトコルで始まる必要があります。URL は改行で区切ります。最大 100 個の URL を追加できます。

c. ドメイン — ドメインを入力します。 SharePoint 例えば、URL *https://yourdomain.sharepoint.com/sites/mysite* のドメインは *yourdomain* です。

d. [承認] では、以下の ACL オプションの中から選択できます。

- [ユーザープリンシパル名] – アクセス制御は、Azure ポータルから取得した [ユーザープリンシパル名] に基づいて行われます。
- [E メール] – アクセス制御は、Azure ポータルから取得した E メール ID に基づいて行われます。

> **Note**
>
> 値を指定しない場合、[E メール] がデフォルト値と見なされます。

e. 認証では、ユースケースに基づいて、基本、OAuth 2.0、Azure AD アプリのみの認証、SharePoint アプリのみの認証、OAuth 2.0 更新トークン認証のいずれかを選択します。

i. [基本認証] を使用する場合は、次の情報を入力します。

- AWS Secrets Manager シークレットの場合 — 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。 Secrets Manager SharePoint新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。
  - [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
  - ユーザー名 — アカウントのユーザー名。 SharePoint
  - パスワード — SharePoint アカウントのパスワード。

ii. [OAuth 2.0 認証] を使用する場合は、次の情報を入力します。

- テナント ID — アカウントのテナント ID。 SharePoint
- AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。
  - [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
  - ユーザー名 — アカウントのユーザー名。 SharePoint
  - パスワード — SharePoint アカウントのパスワード。
  - クライアント ID — Azure AD SharePoint への登録時に生成される Azure AD クライアント ID。
  - クライアントシークレット — Azure AD SharePoint への登録時に生成される Azure AD クライアントシークレット。

iii. [Azure AD アプリ専用認証] を使用する場合は、次の情報を入力します。

- テナント ID — アカウントのテナント ID。 SharePoint
- [Azure AD の自己署名 X.509 証明書] - Azure AD のコネクタを認証するための証明書。
- AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。
  - [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。

- クライアント ID — Azure AD SharePoint に登録したときに生成される Azure AD クライアント ID。
- [プライベートキー] - Azure AD のコネクタを認証するためのプライベートキー。

iv. SharePointアプリ限定認証を使用する場合は、次の情報を入力します。

- テナント ID — アカウントのテナント ID。 SharePoint
- AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。
  - [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
  - SharePoint クライアント ID — テナントレベルで App Only SharePoint を登録したときに生成したクライアント ID。*###### ID ## ## ClientID@ ###TenantId*例: *ffa956f3-8f89-44e7-b0e4-49670756342c@888d0b57-69f1-4fb8-957f-e1f0bedf82fe*。
  - SharePoint クライアントシークレット — テナントレベルで SharePoint App Only に登録したときに生成されるクライアントシークレット。
  - クライアント ID — Azure AD SharePoint への登録時に生成される Azure AD クライアント ID。
  - クライアントシークレット — Azure AD SharePoint への登録時に生成される Azure AD クライアントシークレット。

v. [OAuth 2.0 更新トークン認証] を使用する場合は、次の情報を入力します。

- テナント ID — アカウントのテナント ID。 SharePoint
- AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。
  - [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
  - クライアント ID — Azure AD SharePoint への登録時に生成される一意の Azure AD クライアント ID。

- クライアントシークレット — Azure AD SharePoint への登録時に生成される Azure AD クライアントシークレット。
- 更新トークン — Amazon Kendra 接続用に生成された更新トークン。SharePoint

f. [ID クローラー] - (ACL が有効な場合のみ有効) ID 情報を同期するための Amazon Kendra ID クローラーを有効化することを選択します。ID クローラーをオフにする場合は、API を使用してプリンシパル情報をアップロードする必要があります。PutPrincipalMapping

次を選択することもできます。

i. [ローカルグループマッピングをクロール] - 有効にすると、ローカルグループマッピングがクロールされます。

ii. [AD グループマッピングをクロール] - 有効にすると、Azure Active Directory グループマッピングがクロールされます。

> **Note**
>
> AD グループマッピングのクロールは OAuth 2.0、OAuth 2.0 更新トークン、およびアプリのみの認証でのみ使用できます。 SharePoint

g. (オプション) VPC とセキュリティグループの設定 — インスタンスで使用する VPC を選択します。 SharePoint 選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

i. [エンティティの選択] - クロールするエンティティを選択します。[すべて] のエンティティをクロールするか、[ファイル]、[添付ファイル]、[リンク]、[ページ]、[イベント]、[コメント]、[リストデータ] を組み合わせてクロールするかを選択できます。

ii. [追加の設定] では、[エンティティ正規表現パターン] の場合 - [リンク]、[ページ]、[イベント] に正規表現パターンを追加して、すべてのドキュメントを同期する代わりに特定のエンティティを含めることができます。

iii. 正規表現パターン — すべてのドキュメントを同期する代わりに、正規表現パターンを追加して、ファイルパス、ファイル名、ファイルタイプ、OneNote セクション名、OneNote ページ名でファイルを含めたり除外したりします。最大 100 個を追加できます。

> **Note**
>
> OneNote クロールは OAuth 2.0、OAuth 2.0 更新トークン、およびアプリのみの認証でのみ使用できます。 SharePoint

b. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

c. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. イベントページ、ファイル、リンク、添付ファイル、コメント用 — Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングする項目を選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

Console: SharePoint Server

Amazon Kendra 接続するには SharePoint

1. AWS 管理コンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [SharePoint Connector V2.0] を選択し、[データソースの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] の [ホスティング方法] で、[サーバー] を選択しますSharePoint。

b. SharePointバージョンを選択 — SharePoint 2013、SharePoint 2016、SharePoint 2019、および SharePoint (サブスクリプションエディション) から選択します。

c. SharePointリポジトリ固有のサイト URL — SharePoint ホスト URL を入力します。入力したホスト URL の形式は *https://yourcompany/sites/mysite* です。URL は `https` プロトコルで始まる必要があります。URL は改行で区切ります。最大 100 個の URL を追加できます。

d. ドメイン — ドメインを入力します。 SharePoint 例えば、URL *https://yourcompany/sites/mysite* のドメインは *yourcompany* です

e. SSL 証明書の場所-SSL Amazon S3 証明書ファイルへのパスを入力します。

f. (オプション) [ウェブプロキシ] の場合: ホスト名 (`http://` または `https://` プロトコルなし) と、ホスト URL トランスポートプロトコルで使用されるポート番号を入力します。ポート番号の数値は 0 ~ 65535 の間である必要があります。

g. 承認 — ACL があり、それをアクセス制御に使用したい場合は、ドキュメントのアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

   SharePoint Server では、次の ACL オプションから選択できます。

   i. [IDP からのドメイン付き E メール ID] - アクセス制御は、基盤となる ID プロバイダー (IDP) から取得した E メールドメインから抽出された E メール ID に基づいて行われます。認証時に IDP Secrets Manager 接続の詳細をシークレットに入力します。

   ii. [カスタムドメイン付き E メール ID] - アクセス制御は E メール ID に基づいて行われます。E メールドメイン値を指定する必要があります。例: *"amazon.com"*。E メールドメインは、アクセス制御用の E メール ID の作成に使用されます。[E メールドメインを追加] を使用して E メールドメインを入力する必要があります。

   iii. [ドメイン\ドメイン付きユーザー] - アクセス許可は、ドメイン\ユーザー ID 形式を使用して構成されます。有効なドメイン名を指定する必要があります。例えば、アクセス制御を構築するには *"sharepoint2019"* と入力します。

h. 認証では、ユースケースに基づいて、SharePoint アプリのみの認証、NTLM 認証、Kerberos 認証のいずれかを選択します。

   i. [NTLM 認証] と [Kerberos 認証] の両方について、次の情報を入力します。

シークレットの場合 — AWS Secrets Manager 既存のシークレットを選択するか、 Secrets Manager 認証情報を保存する新しいシークレットを作成します。SharePoint新しいシークレットを作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。

- [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
- ユーザー名 — アカウントのユーザー名。 SharePoint
- パスワード — SharePoint アカウントのパスワード。

[IDP からのドメイン付き E メール ID] を使用している場合は、次の情報も入力してください。

- [LDAP サーバーエンドポイント] - プロトコルとポート番号を含む LDAP サーバーのエンドポイント。例: *ldap://example.com:389*。
- [LDAP 検索ベース] - LDAP ユーザーの検索ベース。例: *CN=Users#DC=sharepoint#DC=com*.
- [LDAP ユーザー名] - LDAP ユーザー名。
- [LDAP パスワード] - LDAP パスワード。

ii. SharePoint アプリのみの認証には、次の情報を入力します。

AWS Secrets Manager シークレット用 — 既存のシークレットを選択するか、Secrets Manager SharePoint認証情報を保存する新しいシークレットを作成します。新しいシークレットを作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。

- [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraSharePoint-' がシークレット名に自動的に追加されます。
- クライアント ID — サイトレベルで App Only SharePoint を登録したときに生成したクライアント ID。クライアント ID の形式は ClientID@ です。TenantId例: *ffa956f3-8f89-44e7-b0e4-49670756342c@888d0b57-69f1-4fb8-957f-e1f0bedf82fe*。
- SharePoint クライアントシークレット — サイトレベルで SharePoint App Only に登録したときに生成されるクライアントシークレット。

注:クライアント ID とクライアントシークレットは、 SharePoint サーバーをアプリ専用認証に登録した場合にのみ単一サイト用に生成されるため、アプリ専用認証でサポートされるサイト URL は 1 つだけです。 SharePoint

[IDP からのドメイン付き E メール ID] を使用している場合は、次の情報も入力してください。

- [LDAP サーバーエンドポイント] - プロトコルとポート番号を含む LDAP サーバーのエンドポイント。例: *ldap://example.com:389*。
- [LDAP 検索ベース] - LDAP ユーザーの検索ベース。例: *CN=Users#DC=sharepoint#DC=com*.
- [LDAP ユーザー名] - LDAP ユーザー名。
- [LDAP パスワード] - LDAP パスワード。

i. ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

   i. [ローカルグループマッピングをクロール] - 有効にすると、ローカルグループマッピングがクロールされます。

   ii. ([IDP からのドメイン付き E メール ID] のみ) [Active Directory マッピングをクロールする] - Active Directory マッピングをクロールする有効にします。

> **Note**
>
> AD グループマッピングのクロールは、 SharePoint アプリのみの認証でのみ使用できます。

j. (オプション) VPC とセキュリティグループの設定 — インスタンスで使用する VPC を選択します。 SharePoint 選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

k. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

l. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

i. [エンティティの選択] - クロールするエンティティを選択します。[すべて] のエンティティをクロールするか、[ファイル]、[添付ファイル]、[リンク]、[ページ]、[イベント]、[リストデータ] を組み合わせてクロールするかを選択できます。

ii. [追加の設定] では、[エンティティ正規表現パターン] の場合 - [リンク]、[ページ]、[イベント] に正規表現パターンを追加して、すべてのドキュメントを同期する代わりに特定のエンティティを含めることができます。

iii. 正規表現パターン — すべてのドキュメントを同期する代わりに、ファイルパス、ファイル名、ファイルタイプ、OneNoteセクション名、OneNoteページ名でファイルを含めたり除外したりする正規表現パターンを追加します。最大 100 個を追加できます。

> **Note**
>
> OneNote クロールはアプリのみの認証でのみ可能です。 SharePoint

b. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。

- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

c. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. イベントページ、ファイル、リンク、添付ファイル、リストデータの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングするデータを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra 接続するには SharePoint

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON SHAREPOINTV2 スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- リポジトリエンドポイントメタデータ — siteUrls SharePoint インスタンスの終了を指定します。tenantID domain
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。

- FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

  > **Note**
  >
  > ID クローラーは、に設定した場合にのみ使用できます。`crawlAcl` `true`

- リポジトリの追加プロパティ - 以下を指定します。
  - (Azure AD の場合) `s3bucketName`、`s3certificateName` Azure AD の自己署名 X.509 証明書を保存するのに使用します。
  - 、、、、、、およびのいずれかにかかわらず`OAuth2``OAuth2App``OAuth2CertificateBasic`、`OAuth2_RefreshToken`使用する認証タイプ (`auth_Type`)。`NTLM` `Kerberos`
  - 使用するバージョン (`version`) (`Server`またはにかかわらず) `Online`。`Server` を使用する場合、`onPremVersion` を `2013`、`2016`、`2019`、または `SubscriptionEdition` としてさらに指定できます。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 SharePoint

  SharePoint オンラインを使用する場合は、基本認証、OAuth 2.0 認証、Azure AD アプリのみ、アプリのみの認証を選択できます。 SharePoint 各認証オプションでシークレットに含める必要がある最小の JSON 構造を次に示します。
  - 基本認証

```
{
```

```
    "userName": "SharePoint account user name",
    "password": "SharePoint account password"
}
```

- OAuth 2.0 認証

```
{
    "clientId": "client id generated when registering SharePoint with Azure AD",
    "clientSecret": "client secret generated when registering SharePoint with
 Azure AD",
    "userName": "SharePoint account user name",
    "password": "SharePoint account password"
}
```

- Azure AD アプリ専用認証

```
{
    "clientId": "client id generated when registering SharePoint with Azure AD",
    "privateKey": "private key to authorize connection with Azure AD"
}
```

- SharePoint アプリ専用認証

```
{
    "clientId": "client id generated when registering SharePoint for App Only at
 Tenant Level",
    "clientSecret": "client secret generated when registering SharePoint for App
 Only at Tenant Level",
    "adClientId": "client id generated while registering SharePoint with Azure
 AD",
    "adClientSecret": "client secret generated while registering SharePoint with
 Azure AD"
}
```

- OAuth 2.0 更新トークン認証

```
{
    "clientId": "client id generated when registering SharePoint with Azure AD",
    "clientSecret": "client secret generated when registering SharePoint with
 Azure AD",
    "refreshToken": "refresh token generated to connect to SharePoint"
}
```

SharePoint サーバーを使用する場合は、 SharePoint アプリ限定認証、NTLM 認証、Kerberos 認証のいずれかを選択できます。各認証オプションでシークレットに含める必要がある最小の JSON 構造を次に示します。

- SharePoint アプリ限定認証

```
{
    "siteUrlsHash": "Hash representation of SharePoint site URLs",
    "clientId": "client id generated when registering SharePoint for App Only at
 Site Level",
    "clientSecret": "client secret generated when registering SharePoint for App
 Only at Site Level"
}
```

- SharePoint IDP 認証によるドメインによるアプリ限定認証

```
{
    "siteUrlsHash": "Hash representation of SharePoint site URLs",
    "clientId": "client id generated when registering SharePoint for App Only at
 Site Level",
    "clientSecret": "client secret generated when registering SharePoint for App
 Only at Site Level",
    "ldapUrl": "LDAP Account url eg. ldap://example.com:389",
    "baseDn": "LDAP Account base dn eg. CN=Users,DC=sharepoint,DC=com",
    "ldapUser": "LDAP account user name",
    "ldapPassword": "LDAP account password"
}
```

- (サーバーのみ) NTLM または Kerberos 認証

```
{
    "siteUrlsHash": "Hash representation of SharePoint site URLs",
    "userName": "SharePoint account user name",
    "password": "SharePoint account password"
}
```

- (サーバーのみ) IDP 認証からのドメインによる NTLM または Kerberos 認証

```
{
    "siteUrlsHash": "Hash representation of SharePoint site URLs",
    "userName": "SharePoint account user name",
    "password": "SharePoint account password",
```

```
    "ldapUrl": "ldap://example.com:389",
    "baseDn": "CN=Users,DC=sharepoint,DC=com",
    "ldapUser": "LDAP account user name",
    "ldapPassword": "LDAP account password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに付与し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 SharePoint Amazon Kendra詳細については、「IAM SharePoint データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター — 特定のファイルやその他のコンテンツを含めるか除外するかを指定できます。 OneNotes

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング — SharePoint データソースフィールドをインデックスフィールドにマップすることを選択できます。 Amazon Kendra 詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Microsoft SharePoint テンプレートスキーマ」を参照してください。

メモ

- コネクタは [ファイル] エンティティのカスタムフィールドマッピングのみをサポートします。
- SharePoint すべてのサーバーバージョンで、ACL トークンは小文字でなければなりません。[IDP からのドメイン付き E メール ID] および [カスタムドメイン付き E メール ID] の場合 (例: *user@sharepoint2019.com*)。[ドメイン\ドメイン付きユーザー] ACL の場合 (例: *sharepoint2013\user*)。
- コネクタは 2013 年の変更ログモード/新規または変更されたコンテンツの同期をサポートしていません。 SharePoint
- エンティティ名の名前に % 文字が含まれている場合、API の制限によりコネクタはこれらのファイルをスキップします。
- OneNote コネクタがクロールできるのは、テナント ID を使用し、OAuth 2.0 更新トークン、またはオンラインで OAuth 2.0、OAuth 2.0 更新トークン、またはアプリのみの認証が有効になっている場合のみです。 SharePoint SharePoint
- コネクタは、ドキュメントの名前が変更された場合でも、 OneNote デフォルト名のみを使用してドキュメントの最初のセクションをクロールします。
- クロール対象のエンティティとして「リンク」に加えて「ページ」と「ファイル」が選択されている場合のみ、 SharePoint 2019、 SharePointオンラインエディション、サブスクリプションエディションのリンクがクロールされます。
- クロールするエンティティとしてリンクが選択されている場合、コネクタは SharePoint 2013 SharePoint 年と 2016 年にリンクをクロールします。
- コネクタがリストの添付ファイルとコメントをクロールするのは、クロール対象のエンティティとして [リストデータ] も選択されている場合のみです。

- コネクタがイベント添付ファイルをクロールするのは、クロール対象のエンティティとして [イベント] も選択されている場合のみです。
- SharePoint オンラインバージョンでは、ACL トークンは小文字になります。たとえば、Azure ポータルのユーザープリンシパル名が MaryMajor *@domain .com* の場合、 SharePoint コネクタの ACL トークンは *marymajor@domain.com* になります。
- SharePoint オンラインおよびサーバー用 Identity Crawler で、ネストされたグループをクロールする場合は、AD グループクロールだけでなくローカルクロールも有効にする必要があります。
- SharePoint Online を使用していて、Azure Portal のユーザープリンシパル名が大文字と小文字の組み合わせである場合、 SharePoint API は内部的に小文字に変換します。このため、 Amazon Kendra SharePoint コネクタは ACL を小文字に設定します。

Microsoft SQL Server

Microsoft SQL Server は、Microsoft が開発したリレーショナルデータベース管理システム (RDBMS) です。Microsoft SQL Serverユーザーであれば、 Amazon Kendra Microsoft SQL Serverを使用してデータソースのインデックスを作成できます。 Amazon Kendra Microsoft SQL Serverデータソースコネクタは MS SQL Server 2019 をサポートしています。

Amazon Kendra コンソールと TemplateConfigurationAPI Amazon Kendra Microsoft SQL Server を使用してデータソースに接続できます。

Amazon Kendra Microsoft SQL Serverデータソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター

- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Microsoft SQL Serverを使用してデータソースのインデックスを作成する前に、Microsoft SQL Server AWS とアカウントでこれらの変更を行ってください。

Microsoft SQL Server で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが Microsoft SQL Server および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Microsoft SQL Server の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Microsoft SQL Serverデータソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Microsoft SQL ServerMicrosoft SQL Server Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Microsoft SQL Server Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Microsoft SQL Server

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Microsoft SQL Serverコネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] - データベースのホスト名を入力します。

c. [ポート] - データベースのポートを入力します。

d. [インスタンス] - データベースインスタンスを入力します。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Microsoft SQL Server 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraMicrosoft SQL Server-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。

> **Note**
>
> テーブル名の名前に特殊文字 (英数字以外) が含まれている場合は、テーブル名を角括弧で囲む必要があります。たとえば、`[] ## [*] ######`。my-database-table

- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- 本文列 — データベーステーブル内の文書本文列の名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。

- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Microsoft SQL Server

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを `sqlserver` として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。

  > **Note**
  >
  > テーブル名の名前に特殊文字 (英数字以外) が含まれている場合は、テーブル名を角括弧で囲む必要があります。たとえば、*`[] ## [*] ######`*。my-database-table

- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - `FORCED_FULL_CRAWL` は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - `FULL_CRAWL` は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - `CHANGE_LOG` は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Microsoft SQL Serverシークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。Microsoft SQL Server Amazon Kendra詳細については、「IAM roles for Microsoft SQL Server data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Microsoft SQL Server データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Microsoft SQL サーバーテンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Microsoft Teams

Microsoft Teams は、メッセージング、会議、ファイル共有のためのエンタープライズコラボレーションツールです。Microsoft Teams ユーザーの場合は、 Amazon Kendra を使用して Microsoft Teams データソースのインデックスを作成できます。

Amazon Kendra コンソールと TemplateConfigurationAPI Amazon Kendra を使用して Microsoft Teams データソースに接続できます。

Amazon Kendra Microsoft Teams データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザー ID クロール
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Microsoft Teams データソースのインデックスを作成する前に、Microsoft Teams AWS とアカウントでこれらの変更を行ってください。

Microsoft Teams で以下を確認してください。

- Office 365 で Microsoft Teams アカウントを作成しました。
- Microsoft 365 のテナント ID を記録しました。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- Azure ポータルで OAuth アプリケーションを作成し、クライアント ID、クライアントシークレット、またはクライアント認証情報を記録しました。詳細については、「Microsoft チュートリアル」と「登録済みアプリの例」を参照してください。

  > **Note**
  >
  > Azure Portal でアプリを作成または登録すると、シークレット ID は実際のシークレット値を表します。シークレットとアプリを作成したら、すぐに実際のシークレット値を書き留めるか保存する必要があります。シークレットにアクセスするには、Azure Portal でアプリケーションの名前を選択し、証明書とシークレットのメニューオプションに移動します。
  > クライアント ID にアクセスするには、Azure Portal でアプリケーションの名前を選択し、概要ページに移動します。アプリケーション (クライアント) ID はクライアント ID です。

- 必要なアクセス権限を追加しました。すべてのアクセス許可を追加できますが、クロールするエンティティに基づいて選択するアクセス許可を減らすことによって範囲を制限することもできます。対応するエンティティ別のアクセス許可の表を以下に示します。

エンティティ	データ同期に必要なアクセス許可	ID 同期に必要なアクセス許可
チャネルポスト	• ChannelMessage.Read.All • Group.Read.All • User.Read • User.Read.All	TeamMember.Read. All
チャネルアタッチメント	• ChannelMessage.Read. All • Group.Read.All • User.Read • User.Read.All	TeamMember.Read. All
チャネル Wiki	• Group.Read.All • User.Read • User.Read.All	TeamMember.Read. All
チャットメッセージ	• Chat.Read.All • ChatMessage.Read. All • ChatMember.Read. All • User.Read • User.Read.All • Group.Read.All	TeamMember.Read. All
会議チャット	• Chat.Read.All • ChatMessage.読み取り • ChatMember.Read. All • User.Read • User.Read.All • Group.Read.All	TeamMember.Read. All

エンティティ	データ同期に必要なアクセス許可	ID 同期に必要なアクセス許可
チャットアタッチメント	• Chat.Read.All • ChatMessage.読み取り • ChatMember.Read. All • User.Read • User.Read.All • Group.Read.All	TeamMember.Read. All
会議ファイル	• Chat.Read.All • ChatMessage.Read. All • ChatMember.Read. All • User.Read • User.Read.All • Group.Read.All • Files.Read.All	TeamMember.Read. All
カレンダー会議	• Chat.Read.All • ChatMessage.Read. All • ChatMember.Read. All • User.Read • User.Read.All • Group.Read.All • Files.Read.All	TeamMember.Read. All
会議メモ	• User.Read • User.Read.All • Group.Read.All • Files.Read.All	TeamMember.Read. All

- 各ドキュメントが Microsoft Teams および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体

に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

の中に AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Microsoft Teams の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールまたはシークレットがない場合は、Microsoft Teams データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Microsoft Teams Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Microsoft Teams データソースの必要な詳細情報を入力する必要があります。まだ Microsoft Teams を設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Microsoft Amazon Kendra チームに接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Microsoft Teams コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引の対象となるドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [ソース] - Microsoft 365 テナント ID を入力します。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
   b. AWS Secrets Manager secret — Microsoft Teams Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス「AmazonKendra-Microsoft Teams-」がシークレット名に自動的に追加されます。

B. [クライアント ID] および [クライアントシークレット] - Azure ポータルの Microsoft Teams アカウントで生成した認証情報の値を入力します。

ii. [保存] を選択します。

c. [支払いモデル] - Microsoft Teams アカウントのライセンスと支払いモデルを選択できます。モデル A の支払いモデルは、セキュリティコンプライアンスを必要とするライセンスモデルと支払いモデルに限定されます。モデル B の支払いモデルは、セキュリティコンプライアンスを必要としないライセンスモデルや支払いモデルに適しています。

d. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

e. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

f. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

g. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [コンテンツを同期] - 同期するコンテンツを選択します。

b. [追加設定] - オプションで、すべてのドキュメントを同期する代わりに、特定のコンテンツのインデックスを作成する設定を使用できます。

c. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択できます。

i. 完全同期を選択すると、 Amazon Kendra は、以前の同期ステータスに関係なく、すべてのエンティティのすべてのコンテンツを同期します。

ii. 新規または変更されたコンテンツの同期を選択すると、 Amazon Kendra は、新しいコンテンツまたは変更されたコンテンツのみが同期されます。

iii. 新規、変更、削除済みコンテンツの同期を選択すると、新規、変更、 Amazon Kendra 削除済みのコンテンツのみが同期されます。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. デフォルトデータソースフィールド- Amazon Kendra 生成されたデフォルトデータソースフィールドの中から、インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Microsoft Amazon Kendra チームに接続するには

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON `MSTEAMS` スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI `TEMPLATE` を呼び出すときと同じようにデータソースを指定します。
- テナント ID - テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。

- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレットAmazon リソースネーム (ARN) — Microsoft Teams Secrets Manager アカウントの認証認証情報を含むシークレットのAmazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "clientId": "client ID",
    "clientSecret": "client secret"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Microsoft Teams Amazon Kendraコネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。詳細については、「IAM roles for Microsoft Teams data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。

- 包含フィルターと除外フィルター - Microsoft Teams で特定のコンテンツを含めるか除外するかを指定します。チーム名、チャンネル名、ファイル名とファイルタイプ、ユーザーの電子メール、 OneNote セクション、 OneNote ページを含めたり除外したりできます。また、チャットのメッセージと添付ファイル、チャネルの投稿と添付ファイル、チャネル Wiki、カレンダーコンテンツ、会議チャット、ファイルとメモのインデックスを作成するかどうかも指定できます。

  > **Note**
  >
  > ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- フィールドマッピング - 選択すると、Microsoft Teams データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

  > **Note**
  >
  > ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Microsoft Teams テンプレートスキーマ」を参照してください。

詳細はこちら

Microsoft Teams Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Microsoft Amazon Kendra Teams用のコネクタを使用して、組織のMicrosoft Teamsデータソースをインテリジェントに検索します

Microsoft Yammer

Microsoft Yammer は、メッセージング、会議、ファイル共有のためのエンタープライズコラボレーションツールです。Microsoft Yammer ユーザーの場合は、 Amazon Kendra を使用して Microsoft Yammer データソースのインデックスを作成できます。

Amazon Kendra コンソールと TemplateConfigurationAPI Amazon Kendra を使用して Microsoft Yammer データソースに接続できます。

Amazon Kendra Microsoft Yammer データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

サポートされている機能

- フィールドマッピング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Microsoft Yammer データソースのインデックスを作成する前に、Microsoft Yammer AWS とアカウントでこれらの変更を行ってください。

Microsoft Yammer で以下を確認してください。

- Office 365 でMicrosoft Yammer 管理アカウントを作成しました。

- Microsoft Yammer のユーザー名とパスワードを記録しました。
- Microsoft 365 のテナント ID を記録しました。テナント ID は Azure Active Directory ポータルのプロパティまたは OAuth アプリケーションで確認できます。
- Azure ポータルで OAuth アプリケーションを作成し、クライアント ID、クライアントシークレット、またはクライアント認証情報を記録しました。詳細については、「Microsoft チュートリアル」と「登録済みアプリの例」を参照してください。

> **Note**
>
> Azure Portal でアプリを作成または登録すると、シークレット ID は実際のシークレット値を表します。シークレットとアプリを作成したら、すぐに実際のシークレット値を書き留めるか保存する必要があります。シークレットにアクセスするには、Azure Portal でアプリケーションの名前を選択し、証明書とシークレットのメニューオプションに移動します。
> クライアント ID にアクセスするには、Azure Portal でアプリケーションの名前を選択し、概要ページに移動します。アプリケーション (クライアント) ID はクライアント ID です。

- 各ドキュメントが Microsoft Yammer および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、以下の情報が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Microsoft Yammer の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールまたはシークレットがない場合は、Microsoft Yammer データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Microsoft Yammer Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Microsoft Yammer データソースの必要な詳細情報を入力する必要があります。まだ Microsoft Yammer を設定していない場合は Amazon Kendra、を参照してください前提条件。

Console

Microsoft Amazon Kendra Yammer に接続するには

1. AWS Management Console にサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Microsoft Yammer コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引の対象となるドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. ソース - Microsoft Yammer URL を使用します。

b. AWS Secrets Manager secret — Microsoft Yammer Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Microsoft Yammer-' がシークレット名に自動的に追加されます。

B. [ユーザー名]、[パスワード] - Microsoft Yammer のユーザー名とパスワードを入力します。

C. [クライアント ID]、[クライアントシークレット] - Azure ポータルの Microsoft Yammer アカウントから生成した認証情報の値を入力します。

ii. [保存] を選択します。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID ク

ローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

e. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [開始日] - Microsoft Yammer でデータのクローリングを開始する日付を指定します。

b. [コンテンツを同期] - インデックスを作成するコンテンツのタイプを選択します。例えば、公開メッセージ、非公開メッセージ、添付ファイルなどです。

c. [追加設定] - オプションで、すべてのドキュメントを同期する代わりに、特定のコンテンツのインデックスを作成するオプションを使用できます。例えば、特定のコミュニティ名のインデックスを作成し、正規表現パターンを使用して、特定のファイルを含めたり除外したりすることができます。

d. [同期モード] - データソースのコンテンツが変更されたときのインデックスの更新方法を選択できます。

i. 完全同期を選択すると、 Amazon Kendra は、以前の同期ステータスに関係なく、すべてのエンティティのすべてのコンテンツを同期します。

ii. 新規または変更されたコンテンツの同期を選択すると、 Amazon Kendra は、新しいコンテンツまたは変更されたコンテンツのみが同期されます。

iii. 新規、変更、削除済みコンテンツの同期を選択すると、新規、変更、 Amazon Kendra 削除済みのコンテンツのみが同期されます。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. デフォルトデータソースフィールド- Amazon Kendra 生成されたデフォルトデータソースフィールドの中から、インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Microsoft Amazon Kendra Yammer に接続するには

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON YAMMER スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレットアマゾンリソースネーム (ARN) — Microsoft Yammer Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password",
    "clientId": "client ID",
    "clientSecret": "client secret"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Microsoft Yammer コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Microsoft Yammer data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 特定のコンテンツを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使

いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

- フィールドマッピング - 選択すると、Microsoft Yammer データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

詳細はこちら

Microsoft Yammer Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- 以下のための Yammer コネクタについてのお知らせ Amazon Kendra

MySQL

MySQL は、オープンソースのリレーショナルデータベース管理システムです。MySQLユーザーであれば、 Amazon Kendra MySQLを使用してデータソースのインデックスを作成できます。 Amazon Kendra MySQLデータソースコネクタは MySQL 8.0 をサポートします。21.

Amazon Kendra コンソールと API Amazon Kendra MySQL を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra MySQLデータソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順

- メモ

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra MySQLを使用してデータソースのインデックスを作成する前に、MySQL AWS とアカウントでこれらの変更を行ってください。

MySQL で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが MySQL および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- MySQL の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager MySQLデータソースを に接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、MySQLMySQL Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、MySQL Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには MySQL

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [MySQLコネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [ソース] には、次の情報を入力します。
   b. [ホスト] - データベースのホスト名を入力します。
   c. [ポート] - データベースのポートを入力します。
   d. [インスタンス] - データベースインスタンスを入力します。
   e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。
   f. [認証] には、次の情報を入力します。
      - AWS Secrets Manager secret — MySQL 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraMySQL-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra MySQL

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを mySql として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。MySQLシークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。MySQL Amazon Kendra詳細については、「IAM roles for MySQL data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、MySQL データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Oracle Database

Oracle Database はデータベース管理システムです。Oracle Databaseユーザーであれば、 Amazon Kendra Oracle Databaseを使用してデータソースのインデックスを作成できます。 Amazon Kendra Oracle Databaseデータソースコネクタは Oracle データベース 18c、19c、および 21c をサポートします。

Amazon Kendra コンソールと API Amazon Kendra Oracle Database を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra Oracle Databaseデータソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

- フィールドマッピング

- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra Oracle Databaseを使用してデータソースのインデックスを作成する前に、Oracle Database AWS とアカウントでこれらの変更を行ってください。

Oracle Database で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが Oracle Database および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Oracle Database の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager Oracle Databaseデータソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、Oracle DatabaseOracle Database Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、Oracle Database Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには Oracle Database

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Oracle Databaseコネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. [ソース] には、次の情報を入力します。

b. [ホスト] - データベースのホスト名を入力します。

c. [ポート] - データベースのポートを入力します。

d. [インスタンス] - データベースインスタンスを入力します。

e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

f. [認証] には、次の情報を入力します。

- AWS Secrets Manager secret — Oracle Database 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraOracle Database-' がシークレット名に自動的に追加されます。

II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

B. [保存] を選択します。

g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。

- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra Oracle Database

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON JDBC スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを oracle として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。Oracle Database シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。Oracle Database Amazon Kendra詳細については、「IAM roles for Oracle Database data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Oracle Database データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Oracle Database テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。

- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

PostgreSQL

PostgreSQL は、オープンソースのデータベース管理システムです。PostgreSQLユーザーであれば、 Amazon Kendra PostgreSQLを使用してデータソースのインデックスを作成できます。Amazon Kendra PostgreSQLデータソースコネクタは PostgreSQL 9.6 をサポートしています。

Amazon Kendra コンソールと API Amazon Kendra PostgreSQL を使用してデータソースに接続できます。TemplateConfiguration

Amazon Kendra PostgreSQLデータソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

サポートされている機能

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra PostgreSQLを使用してデータソースのインデックスを作成する前に、PostgreSQL AWS とアカウントでこれらの変更を行ってください。

PostgreSQL で以下を確認してください。

- データベースユーザー名とパスワードを記録済み。

> **Important**
>
> ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。Amazon Kendra

- コピーしたデータベースのホスト URL、ポート、インスタンス。
- 各ドキュメントが PostgreSQL および同じインデックスを使用予定の他のデータソース間で一意であることが確認されていること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- PostgreSQL の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager PostgreSQLデータソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます Amazon Kendra。API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、PostgreSQLPostgreSQL Amazon Kendra データにアクセスできるように認証情報の詳細を入力する必要があります。まだ設定していない場合は、PostgreSQL Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには PostgreSQL

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [PostgreSQLコネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [ソース] には、次の情報を入力します。

   b. [ホスト] - データベースのホスト名を入力します。

   c. [ポート] - データベースのポートを入力します。

   d. [インスタンス] - データベースインスタンスを入力します。

   e. SSL 証明書の場所を有効にする-SSL Amazon S3 証明書ファイルへのパスを入力することを選択します。

   f. [認証] には、次の情報を入力します。

      - AWS Secrets Manager secret — PostgreSQL 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

        A. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

           I. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraPostgreSQL-' がシークレット名に自動的に追加されます。

           II. [データベースユーザー名] と [パスワード] - データベースからコピーした認証情報の値を入力します。

        B. [保存] を選択します。

   g. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

   h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

      > **Note**
      >
      > IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

   i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [同期の範囲] で、次のオプションから選択します。

- [SQL クエリ] - SELECT や JOIN オペレーションなどの SQL クエリステートメントを入力します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- [プライマリキー列] - データベーステーブルのプライマリキーを指定します。これにより、データベース内のテーブルが識別されます。
- [タイトル列] - データベーステーブル内のドキュメントタイトル列の名前を指定します。
- ボディカラム — データベーステーブル内のドキュメントボディカラムの名前を指定します。

b. [その他の設定 - オプション] で、すべてのファイルを同期する代わりに特定のコンテンツを同期するには、次のオプションから選択します。

- 変更検出列- Amazon Kendra コンテンツの変更を検出するために使用する列の名前を入力します。 Amazon Kendra これらの列のいずれかに変更があると、コンテンツのインデックスを再作成します。
- [ユーザー ID 列] - コンテンツへのアクセスを許可するユーザー ID を含む列の名前を入力します。
- [グループ列] - コンテンツへのアクセスを許可するグループを含む列の名前を入力します。
- [ソース URL 列] - インデックスを作成するソース URL を含む列の名前を入力します。
- タイムスタンプ列-タイムスタンプを含む列の名前を入力します。 Amazon Kendra タイムスタンプ情報を使用してコンテンツの変更を検出し、変更されたコンテンツのみを同期します。
- [タイムゾーン列] - クロールするコンテンツのタイムゾーンを含む列の名前を入力します。
- [タイムスタンプの形式] - コンテンツの変更を検出してコンテンツを再同期するために使用するタイムスタンプの形式を含む列の名前を入力します。

c. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。 Amazon Kendra

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。

- [新規または変更済みのドキュメントを同期] - 新規または変更済みのドキュメントのみを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

d. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールド (ドキュメント ID、ドキュメントタイトル、ソース URL) から、 Amazon Kendra インデックスにマップしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra に接続するには:PostgreSQL

TemplateConfigurationAPI を使用して以下を指定する必要があります。

- データソース — TemplateConfigurationJSON `JDBC` スキーマを使用する場合と同様にデータソースタイプを指定します。また、CreateDataSourceAPI `TEMPLATE` を呼び出すときと同じようにデータソースを指定します。
- データベースタイプ - データベースタイプを `postgresql` として指定する必要があります。
- SQL クエリ — SELECT や JOIN オペレーションなどの SQL クエリステートメントを指定します。SQL クエリは 32 KB 未満にする必要があります。 Amazon Kendra はクエリに一致するすべてのデータベースコンテンツをクロールします。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。

- FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
- FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。

- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。PostgreSQLシークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "user name": "database user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。PostgreSQL Amazon Kendra詳細については、「IAM roles for PostgreSQL data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ユーザー ID、グループ、ソース URL、タイムスタンプ、タイムゾーンを使用して、特定のコンテンツを含めるかどうかを指定できます。
- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のAmazon Kendra ACL がある場合、ドキュメントのアクセス制御リスト (ACL) をクロールします。ACL 情報

は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

- フィールドマッピング - 選択すると、PostgreSQL データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「PostgreSQL テンプレートスキーマ」を参照してください。

メモ

- 削除されたデータベース行は、 Amazon Kendra 更新されたコンテンツをチェックしても追跡されません。
- データベースの 1 行のフィールド名と値のサイズは 400 KB を超えることはできません。
- データベースデータソースに大量のデータがあり、 Amazon Kendra 初回同期後にすべてのデータベースコンテンツにインデックスを付けたくない場合は、新規、変更、または削除されたドキュメントのみを同期するように選択できます。
- ベストプラクティスとして、読み取り専用のデータベース認証情報を指定してください。 Amazon Kendra
- ベストプラクティスとして、機密データや個人を特定できる情報 (PII) を含むテーブルを追加することは避けてください。

Quip

Quip は、リアルタイムのドキュメント作成機能を提供する共同生産性向上ソフトウェアです。Amazon Kendra を使用して Quip フォルダ、ファイル、ファイルコメント、チャットルーム、添付ファイルのインデックスを作成できます。

Amazon Kendra コンソールと API を使用して Quip Amazon Kendra データソースに接続できます。QuipConfiguration

Amazon Kendra Quip データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Quip データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Quip データソースのインデックスを作成する前に、Quip とアカウントで以下の変更を行ってください。 AWS

Quip で以下を確認してください。

- 管理者アクセス許可を持つ Quip アカウント。
- 個人アクセストークンを含む Quip 認証情報の作成。詳細については、認証に関する Quip ドキュメントを参照してください。

- Quip サイトドメインのコピー。例えば、*https://quip-company.quipdomain.com/browse* の場合、ドメインは *quipdomain* です。
- 各ドキュメントが Quip および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Quip の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Quip データソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Quip Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Quip データソースの必要な詳細情報を入力する必要があります。Quip をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Quip Amazon Kendra に接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [Quip コネクタ] を選択し、[コネクタの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. Quip ドメイン名 - Quip アカウントからコピーした Quip を入力します。

b. AWS Secrets Manager secret — Quip Secrets Manager 認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

   i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

      A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Quip-' がシークレット名に自動的に追加されます。

      B. Quip トークン - Quip アカウントで作成した Quip 個人アクセストークンを入力します。

   ii. [保存] を選択します。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. IAM role — 既存のロールを選択するか、 IAM 新しいロールを作成して、 IAM リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. クロールする Quip フォルダ ID を追加 - クロールする Quip フォルダ ID。

> **Note**
>
> ルートフォルダー (その中のすべてのサブフォルダーとドキュメントを含む) をクロールするには、ルートフォルダー ID を入力します。特定のサブフォルダーをクロールするには、特定のサブフォルダー ID を追加します。

b. 追加設定 (コンテンツタイプ) - クロールするコンテンツタイプを入力します。

c. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

d. [同期実行スケジュール] の [頻度] で、Amazon Kendra がデータソースと同期する頻度を選択します。

e. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 生成されたデフォルトのデータソースフィールドから、インデックスにマップする項目を選択します。 Amazon Kendra

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Quip Amazon Kendra に接続するには

QuipConfiguration API を使用して以下を指定する必要があります。

- Quip サイトドメイン - 例えば、*https://quip-company.quipdomain.com/browse* の場合、ドメインは *quipdomain* です。
- シークレットの Amazon リソースネーム (ARN) — Quip Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "accessToken": "token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Quip コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「Quip データソースのためのIAM ロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - データソース設定の一部として `VpcConfiguration` を指定します。「VPC を使用するための Amazon Kendra の設定」を参照してください。
- 包含フィルターと除外フィルター - 特定のファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フォルダー — インデックスを作成する Quip フォルダーとサブフォルダーを指定します。

> **Note**
>
> ルートフォルダ (その中のすべてのサブフォルダとドキュメントを含む) をクロールするには、ルートフォルダ ID を入力します。特定のサブフォルダーをクロールするには、特定のサブフォルダー ID を追加します。

- 添付ファイル、チャットルーム、ファイルコメント — 添付ファイル、チャットルームのコンテンツ、ファイルコメントのクロールを含めるかどうかを選択します。
- フィールドマッピング - 選択すると、Quip データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

詳細はこちら

Quip Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- Quip コネクタを使用したインテリジェント検索により Quip ドキュメント内のナレッジを検索できます。 Amazon Kendra

Salesforce

Salesforce は、サポート、営業、マーケティングの各チームを管理するための顧客関係管理 (CRM) ツールです。 Amazon Kendra を使用して Salesforce 標準オブジェクト、さらにはカスタムオブジェクトにもインデックスを付けることができます。

Amazon Kendra コンソール、API、または TemplateConfigurationAPI のいずれかを使用して Salesforce Amazon Kendra データソースに接続できます。SalesforceConfiguration

Amazon Kendra には 2 つのバージョンの Salesforce コネクタがあります。各バージョンでサポートされる機能は次のとおりです。

Salesforce コネクタ V1.0/ API SalesforceConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター

Salesforce コネクター V2.0/API TemplateConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザ ID クローリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- エンティティ添付ファイル:クロール
- 仮想プライベートクラウド (VPC)

> **Note**
>
> Salesforce コネクター SalesforceConfiguration V1.0/API Support は 2023 年に終了する予定です。Salesforce コネクタ V2.0/ API に移行するか、使用することをお勧めします。TemplateConfiguration

Amazon Kendra Salesforce データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- Salesforce コネクタ V1.0
- Salesforce コネクタ V2.0

Salesforce コネクタ V1.0

Salesforce は、サポート、営業、マーケティングの各チームを管理するための顧客関係管理 (CRM) ツールです。 Amazon Kendra を使用して Salesforce 標準オブジェクトやカスタムオブジェクトのインデックスを作成できます。

> **Important**
>
> Amazon Kendra Salesforce API バージョン 48 を使用しています。Salesforce API は、1 日あたりに可能なリクエストの数を制限します。Salesforce がこれらのリクエストを超えると、続行できるまで再試行されます。

> **Note**
>
> Salesforce コネクター SalesforceConfiguration V1.0/API Support は 2023 年に終了する予定です。Salesforce コネクタ V2.0/ API に移行するか、使用することをお勧めします。TemplateConfiguration

Amazon Kendra Salesforce データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順

サポートされている機能

Amazon Kendra Salesforce データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター

前提条件

Amazon Kendra を使用して Salesforce データソースのインデックスを作成する前に、Salesforce とアカウントでこれらの変更を行ってください。 AWS

Salesforce で、次の作業を行ったことを確認してください。

- Salesforce アカウントを作成し、Salesforce への接続に使用するユーザー名とパスワードをメモすること。
- OAuth を有効にした Salesforce 接続アプリケーションアカウントを作成し、Salesforce 接続アプリケーションに割り当てられたコンシューマーキー (クライアント ID) とコンシューマーシークレット (クライアントシークレット) をコピーすること。詳細については、接続アプリケーションに関する Salesforce のドキュメントを参照してください。

- Salesforce への接続に使用されるアカウントに関連付けられた Salesforce セキュリティトークンをコピーすること。
- インデックスを作成する Salesforce インスタンスの URL をコピーすること。通常、URL は *https://<company>.salesforce.com/* です。サーバーは Salesforce 接続アプリケーションを実行している必要があります。
- ReadOnly プロファイルを複製し、[すべてのデータの表示] 権限と [記事の管理] 権限を追加して、Salesforce への読み取り専用アクセス権を持つユーザの認証情報を Salesforce サーバに追加しました。これらの認証情報は、接続を行うユーザと、接続先の Salesforce 接続アプリケーションを識別します。 Amazon Kendra
- 各ドキュメントが Salesforce および同じインデックスに使用する予定の他のデータソース間で一意であると確認すること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Salesforce の認証資格情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Salesforce IAM Secrets Manager データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Salesforce Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Salesforce データソースに関する必要な詳細情報を入力する必要があります。まだ Salesforce を設定していない場合は、を参照してください。 Amazon Kendra 前提条件

Console

Salesforce Amazon Kendra に接続するには

1. AWS マネジメントコンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースを追加する] ページで [Salesforce コネクタ V1.0] を選択し、[コネクタを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [データソース名] - データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. [デフォルト言語] - インデックスのドキュメントをフィルターする言語。特に指定しない限り、言語はデフォルトで英語に設定されます。メタデータで指定された言語は、選択した言語よりも優先されます。
   d. [新しいタグを追加] - リソースの検索とフィルタリング、共有コストの追跡を行うためのタグ。
   e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. [Salesforce URL] - インデックスを作成する Salesforce サイトのインスタンス URL を入力します。

   b. [認証のタイプ] で [既存] または [新規] を選択して Salesforce 認証情報を保存します。新しいシークレットを作成することを選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      - [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

        A. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendra-Salesforce-' がシークレット名に自動的に追加されます。

        B. [ユーザー名]、[パスワード]、[セキュリティトークン]、[コンシューマーキー]、[コンシューマーシークレット]、[認証 URL] には、Salesforce アカウントで作成した認証情報の値を入力します。

        C. [認証を保存] を選択します。

   c. IAM role — 既存のロールを選択するか、 IAM 新しいロールを作成して、 IAM リポジトリの認証情報とインデックスコンテンツにアクセスします。

      > **Note**
      >
      > IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

   d. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [添付ファイルのクロール] - 選択すると、添付されたすべてのオブジェクト、記事、フィードがクロールされます。

   b. [標準オブジェクト]、[ナレッジ記事]、[Chatter フィード] - クロールする Salesforce エンティティまたはコンテンツタイプを選択します。

> **Note**
>
> インデックス作成には、標準オブジェクト、ナレッジ記事、または Chatter フィードのうち少なくとも 1 つの設定情報を入力する必要があります。[ナレッジ記事] をクロールすると選択した場合は、インデックスを作成するナレッジ記事のタイプ、記事の名前、すべてのナレッジ記事の標準フィールドをインデックス化するのか、カスタム記事タイプのフィールドのみをインデックス化するのかを指定する必要があります。カスタム記事のインデックスを作成する場合は、記事タイプの内部名を指定する必要があります。最大 10 個の記事タイプを指定できます。

c. 頻度 — Amazon Kendra データソースと同期する頻度。

d. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 標準ナレッジ記事、標準オブジェクト添付ファイル、その他の推奨フィールドマッピングの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマッピングするフィールドを選択します。

> **Note**
>
> `_document_body` へのインデックスマッピングが必要です。`Salesforce ID` フィールドと Amazon Kendra `_document_id` フィールド間のマッピングは変更できません。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Salesforce Amazon Kendra に接続するには

次の SalesforceConfigurationAPI を指定する必要があります。

- サーバー URL - インデックスを作成する Salesforce サイトのインスタンス URL。
- シークレットアマゾンリソースネーム (ARN) — Salesforce Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "authenticationUrl": "OAUTH endpoint that Amazon Kendra connects to get an
 OAUTH token",
    "consumerKey": "Application public key generated when you created your
 Salesforce application",
    "consumerSecret": "Application private key generated when you created your
 Salesforce application.",
    "password": "Password associated with the user logging in to the Salesforce
 instance",
    "securityToken": "Token associated with the user account logging in to the
 Salesforce instance",
    "username": "User name of the user logging in to the Salesforce instance"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Salesforce コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「Salesforce データソースのIAM ロール」を参照してください。
- インデックス作成には、標準オブジェクト、ナレッジ記事、または Chatter フィードのうち少なくとも 1 つの設定情報を入力する必要があります。
  - 標準オブジェクト - [標準オブジェクト] のクロールを選択した場合は、標準オブジェクトの名前と、ドキュメントコンテンツを含む標準オブジェクトテーブル内のフィールドの名前を指定する必要があります。
  - ナレッジ記事 - [ナレッジ記事] をクロールすると選択した場合は、インデックスを作成するナレッジ記事のタイプ、インデックスを作成するナレッジの状態、すべてのナレッジ記事

の標準フィールドをインデックス化するのか、カスタム記事タイプのフィールドのみをインデックス化するのかを指定する必要があります。

- Chatter フィード — Chatter フィードをクロールする場合は、インデックスを作成するコンテンツを含む Salesforce FeedItem テーブルの列の名前を指定する必要があります。

オプションで、次の機能を追加することもできます。

- 包含フィルターと除外フィルター - 特定の添付ファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング - 選択すると、Salesforce データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- ユーザーコンテキストフィルタリングとアクセス制御 — 文書用の ACL がある場合、文書のアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。

Salesforce コネクタ V2.0

Salesforce は、サポート、営業、マーケティングの各チームを管理するための顧客関係管理 (CRM) ツールです。 Amazon Kendra を使用して Salesforce 標準オブジェクトやカスタムオブジェクトにインデックスを付けることができます。

Amazon Kendra Salesforce データソースコネクタは、開発者版とエンタープライズ版の Salesforce エディションをサポートしています。

> **Note**
>
> Salesforce コネクター SalesforceConfiguration V1.0/API Support は 2023 年に終了する予定です。Salesforce コネクタ V2.0/ API に移行するか、使用することをお勧めします。TemplateConfiguration

Amazon Kendra Salesforce データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Salesforce データソースコネクタは次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザ ID クロール
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- エンティティ添付ファイル:クロール
- 仮想プライベートクラウド (VPC)

前提条件

を使用して Salesforce Amazon Kendra データソースをインデックス化する前に、Salesforce とアカウントでこれらの変更を行ってください。 AWS

Salesforce で、次の作業を行ったことを確認してください。

- Salesforce 管理アカウントを作成し、Salesforce への接続に使用するユーザー名とパスワードをメモすること。
- Salesforce への接続に使用されるアカウントに関連付けられた Salesforce セキュリティトークンをコピーすること。
- OAuth を有効にした Salesforce 接続アプリケーションアカウントを作成し、Salesforce 接続アプリケーションに割り当てられたコンシューマーキー (クライアント ID) とコンシューマーシークレット (クライアントシークレット) をコピーすること。詳細については、接続アプリケーションに関する Salesforce のドキュメントを参照してください。
- インデックスを作成する Salesforce インスタンスの URL をコピーすること。通常、URL は *https://<company>.salesforce.com/* です。サーバーは Salesforce 接続アプリケーションを実行している必要があります。
- ReadOnly プロファイルを複製し、[すべてのデータの表示] 権限と [記事の管理] 権限を追加して、Salesforce への読み取り専用アクセス権を持つユーザの認証情報を Salesforce サーバに追加しました。これらの認証情報は、接続を行うユーザと、接続先の Salesforce 接続アプリケーションを識別します。 Amazon Kendra
- 各ドキュメントが Salesforce および同じインデックスに使用する予定の他のデータソース間で一意であると確認すること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Salesforce の認証資格情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Salesforce IAM Secrets Manager データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Salesforce Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるように Salesforce データソースに関する必要な詳細情報を入力する必要があります。まだ Salesforce を設定していない場合は、を参照してください。 Amazon Kendra 前提条件

Console

Salesforce Amazon Kendra に接続するには:

1. AWS マネジメントコンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースを追加する] ページで [Salesforce コネクタ V2.0] を選択し、[コネクタを追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [データソース名] - データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. [デフォルト言語] - インデックスのドキュメントをフィルターする言語。特に指定しない限り、言語はデフォルトで英語に設定されます。メタデータで指定された言語は、選択した言語よりも優先されます。
   d. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. [Salesforce URL] - インデックスを作成する Salesforce サイトのインスタンス URL を入力します。
   b. 承認 — ACL があり、それをアクセス制御に使用したい場合、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
   c. 既存のシークレットを入力するか、新しいシークレットを作成すると、 AWS Secrets Manager シークレットウィンドウが開きます。
      - 認証 —「 AWS Secrets Manager シークレットを作成」ウィンドウに次の情報を入力します。
        A. [シークレット名] - シークレットの名前。シークレット名には「AmazonKendra-Salesforce-」というプレフィックスが自動的に追加されます。

B. [ユーザー名]、[パスワード]、[セキュリティトークン]、[コンシューマーキー]、[コンシューマーシークレット]、[認証 URL] には、Salesforce アカウントで生成してダウンロードした認証情報の値を入力します。

C. [認証を保存] を選択します。

d. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

e. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

f. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

g. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [添付ファイルのクロール] - 選択すると、添付されたすべての Salesforce オブジェクトがクロールされます。

b. [標準オブジェクト]、[添付ファイル付き標準オブジェクト]、[添付ファイルなしの標準オブジェクト]、[ナレッジ記事] - クロールする Salesforce エンティティまたはコンテンツタイプを選択します。

c. インデックス作成には、標準オブジェクト、ナレッジ記事、または Chatter フィードのうち少なくとも 1 つの設定情報を入力する必要があります。[ナレッジ記事] のクロール

を選択した場合は、インデックスを作成するナレッジ記事の種類を指定する必要があります。公開済み、アーカイブ済み、ドラフト、添付ファイルを選択できます。

[正規表現フィルター] - 特定のカタログアイテムを含む正規表現パターンを指定します。

8. [追加の設定]:

- [ACL 情報] デフォルトでは、すべてのアクセスコントロールリストが含まれます。アクセスコントロールリストを選択解除すると、そのカテゴリのファイルがすべて公開されます。
- [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターンを追加します。最大 100 のパターンを追加できます。

Salesforce v2 の [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更、削除済みコンテンツを同期] - 新規、変更、削除されたコンテンツのみを同期します。

[新規および変更済みコンテンツを同期] - 新規および変更されたコンテンツのみを同期します。

9. [次へ] を選択します。
10. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. 標準ナレッジ記事、標準オブジェクト添付ファイル、その他の推奨フィールドマッピングの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドの中から、インデックスにマップするフィールドを選択します。

> **Note**
>
> `_document_body` へのインデックスマッピングが必要です。`Salesforce ID` フィールドと Amazon Kendra `_document_id` フィールド間のマッピングは変更できません。任意の Salesforce フィールドをドキュメントタイトルまたはドキュメント本文 Amazon Kendra の予約済み/デフォルトインデックスフィールドにマッピングできます。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

11. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Salesforce に接続するには Amazon Kendra

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON SALESFORCEV2 スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- ホスト URL - Salesforce インスタンスのホスト URL を指定します。
- 同期モード — すべてのドキュメントを同期してインデックスを更新するか、新規、変更、Amazon Kendra 削除したドキュメントのみを同期するかを指定します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- シークレットアマゾンリソースネーム (ARN) — Salesforce Secrets Manager アカウントの認証認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "authenticationUrl": "OAUTH endpoint that Amazon Kendra connects to get an
 OAUTH token",
    "consumerKey": "Application public key generated when you created your
 Salesforce application",
```

```
    "consumerSecret": "Application private key generated when you created your
 Salesforce application",
    "password": "Password associated with the user logging in to the Salesforce
 instance",
    "securityToken": "Token associated with the user account logging in to the
 Salesforce instance",
    "username": "User name of the user logging in to the Salesforce instance"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、Salesforce コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 Amazon Kendra詳細については、「Salesforce データソースのIAM ロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 特定のドキュメント、アカウント、キャンペーン、ケース、連絡先、リード、機会、ソリューション、タスク、グループ、Chatter、カスタムエンティティファイルを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定

した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使いたい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- フィールドマッピング - 選択すると、Salesforce データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文が必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Salesforce テンプレートスキーマ」を参照してください。

詳細はこちら

Salesforce Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- 以下のための Salesforce コネクター (V2) が新しくなったことをお知らせします Amazon Kendra

ServiceNow

ServiceNow IT サービス、チケットシステム、サポートなどの組織レベルのワークフローを作成および管理するためのクラウドベースのサービス管理システムを提供します。 Amazon Kendra を使用し

て、 ServiceNow カタログ、ナレッジ記事、インシデント、およびそれらの添付ファイルのインデックスを作成できます。

Amazon Kendra コンソール、API、または TemplateConfigurationAPI Amazon Kendra ServiceNow のいずれかを使用してデータソースに接続できます。ServiceNowConfiguration

Amazon Kendra には 2 ServiceNow つのバージョンのコネクタがあります。各バージョンでサポートされる機能は次のとおりです。

ServiceNow コネクタ V1.0/ API ServiceNowConfiguration

- フィールドマッピング
- ServiceNow インスタンスバージョン:ロンドン、その他
- 包含/除外パターン: サービスカタログ、ナレッジ記事、添付ファイル

ServiceNow コネクター V2.0/ API TemplateConfiguration

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- ServiceNow インスタンスバージョン:ローマ、サンディエゴ、東京、その他
- 仮想プライベートクラウド (VPC)

> **Note**
>
> ServiceNow コネクター ServiceNowConfiguration V1.0/API Support は 2023 年に終了する予定です。 ServiceNow コネクタ V2.0/ API への移行または使用をお勧めします。 TemplateConfiguration

Amazon Kendra ServiceNow データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- ServiceNow コネクタ V1.0

- ServiceNow コネクタ V2.0
- クエリでインデックス作成するドキュメントを指定する

ServiceNow コネクタ V1.0

ServiceNow IT サービス、チケットシステム、サポートなどの組織レベルのワークフローを作成および管理するためのクラウドベースのサービス管理システムを提供します。 Amazon Kendra を使用して、 ServiceNow カタログ、ナレッジ記事、およびそれらの添付ファイルのインデックスを作成できます。

> **Note**
>
> ServiceNow コネクター ServiceNowConfiguration V1.0/API Support は 2023 年に終了する予定です。 ServiceNow コネクタ V2.0/ API への移行または使用をお勧めします。 TemplateConfiguration

Amazon Kendra ServiceNow データソースコネクタのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra ServiceNow データソースコネクタは次の機能をサポートしています。

- ServiceNow インスタンスバージョン:ロンドン、その他
- 包含/除外パターン: サービスカタログ、ナレッジ記事、およびその添付ファイル

前提条件

Amazon Kendra ServiceNow を使用してデータソースのインデックスを作成する前に、 ServiceNow AWS およびアカウントでこれらの変更を行ってください。

で ServiceNow、次のものが揃っていることを確認してください。

- ServiceNow 管理者アカウントを作成し、 ServiceNowインスタンスを作成した。
- ServiceNow インスタンス URL のホストをコピーしました。例えば、インスタンスの URL が *https://your-domain.service-now.com* の場合、入力するホスト URL の形式は *your-domain.service-now.com* です。
- Amazon Kendra ServiceNow インスタンスへの接続を許可するユーザー名とパスワードを含む基本認証認証情報を書き留めました。
- オプション:ユーザー名、パスワード、クライアント ID、 Amazon Kendra クライアントシークレットを識別して生成できる OAuth 2.0 認証トークンを設定しました。ユーザー名とパスワードは、 ServiceNow ナレッジベースとサービスカタログへのアクセスを可能にする必要があります。詳細については、OAuth 2.0 ServiceNow 認証に関するドキュメントを参照してください。
- 以下の権限を追加しました。
  - kb_category
  - kb_knowledge
  - kb_knowledge_base
  - kb_uc_cannot_read_mtom
  - kb_uc_can_read_mtom
  - sc_catalog
  - sc_category
  - sc_cat_item
  - sys_attachment
  - sys_attachment_doc
  - sys_user_role
- 各ドキュメントが、 ServiceNow 同じインデックスに使用する予定の他のデータソースと重複していないことを確認した。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。

- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- ServiceNow AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager ServiceNow データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 ServiceNow Amazon Kendra データにアクセスできるようにデータソースの必要な詳細情報を入力する必要があります。 ServiceNow まだ設定していない場合は、 ServiceNow Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには ServiceNow

1. AWS 管理コンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで、[ServiceNowコネクタ V1.0] を選択し、[データソースの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. ServiceNow ホスト — ServiceNow ホスト URL を入力します。
   b. ServiceNow バージョン — バージョンを選択します。 ServiceNow
   c. ユースケースに基づいて、[基本認証] と [Oauth 2.0 認証] のどちらかを選択します。
   d. AWS Secrets Manager secret — ServiceNow 認証情報を保存する既存のシークレットを選択するか、 Secrets Manager 新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。
      i. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraServiceNow-' がシークレット名に自動的に追加されます。
      ii. 基本認証を使用している場合は、アカウントのシークレット名、ユーザー名、パスワードを入力します。 ServiceNow

OAuth2 認証を使用する場合、アカウントで作成したシークレット名、ユーザー名、パスワード、クライアント ID、クライアントシークレットを入力します。ServiceNow

iii. [シークレットを保存して追加する] を選択します。

e. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [ナレッジ記事を含める] - ナレッジ記事にインデックスを作成することを選択します。

b. ナレッジ記事のタイプ — ユースケースに基づいて、[公開記事のみを含める] または [ServiceNow フィルタークエリに基づいて記事を含める] を選択します。[ServiceNow フィルタークエリに基づいて記事を含める] を選択した場合は、 ServiceNowアカウントからコピーしたフィルタークエリを入力する必要があります。

c. [ナレッジ記事の添付ファイルを含める] - ナレッジ記事の添付ファイルにインデックスを作成することを選択します。特定のファイルタイプを選択してインデックスを作成することもできます。

d. [カタログ項目を含める] - カタログ項目のインデックスを作成することを選択します。

e. [カタログ項目の添付ファイルを含める] - カタログ項目の添付ファイルのインデックスを作成することを選択します。特定のファイルタイプを選択してインデックスを作成することもできます。

f. 頻度 — Amazon Kendra データソースと同期する頻度。

g. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. ナレッジ記事とサービスカタログ — Amazon Kendra 生成されたデフォルトのデータソースフィールドと、その他の推奨フィールドマッピングの中から、インデックスにマッピングしたいものを選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

に接続するには Amazon Kendra ServiceNow

ServiceNowConfiguration API を使用して以下を指定する必要があります。

- データソース URL — ServiceNow URL を指定します。ホストエンドポイントは、*your-domain.service-now.com* のようになります。
- データソースホストインスタンス — ServiceNow LONDON ホストインスタンスのバージョンをまたはとして指定します。OTHERS
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 ServiceNow

  基本認証を使用している場合、シークレットは以下のキーを持つ JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password"
}
```

  OAuth 2.0 認証を使用している場合、シークレットは以下のキーを持つ JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password",
    "clientId": "client id",
    "clientSecret": "client secret"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM role — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API `RoleArn` を呼び出すタイミングを指定します。 ServiceNow Amazon Kendra詳細については、「IAM ServiceNow データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- フィールドマッピング — ServiceNow Amazon Kendra データソースフィールドをインデックスフィールドにマップすることを選択します。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

- 包含フィルターと除外フィルター - カタログとナレッジ記事の特定の添付ファイルを含めるか除外するかを指定します。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- インデックスパラメータ - 次のことを行うかどうかも指定できます。
  - ナレッジ記事とサービスカタログ、またはその両方にインデックスを作成します。ナレッジ記事とサービスカタログアイテムにインデックスを付ける場合は、 ServiceNow インデックスのインデックスドキュメントコンテンツフィールドにマップされるフィールドの名前を指定する必要があります。 Amazon Kendra
  - ナレッジ記事とカタログ項目へのインデックス添付ファイル。
  - 1 ServiceNow つ以上のナレッジベースからドキュメントを選択するクエリを使用してください。ナレッジベースは、パブリックまたはプライベートのいずれかです。詳細については、クエリでインデックス作成するドキュメントを指定するを参照してください。

詳細はこちら

Amazon Kendra ServiceNow データソースとの統合について詳しくは、以下を参照してください。

- Amazon Kendra ServiceNow オンラインコネクタ入門

ServiceNow コネクタ V2.0

ServiceNow IT サービス、チケットシステム、サポートなどの組織レベルのワークフローを作成および管理するためのクラウドベースのサービス管理システムを提供します。 Amazon Kendra を使用して、 ServiceNow カタログ、ナレッジ記事、インシデント、およびそれらの添付ファイルのインデックスを作成できます。

Amazon Kendra ServiceNow データソースコネクタのトラブルシューティングについては、を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra ServiceNow データソースコネクタは次の機能をサポートしています。

- フィールドマッピング

- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- コンテンツの完全同期と差分同期
- ServiceNow インスタンスバージョン:ローマ、サンディエゴ、東京、その他
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra ServiceNow を使用してデータソースのインデックスを作成する前に、 ServiceNow AWS およびアカウントで以下の変更を行います。

で ServiceNow、次のものが揃っていることを確認してください。

- 個人用または企業用の開発者インスタンスを作成し、 ServiceNow管理者権限を持つインスタンスを作成した。
- ServiceNow インスタンス URL のホストをコピーしました。入力するホスト URL の形式は *your-domain.service-now.com* です。 ServiceNow 接続するにはインスタンス URL が必要です Amazon Kendra。
- Amazon Kendra ServiceNow インスタンスに接続するためのユーザー名とパスワードの基本認証情報を書き留めました。
- オプション:ユーザー名、パスワード、生成されたクライアント ID、 Amazon Kendra およびクライアントシークレットを使用して識別できる OAuth 2.0 クライアント認証情報を設定しました。詳細については、OAuth 2.0 ServiceNow 認証に関するドキュメントを参照してください。
- 各ドキュメントが、 ServiceNow 同じインデックスに使用する予定の他のデータソースと重複していないことを確認した。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、次のものが揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- ServiceNow AWS Secrets Manager 認証情報をシークレットに保存し、API を使用している場合はシークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、 IAM Secrets Manager ServiceNow データソースをに接続するときにコンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Amazon Kendra データソースに接続するには、 ServiceNow Amazon Kendra データにアクセスできるようにデータソースの必要な詳細情報を入力する必要があります。 ServiceNow まだ設定していない場合は、 ServiceNow Amazon Kendra を参照してください前提条件。

Console

Amazon Kendra に接続するには ServiceNow

1. AWS 管理コンソールにサインインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. [データソースの追加] ページで [ServiceNowConnector V2.0] を選択し、[データソースの追加] を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。
   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。
   b. (オプション) [説明] - オプションで、データソースの説明を入力します。
   c. デフォルト言語-索引用のドキュメントをフィルターする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。
   d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS
   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。
   a. ServiceNow ホスト — ServiceNow ホスト URL を入力します。入力するホスト URL の形式は *your-domain.service-now.com* です。
   b. ServiceNow version — ServiceNow インスタンスのバージョンを選択します。ローマ、サンディエゴ、東京、その他から選択できます。
   c. 承認 — ACL があり、それをアクセス制御に使用したい場合は、文書のアクセス制御リスト (ACL) 情報をオンまたはオフにします。ACL は、ユーザーとグループがアクセスできるドキュメントを指定します。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
   d. 認証 — 基本認証と OAuth 2.0 認証のどちらかを選択します。
   e. AWS Secrets Manager secret — 既存のシークレットを選択するか、 Secrets Manager 認証情報を保存する新しいシークレットを作成します。 ServiceNow 新しいシークレッ

トの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。ウィンドウで、以下の情報を入力します。

i. [シークレット名] - シークレットの名前。プレフィックス 'AmazonKendraServiceNow-' がシークレット名に自動的に追加されます。

ii. 基本認証を使用している場合は、アカウントのシークレット名、ユーザー名、パスワードを入力します。 ServiceNow

OAuth2.0 認証を使用している場合 — アカウントで作成したシークレット名、ユーザー名、パスワード、クライアント ID、クライアントシークレットを入力します。 ServiceNow

iii. [シークレットを保存して追加する] を選択します。

f. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

g. ID クローラー — の ID クローラーを有効にするかどうかを指定します。 Amazon Kendra ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。

h. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

i. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [ナレッジ記事] の場合は、以下のオプションから選択してください。

- [ナレッジ記事] - ナレッジ記事にインデックスを作成することを選択します。
- [ナレッジ記事の添付ファイル] - ナレッジ記事の添付ファイルにインデックスを作成することを選択します。
- ナレッジ記事のタイプ — ユースケースに基づいて、[公開記事のみ] または [ServiceNow フィルタークエリに基づくナレッジ記事] を選択します。[ServiceNow フィルタークエリに基づく記事を含める] を選択した場合は、 ServiceNowアカウントからコピーしたフィルタークエリを入力する必要があります。フィルタークエリの例には、*workflow_state=draft^EQ*、*kb_knowledge_base=dfc19531bf2021003f07e2c*
ISNOTEMPTY^EQ、*article_type=text^active=true^EQ* などがあります。

> **Important**
>
> 公開記事のみをクロールすることを選択した場合、 Amazon Kendra 公開アクセスロールが割り当てられているナレッジ記事のみがクロールされます。
> ServiceNow

- [簡単な説明フィルターに基づいて記事を含める] - 特定の記事を含めるか除外する正規表現パターンを指定します。

b. [サービスカタログ項目] の場合:

- [サービスカタログ項目] - サービスカタログ項目のインデックスを作成することを選択します。
- [サービスカタログ項目の添付ファイル] - サービスカタログ項目の添付ファイルのインデックスを作成することを選択します。
- [アクティブなサービスカタログ項目] - アクティブなサービスカタログ項目のインデックスを作成することを選択します。
- [非アクティブなサービスカタログ項目] - 非アクティブなサービスカタログ項目のインデックスを作成することを選択します。
- フィルタクエリ — インスタンスに定義されているフィルタに基づいてサービスカタログアイテムを含めることを選択します。 ServiceNow フィルタークエリの例としては、*short_descriptionLIKEAccess^category=2809952237b1300054b6a3549dbe5* などがあります。
- [簡潔な説明フィルターに基づいてサービスカタログ項目を含める] - 特定のカタログ項目を含めるための正規表現パターンを指定します。

c. [インシデント] の場合:

- [インシデント] - サービスインシデントのインデックスを作成することを選択します。
- [インシデント添付ファイル] - インシデント添付ファイルのインデックスを作成することを選択します。
- [アクティブなインシデント] - アクティブなインシデントのインデックスを作成することを選択します。
- [非アクティブなインシデント] - 非アクティブなインシデントのインデックスを作成することを選択します。
- [アクティブなインシデントタイプ] - ユースケースに応じて、[すべてのインシデント]、[未解決のインシデント]、[オープン - 未割り当てのインシデント]、[解決済みのインシデント] から選択します。
- フィルタクエリ — インスタンスに定義されているフィルタに基づいてインシデントを含めることを選択します。 ServiceNow フィルタークエリの例には、*short_descriptionLIKETest^urgency=3^state=1^EQ*、*priority=2^categor* などがあります。
- [簡単な説明フィルターに基づいてインシデントを含める] -特定のインシデントを含む正規表現パターンを指定します。

d. [追加の設定]:

- [ACL 情報] - 選択したエンティティのアクセス制御リストがデフォルトで含まれます。アクセスコントロールリストを選択解除すると、そのカテゴリのファイルがすべて公開されます。選択されていないエンティティの ACL オプションは自動的に無効になります。パブリック記事には ACL は適用されません。
- 最大ファイルサイズの場合 — Amazon Kendra がクロールするファイルサイズの制限を MB 単位で指定します。Amazon Kendra は、定義したサイズ制限内のファイルのみをクロールします。デフォルトのファイルサイズは 50 MB です。最大ファイルサイズは 0 MB 以上 50 MB 以下でなければなりません。
- [添付ファイルの正規表現パターン] - カタログ、ナレッジ記事、インシデントの特定の添付ファイルを含めたり除外したりする正規表現パターンを追加します。最大 100 のパターンを追加できます。

e. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。 Amazon Kendra でデータソースを初めて同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更、削除済みコンテンツを同期] - 新規、変更、削除されたコンテンツのみを同期します。

f. [同期実行スケジュール] で、[頻度]-[Amazon Kendra データソースと同期する頻度] を選択します。

g. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. ナレッジ記事、サービスカタログ、添付ファイル、インシデント — Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングする項目を選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Amazon Kendra 接続するには ServiceNow

TemplateConfigurationAPI を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON SERVICENOWV2 スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- Host URL — ServiceNow ホストインスタンスのバージョンを指定します。例えば、*your-domain.service-now.com* です。
- 認証タイプ — 使用する認証のタイプ (basicAuthOAuth2 ServiceNowインスタンス用かどうか) を指定します。
- ServiceNow インスタンスバージョン —、、、など、 ServiceNow 使用するインスタンスを指定します。Tokyo Sandiego Rome Others

- 同期モード - Amazon Kendra がすべてのドキュメントを同期してインデックスを更新するか、新しいドキュメント、変更されたドキュメント、削除されたドキュメントのみを同期するかどうかを指定します。以下のいずれかから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
- シークレット Amazon リソースネーム (ARN) — Secrets Manager アカウントで作成した認証情報を含むシークレットの Amazon リソースネーム (ARN) を指定します。 ServiceNow

  基本認証を使用する場合、シークレットは以下のキーを含む JSON 構造に保存されます。

```
{
    "username": "user name",
    "password": "password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- OAuth2 クライアント認証情報を使用する場合、シークレットは以下のキーを含む JSON 構造で保存されます。

```
{
    "username": "user name",
    "password": "password",
    "clientId": "client id",
    "clientSecret": "client secret"
}
```

- IAM role — CreateDataSource IAM Secrets Manager シークレットにアクセスする権限をロールに提供し、コネクタとに必要なパブリック API RoleArn を呼び出すタイミングを指定します。 ServiceNow Amazon Kendra詳細については、「IAM ServiceNow データソースのロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - ナレッジ記事、サービスカタログ、インシデントのファイル名とファイルタイプを使用して、特定の添付ファイルを含めるか除外するかを指定できます。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- インデックスを作成する特定のドキュメント- ServiceNow クエリを使用して、プライベートナレッジベースを含む 1 つ以上のナレッジベースから必要なドキュメントを指定できます。ナレッジベースへのアクセスは、 ServiceNow インスタンスへの接続に使用するユーザーによって決まります。詳細については、クエリでインデックス作成するドキュメントを指定するを参照してください。
- インデックスパラメータ - 次のことを行うかどうかも指定できます。
  - ナレッジ記事、サービスカタログ、インシデントまたはそのすべてにインデックスを作成します。ナレッジ記事、サービスカタログアイテム、インシデントにインデックスを付ける場合は、 ServiceNow インデックスのインデックスドキュメントコンテンツフィールドにマップされるフィールドの名前を指定する必要があります。 Amazon Kendra
  - ナレッジ記事、サービスカタログ項目、インシデントへのインデックス添付ファイル。
  - short description フィルターパターンに基づいてナレッジ記事、サービスカタログ項目、インシデントを含めます。
  - アクティブと非アクティブなサービスカタログ項目とインシデントをフィルタリングすることを選択できます。
  - インシデントタイプに基づいてインシデントをフィルタリングすることを選択します。
  - ACL をクロールするエンティティを選択します。

    - ServiceNow クエリを使用して、プライベートナレッジベースを含む 1 つ以上のナレッジベースから必要なドキュメントを指定できます。ナレッジベースへのアクセスは、ServiceNowインスタンスへの接続に使用するユーザーによって決定されます。詳細については、クエリでインデックス作成するドキュメントを指定するを参照してください。
- ID クローラー — の ID Amazon Kendraクローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセス制御リスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメント用の ACL があり、その ACL を使用することを選択した場合は、 Amazon Kendraの ID クローラーを有効にして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーがオフになっていると、すべてのドキュメントをパブリックに検索できます。ID クローラーがオフになっていて、ドキュメントのアクセス制御を使用したい場合は、PutPrincipalMappingAPI を使用してユーザーおよびグループのアクセス情報をアップロードし、ユーザーコンテキストフィルタリングを行うこともできます。
- フィールドマッピング — ServiceNow データソースフィールドをインデックスフィールドにマップすることを選択します。 Amazon Kendra 詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> ドキュメントを検索するには、ドキュメント本文フィールドまたはドキュメントに対応するドキュメント本文フィールドが必要です。 Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名にマップする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「ServiceNow テンプレートスキーマ」を参照してください。

詳細はこちら

Amazon Kendra ServiceNow データソースとの統合について詳しくは、以下を参照してください。

- はじめに ServiceNow :最新コネクター (V2) Amazon Kendraのアナウンスについて Amazon Kendra

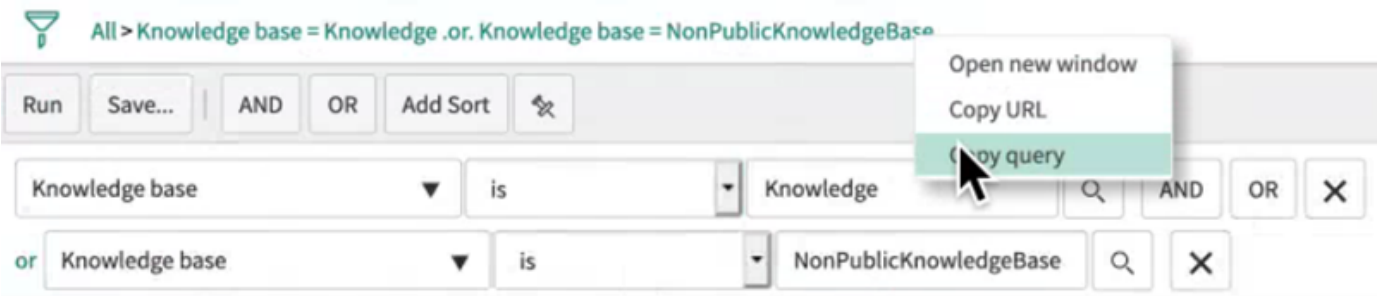
クエリでインデックス作成するドキュメントを指定する

ServiceNow クエリを使用して、 Amazon Kendra インデックスに含めたいドキュメントを指定できます。クエリを使用する場合、プライベートナレッジベースを含む複数のナレッジベースを指定できます。ナレッジベースへのアクセスは、 ServiceNow インスタンスへの接続に使用するユーザーによって決定されます。

クエリを作成するには、 ServiceNow クエリビルダーを使用します。ビルダーを使用して、クエリを作成し、クエリが正しいドキュメントのリストを返すことをテストできます。

ServiceNow コンソールを使用してクエリを作成するには

1. ServiceNow コンソールにログインします。
2. 左側のメニューで、[Knowledge] (ナレッジ)、[Article] (記事)、[All] (すべて) の順に選択します。
3. ページの上部で、フィルターのアイコンをクリックします。
4. クエリビルダーを使用してクエリを作成します。
5. クエリが完了したら、クエリを右クリックし、[Copy query] (クエリのコピー) を選択して、クエリビルダーからクエリをコピーします。 Amazon Kendraこのクエリを保存して使用してください。



クエリをコピーするときに、クエリパラメータを変更しないように注意してください。いずれかのクエリパラメータが認識されない場合、 ServiceNow そのパラメータは空として扱われ、結果のフィルタリングには使用されません。

Slack

Slack は、ユーザーがさまざまなパブリックチャネルやプライベートチャネルを通じてメッセージや添付ファイルを送信できる、エンタープライズコミュニケーションアプリです。を使用して Amazon Kendra 、Slack のパブリックチャネルとプライベートチャネル、ボットとアーカイブのメッセージ、ファイルと添付ファイル、ダイレクトメッセージとグループメッセージにインデックスを作成できます。また、フィルタリングする特定のコンテンツを選択することもできます。

> **Note**
>
> Amazon Kendra がアップグレードされた Slack コネクタをサポートするようになりました。コンソールは自動的にアップグレードされています。コンソールで作成する新しいコネクタは、アップグレードされたアーキテクチャを使用します。API を使用する場合は、TemplateConfiguration オブジェクトの代わりに `SlackConfiguration` オブジェクトを使用してコネクタを設定する必要があります。
> 古いコンソールと API アーキテクチャを使用して設定されたコネクタは、引き続き設定どおりに機能します。ただし、編集または更新することはできません。コネクタ設定を編集または更新する場合は、新しいコネクタを作成する必要があります。
> コネクタワークフローをアップグレードされたバージョンに移行することをお勧めします。古いアーキテクチャを使用して設定されたコネクタのサポートは、2024 年 6 月までに終了します。

Amazon Kendra コンソールまたは TemplateConfiguration API を使用して Slack データソース Amazon Kendra に接続できます。

Amazon Kendra Slack データソースコネクタのトラブルシューティングについては、「」を参照してくださいデータソースのトラブルシューティング。

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Slack データソースコネクタは、次の機能をサポートしています。

- フィールドマッピング
- ユーザーコンテキストフィルタリング
- ユーザー ID クロール
- 包含/除外フィルター
- フルコンテンツ同期と増分コンテンツ同期

- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用して Slack データソースのインデックスを作成する前に、Slack と AWS アカウントでこれらの変更を行ってください。

Slack で、次の作業を行ったことを確認してください。

- Slack Bot User OAuth トークンまたは Slack User OAuth トークンを作成すること。いずれかのトークンを選択して Slack データソース Amazon Kendra に接続できます。詳しくは、アクセストークンに関する Slack のドキュメントをご覧ください。

  > **Note**
  >
  > Slack 認証情報の一部としてボットトークンを使用する場合、ダイレクトメッセージやグループメッセージのインデックスを作成できないため、インデックスを作成するチャネルにボットトークンを追加する必要があります。

- Slack ワークスペースのメインページ URL にある Slack ワークスペースチーム ID をメモすること。例えば、*https://app.slack.com/client/T0123456789/...* の *T0123456789* がチーム ID です。
- 次の Oauth スコープ/読み取りアクセス許可を追加しました。

ユーザートークンの範囲	ボットトークンスコープ
• channels:history • channels:read • チャネル：書き込み • チャット：書き込み • emoji:read • files:read • files:write • groups:history • groups:read • groups:write	• team:read • channels:history • groups:history • im:history • mpim:history • チャット：書き込み • チャネル：管理 • groups:write • groups:read • im:write

ユーザートークンの範囲	ボットトークンスコープ
• im:history • im:read • im:write • mpim:history • mpim:read • mpim:write • リアクション: 書き込み • team:read • usergroups:read • usergroups:write • users.profile:read • users:read • users:read.email • 監査ログ：読み取り	• im:read • files:write • files:read • users:read • links:read • channels:read • mpim:read • chat:write.customize • チャネル: 結合 • emoji:read • mpim:write • usergroups:read • usergroups:write • users.profile:read • users:read.email • ユーザー: 書き込み

- 各ドキュメントが Slack および同じインデックスに使用する予定の他のデータソース間で一意であると確認すること。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

で AWS アカウント、以下があることを確認します。

- Amazon Kendra インデックスを作成し、 API を使用している場合はインデックス ID を記録しました。
- データソースの IAM ロールを作成し、 API を使用している場合は、 IAM ロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更する場合は、 IAM ロールを更新して正しい AWS Secrets Manager シークレット ID にアクセスする必要があります。

- Slack の認証資格情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録済み。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

既存の IAM ロールまたはシークレットがない場合は、Slack データソースを に接続するときに、コンソールを使用して新しい IAM ロールとシーク Secrets Manager レットを作成できます Amazon Kendra。API を使用している場合は、既存の IAM ロールと Secrets Manager シークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Slack データソース Amazon Kendra に接続するには、 がデータ Amazon Kendra にアクセスできるように Slack データソースの必要な詳細を入力する必要があります。の Slack をまだ設定していない場合は Amazon Kendra、「」を参照してください前提条件。

Console

Slack Amazon Kendra に接続するには

1. にサインイン AWS Management Console し、 Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

> **Note**
>
> [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. データソースの追加ページで、Slack コネクタ を選択し、コネクタ を追加 を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

   a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

   b. (オプション) [説明] - オプションで、データソースの説明を入力します。

   c. デフォルト言語 - インデックスのドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

   d. タグ で、新しいタグを追加 - 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するためのオプションのタグを含めます。

   e. [次へ] を選択します。
6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

   a. ソース の Slack ワークスペースチーム ID の場合 — Slack ワークスペースのチーム ID。

   b. AWS Secrets Manager secret — 既存のシークレットを選択するか、新しい Secrets Manager シークレットを作成して Slack 認証情報を保存します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

      i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

         A. [シークレット名] - シークレットの名前。シークレット名には、プレフィックス AmazonKendra「-Slack-」が自動的に追加されます。

         B. [Slack トークン] - Slack アカウントで作成した認証情報の値を入力します。

      ii. [保存] を選択します。

   c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。

e. IAM role — 既存の IAM ロールを選択するか、リポジトリの認証情報とインデックスコンテンツにアクセスするための新しい IAM ロールを作成します。

> **Note**
>
> IAM インデックスに使用される ロールは、データソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

f. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

a. [クロールするコンテンツタイプを選択] - クロールする Slack エンティティまたはコンテンツタイプ。すべてのチャネル 、パブリックチャネル 、プライベートチャネル 、グループメッセージ 、プライベートメッセージ から選択できます。

b. クロール開始日の選択 — Slack コンテンツをクロール Amazon Kendra する日付を入力します。

c. 追加設定 – オプションで、次の情報を入力します。

- （ オプション) チャンネル ID/名前 — チャンネルからコンテンツを同期するように選択した場合は、チャンネル IDs とチャンネル名を指定して、特定のチャンネルから同期するコンテンツを含めることができます。
- メッセージ - ボットメッセージ、アーカイブされたメッセージ、またはボットとアーカイブされたメッセージの両方を含めるかどうかを選択します。

> **Note**
>
> チャネル ID とチャネル名の両方にフィルターを設定する場合、 Amazon Kendra Slack コネクタはチャネル名よりもチャネル IDs を優先します。チャネル ID またはチャネル名 のいずれかにフィルターを設定すると、同期スコープ でプライベートメッセージとグループメッセージをクロールするように選択した場合でも、 Amazon Kendra Slack コネクタはプライベートメッセージとグループメッセージを無視します。

- [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。正規表現パターンの例は次のとおりです。
  - ファイルタイプ – .pdf、.docx
  - ファイル名 – Hello*.txt、 TestFile.*

d. [同期モード] では、データソースのコンテンツが変更されたときのインデックスの更新方法を選択します。データソースを Amazon Kendra 初めて と同期すると、デフォルトですべてのコンテンツが同期されます。

- [完全同期] - 前回の同期ステータスに関係なく、すべてのコンテンツを同期します。
- [新規、変更済み、または削除されたドキュメントを同期] - 新規、変更済み、または削除されたドキュメントのみを同期します。

e. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度。

f. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

a. Slack フィールドマッピングの場合 — インデックスにマッピングする Amazon Kendra デフォルトのデータソースフィールドから選択します。

b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Slack Amazon Kendra に接続するには

TemplateConfiguration API を使用してデータソーススキーマの JSON を指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — JSON スキーマを使用するSLACKときに、データソースタイプを TemplateConfiguration として指定します。また、 CreateDataSource API を呼び出すTEMPLATEときにデータソースを として指定します。
- Slack ワークスペースチーム ID - Slack のメインページ URL からコピーした Slack チーム ID。
- 日付から — Slack ワークスペースチームからのデータのクロールを開始する日付。日付は次の形式に従う必要があります yyyy-mm-dd。
- 同期モード - すべてのドキュメントを同期するか、新規、変更、削除されたドキュメントのみを同期するかを指定して、インデックス Amazon Kendra を更新します。以下のオプションから選択できます。
  - FORCED_FULL_CRAWL は、データソースがインデックスと同期されるたびに、すべてのコンテンツを新たに再クロールし、既存のコンテンツを置き換えます。
  - FULL_CRAWL は、データソースがインデックスと同期されるたびに、新しいコンテンツ、変更されたコンテンツ、削除されたコンテンツのみを段階的にクロールします。
  - CHANGE_LOG は、データソースがインデックスと同期されるたびに、新しいコンテンツと変更されたコンテンツのみを段階的にクロールします。
- ID クローラー — Amazon Kendraの ID クローラーを有効にするかどうかを指定します。ID クローラーは、ドキュメントのアクセスコントロールリスト (ACL) 情報を使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて検索結果をフィルタリングします。ドキュメントの ACL があり、ACL の使用を選択した場合は、 Amazon Kendraの ID クローラーをオンにして、検索結果のユーザーコンテキストフィルタリングを設定することもできます。それ以外の場合、ID クローラーをオフにすると、すべてのドキュメントをパブリックに検索できます。ドキュメントのアクセスコントロールを使用し、ID クローラーがオフになっている場合は、 PutPrincipalMapping API を使用してユーザーコンテキストフィルタリング用のユーザーおよびグループのアクセス情報をアップロードすることもできます。
- シークレット Amazon リソースネーム (ARN) - Slack アカウントの認証情報を含む Secrets Manager シークレットの Amazon リソースネーム (ARN) を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
```

```
    "slackToken": "token"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- role IAM — を呼び出しCreateDataSourceて、シー Secrets Manager クレットにアクセスするためのアクセス許可を IAM ロールに付与し、Slack コネクタと に必要なパブリック APIs を呼び出すRoleArnタイミングを指定します Amazon Kendra。詳細については、「Slack データソースの IAM ロール」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - VpcConfiguration で CreateDataSource を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 特定のチャネル — パブリックチャネルまたはプライベートチャネルでフィルタリングし、ID で特定のチャネルを指定します。
- チャネルとメッセージのタイプ — がパブリックチャネルとプライベートチャネル、グループメッセージとダイレクトメッセージ、ボットメッセージとアーカイブメッセージのどちらをインデックス Amazon Kendra する必要があるか。Slack 認証情報の一部としてボットトークンを使用する場合、インデックスを作成するチャネルにボットトークンを追加する必要があります。ボットトークンを使用してダイレクトメッセージやグループメッセージのインデックスを作成することはできません。
- さかのぼって - Slack コネクタが、前回のコネクタ同期前であれば、更新または削除されたコンテンツをクロールするように lookBackパラメータを設定できます。
- 包含フィルターと除外フィルター — 特定の Slack コンテンツを含めるか除外するかを指定します。Slack 認証情報の一部としてボットトークンを使用する場合、インデックスを作成するチャネルにボットトークンを追加する必要があります。ボットトークンを使用してダイレクトメッセージやグループメッセージのインデックスを作成することはできません。

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- フィールドマッピング - 選択すると、Slack データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> がドキュメント Amazon Kendra を検索するには、ドキュメント本文フィールドまたはドキュメントと同等のドキュメント本文が必要です。データソース内のドキュメント本文フィールド名をインデックスフィールド名 にマッピングする必要があります`_document_body`。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについては、「Slack template schema」を参照してください。

詳細はこちら

Slack データソース Amazon Kendra との統合の詳細については、以下を参照してください。

- Unravel the knowledge in Slack workspaces with intelligent search using the Amazon Kendra Slack connector

Zendesk

Zendesk は、企業がカスタマーサポートのやり取りを自動化および強化するのに役立つ顧客関係管理システムです。 Amazon Kendra を使用して、Zendesk Supportチケット、チケットコメント、チケット添付ファイル、ヘルプセンター記事、記事コメント、記事コメント添付ファイル、ガイドコ

ミュニティトピック、コミュニティ投稿、コミュニティ投稿コメントのインデックスを作成できます。

特定の組織内のチケットのみにインデックスを付ける場合は、組織名でフィルタリングできます。Zendesk からのデータのクローリングを開始するタイミングに合わせて、クロール日を設定することもできます。

Amazon Kendra Amazon Kendra コンソールとAPIを使用してZendeskデータソースに接続できます。TemplateConfiguration

Amazon Kendra Zendesk データソースコネクターのトラブルシューティングについては、を参照してください。データソースのトラブルシューティング

トピック

- サポートされている機能
- 前提条件
- 接続手順
- 詳細はこちら

サポートされている機能

Amazon Kendra Zendesk データソースコネクターは以下の機能をサポートしています。

- 変更ログ
- フィールドマッピング
- ユーザーコンテキストフィルタリング
- 包含/除外フィルター
- 仮想プライベートクラウド (VPC)

前提条件

Amazon Kendra を使用してZendeskデータソースのインデックスを作成する前に、Zendeskとアカウントでこれらの変更を行ってください。 AWS

Zendesk で以下を確認してください。

- Zendesk Suite (Professional/Enterprise) の管理者アカウントを作成しました。

- Zendesk のホスト URL を書き留めました。例えば、*https://{sub-domain (https://{host/)}.zendesk.com/* です。

> **Note**
>
> （オンプレミス/サーバー）AWS Secrets Manager は、に含まれるエンドポイント情報が、 Amazon Kendra データソース設定の詳細で指定されているエンドポイント情報と同じかどうかを確認します。混乱する代理問題は、ユーザーがアクションを実行するアクセス許可がないにもかかわらず、 Amazon Kendra をプロキシとして使用して設定された秘密にアクセスし、アクションを実行するセキュリティの問題です。後でエンドポイント情報を変更する場合は、新しいシークレットを作成してこの情報を同期する必要があります。

- クライアント ID、クライアントシークレット、ユーザー名、パスワードを含む OAuth 2.0 認証情報トークンを生成しました。詳細については、OAuth 2.0 トークンの生成に関する Zendesk のドキュメントを参照してください。
- 次の OAuth 2.0 スコープを追加しました。
  - read
- オプション: Amazon Kendra による接続を許可する SSL 証明書をインストールしました。
- 各ドキュメントが Zendesk および同じインデックスを使用予定の他のデータソース間で一意であることを確認しました。インデックスに使用する各データソースには、データソース全体に同じドキュメントが含まれていてはなりません。ドキュメント ID はインデックス全体に適用され、インデックスごとに一意である必要があります。

には AWS アカウント、以下の情報が揃っていることを確認してください。

- Amazon Kendra インデックスを作成し、API を使用している場合はインデックス ID を書き留めました。
- IAM データソース用のロールを作成し、API IAM を使用している場合はロールの ARN を記録しました。

> **Note**
>
> 認証タイプと認証情報を変更した場合、 AWS Secrets Manager 正しいシークレット ID IAM にアクセスできるようにロールを更新する必要があります。

- Zendesk の認証情報を AWS Secrets Manager シークレットに保存し、API を使用している場合は、シークレットの ARN を記録しました。

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

IAM 既存のロールやシークレットがない場合は、Zendeskデータソースをに接続するときに、 IAM Secrets Manager コンソールを使用して新しいロールとシークレットを作成できます。 Amazon Kendra API を使用している場合は、 IAM Secrets Manager 既存のロールとシークレットの ARN、およびインデックス ID を指定する必要があります。

接続手順

Zendesk Amazon Kendra データソースに接続するには、 Amazon Kendra データにアクセスできるようにZendesk データソースに関する必要な詳細情報を入力する必要があります。Zendesk をまだ設定していない場合は、を参照してください。 Amazon Kendra前提条件

Console

Zendesk Amazon Kendra に接続するには

1. AWS Management Console にログインし、Amazon Kendra コンソールを開きます。
2. 左側のナビゲーションペインで、[インデックス] を選択し、インデックスのリストから使用するインデックスを選択します。

   > **Note**
   >
   > [インデックスの設定] で、[ユーザーアクセスコントロール] 設定を設定または編集できます。

3. [使用開始] ページで、[データソースを追加] を選択します。
4. 「データソースの追加」ページで「Zendesk コネクター」を選択し、「コネクターを追加」を選択します。
5. [データソースの詳細を指定] ページで、次の情報を入力します。

a. [名前と説明] の [データソース名] に、データソースの名前を入力します。ハイフン (-) は使用できますが、スペースは使用できません。

b. (オプション) [説明] - オプションで、データソースの説明を入力します。

c. デフォルト言語 — インデックスの対象となるドキュメントをフィルタリングする言語を選択します。特に指定しない限り、言語はデフォルトで英語に設定されます。ドキュメントのメタデータで指定された言語は、選択した言語よりも優先されます。

d. [タグ] の [新しいタグの追加] — リソースの検索、絞り込み、コストの追跡を行うためのオプションタグを追加します。 AWS

e. [次へ] を選択します。

6. [アクセスとセキュリティの定義] ページで、次の情報を入力します。

a. Zendesk URL - Zendesk URL を入力します。

b. AWS Secrets Manager secret — Secrets Manager Zendeskの認証情報を保存する既存のシークレットを選択するか、新しいシークレットを作成します。新しいシークレットの作成を選択すると、 AWS Secrets Manager シークレットウィンドウが開きます。

i. [AWS Secrets Manager シークレットウィンドウを作成] に次の情報を入力します。

A. [シークレット名] - シークレットの名前。シークレット名には「AmazonKendra-Zendesk」というプレフィックスが自動的に追加されます。

B. [クライアント ID]、[クライアントシークレット]、[ユーザー名]、[パスワード] には、Zendesk アカウントで作成した認証情報の値を入力します。

ii. [保存] を選択します。

c. [仮想プライベートクラウド (VPC)] - VPC の使用を選択できます。選択する場合は、[サブネット] と [VPC セキュリティグループ] を追加する必要があります。

d. IAM ロール — 既存のロールを選択するか、 IAM IAM 新しいロールを作成して、リポジトリの認証情報とインデックスコンテンツにアクセスします。

> **Note**
>
> IAM インデックスに使用されるロールはデータソースには使用できません。インデックスやよくある質問に既存のロールが使用されているかどうかが不明な場合は、エラーを避けるため、[新しいロールを作成] を選択してください。

e. [次へ] を選択します。

7. [同期設定の構成] ページで、次の情報を入力します。

   a. [エンティティまたはコンテンツタイプを選択] - クロールする Zendesk エンティティまたはコンテンツタイプ。

   b. [変更ログ] - 選択すると、すべてのファイルを同期する代わりにインデックスを更新できます。

   c. [組織名] — Zendesk の組織名を入力して同期をフィルタリングします。

   d. [同期開始日] - コンテンツのインデックス作成を開始する日。

   e. [正規表現パターン] - 特定のファイルを含めるまたは除外する正規表現パターン。最大 100 のパターンを追加できます。

   f. [同期実行スケジュール] の [頻度] - Amazon Kendra がデータソースと同期する頻度を選択します。

   g. [次へ] を選択します。

8. [フィールドマッピングを設定] ページで、次の情報を入力します。

   a. チケット、チケットコメント、チケットコメント添付ファイル、記事、記事コメント、記事コメント、記事コメント添付ファイル、コミュニティトピック、コミュニティ投稿、コミュニティ投稿コメントの場合- Amazon Kendra 生成されたデフォルトのデータソースフィールドから、インデックスにマッピングする項目を選択します。

   b. [フィールドを追加] - カスタムデータソースフィールドを追加して、マッピング先のインデックスフィールド名とフィールドデータタイプを作成します。

   c. [次へ] を選択します。

9. [確認と作成] ページで、入力した情報が正しいことを確認し、[データソースを追加] を選択します。このページで情報の編集を選択することもできます。データソースが正常に追加されると、データソースが [データソース] ページに表示されます。

API

Zendesk Amazon Kendra に接続するには

TemplateConfigurationAPIを使用してデータソーススキーマのJSONを指定する必要があります。これには、以下の情報を入力する必要があります。

- データソース — TemplateConfigurationJSON ZENDESK スキーマを使用する場合と同様に、データソースタイプを指定します。また、CreateDataSourceAPI TEMPLATE を呼び出すときと同じようにデータソースを指定します。
- ホスト URL - 接続設定またはリポジトリエンドポイントの詳細の一部として Zendesk ホスト URL を提供します。例えば、 *https://yoursubdomain.zendesk.com* です。
- 変更ログ — Zendeskのデータソース変更ログメカニズムを使用して、インデックス内のドキュメントを更新する必要があるかどうかを判断するかどうか Amazon Kendra 。

> **Note**
>
> Amazon Kendra にすべてのドキュメントをスキャンさせない場合は、変更ログを使用します。変更ログが大きい場合は、 Amazon Kendra 変更ログを処理するよりもZendeskデータソース内のドキュメントをスキャンするほうが時間がかからない場合があります。Zendesk データソースをインデックスに初めて同期する場合は、すべてのドキュメントがスキャンされます。

- シークレットAmazon リソースネーム（ ARN ）—Amazon Secrets Manager endeskアカウントの認証情報を含むシークレットのアマゾンリソースネーム（ ARN ）を指定します。シークレットは、次のキーを含む JSON 構造に保存されます。

```
{
    "hostUrl": "https://yoursubdomain.zendesk.com",
    "clientId": "client ID",
    "clientSecret": "Zendesk client secret",
    "userName": "Zendesk user name",
    "password": "Zendesk password"
}
```

> **Note**
>
> 認証情報とシークレットは、定期的に更新またはローテーションすることをお勧めします。セキュリティに必要なアクセスレベルのみを提供してください。認証情報とシークレットを、データソース、コネクタバージョン 1.0 と 2.0 (該当する場合) で再利用することは推奨しません。

- IAM ロール — `CreateDataSource` IAM Secrets Manager シークレットにアクセスする権限をロールに付与したり、Zendesk `RoleArn` Connectorとに必要なパブリックAPIを呼び出したり

するタイミングを指定します。 Amazon Kendra詳細については、「IAM roles for Zendesk data sources」を参照してください。

オプションで、次の機能を追加することもできます。

- [仮想プライベートクラウド (VPC)] - `VpcConfiguration` で `CreateDataSource` を呼び出すタイミングを指定します。詳細については、「を使用する Amazon Kendra ための の設定 Amazon VPC」を参照してください。
- 包含フィルターと除外フィルター - 以下を含めるか除外するかを指定します。
  - サポートチケット、チケットコメント、チケットコメントの添付ファイル
  - ヘルプセンターの記事、記事の添付ファイル、記事のコメント
  - コミュニティトピック、投稿、投稿コメントのガイド

> **Note**
>
> ほとんどのデータソースは、フィルターと呼ばれる包含または除外パターンである正規表現パターンを使用しています。包含フィルターを指定すると、包含フィルターに一致するコンテンツのみのインデックスが作成されます。包含フィルターに一致しないドキュメントのインデックスは作成されません。包含フィルターと除外フィルターを指定した場合、除外フィルターに一致するドキュメントは、包含フィルターと一致してもインデックスは作成されません。

- ユーザーコンテキストフィルタリングとアクセス制御 — ドキュメント用のACLがある場合、ドキュメントのアクセス制御リスト (ACL)Amazon Kendra をクロールします。ACL 情報は、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングするために使用されます。詳細については、「User context filtering」を参照してください。
- フィールドマッピング - 選択すると、Zendesk データソースフィールドを Amazon Kendra インデックスフィールドにマッピングします。詳細については、データソースフィールドのマッピングを参照してください。

> **Note**
>
> 文書を検索するには、文書本文フィールドまたは文書に対応する文書本文が必要です。Amazon Kendra データソース内の文書本文フィールド名をインデックスフィールド名

にマップする必要があります_document_body。その他のすべてのフィールドはオプションです。

設定が必要なその他の重要な JSON キーのリストについての詳細は、「Zendesk テンプレートスキーマ」を参照してください。

詳細はこちら

Zendesk Amazon Kendra データソースとの統合について詳しくは、以下を参照してください。

- インテリジェント検索でZendeskのインサイトを見つけましょう。 Amazon Kendra

データソースフィールドのマッピング

Amazon Kendra データソースコネクタは、データソースのドキュメントフィールドまたはコンテンツフィールドを Amazon Kendra インデックスのフィールドにマッピングできます。デフォルトでは、各コネクタは特定のデータソースフィールドをクロールするように設計されています。既定のデータソースフィールドとそのプロパティは変更またはカスタマイズできません。 Amazon Kendra コンソールでは、編集できないデフォルトのフィールドとデフォルトのフィールドプロパティはグレー表示されます。

Amazon Kendra コネクタを使用すると、データソースのカスタムドキュメントまたはコンテンツフィールドをインデックスのカスタムフィールドにマッピングすることもできます。例えば、データソースにドキュメントの部門情報を含む「dept」というフィールドがある場合、それを「Department」というインデックスフィールドにマッピングできます。そうすれば、ドキュメントをクエリするときにフィールドを使用できます。

などの Amazon Kendra 予約済みフィールドや共通フィールドをマッピングすることもできます_created_at。データソースに「creation_date」というフィールドがある場合は、これを という同等の Amazon Kendra 予約フィールドにマッピングできます_created_at。 Amazon Kendra 予約済みフィールドの詳細については、「ドキュメント属性またはフィールド」を参照してください。

ほとんどのデータソースのフィールドをマッピングできます。次のデータソースのフィールドマッピングを作成できます。

- Adobe Experience Manager
- Alfresco

- Aurora (MySQL)
- Aurora (PostgreSQL)
- Amazon FSx (Windows)
- Amazon FSx （ NetApp ONTAP)
- Amazon RDS/Aurora
- Amazon RDS (Microsoft SQL Server)
- Amazon RDS (MySQL)
- Amazon RDS (Oracle)
- Amazon RDS (PostgreSQL)
- Amazon Kendra ウェブクローラー
- Amazon WorkDocs
- [Box] (ボックス)
- Confluence
- Dropbox
- Drupal
- GitHub
- Google Workspace ドライブ
- Gmail
- IBM DB2
- Jira
- Microsoft Exchange
- Microsoft OneDrive
- Microsoft SharePoint
- Microsoft Teams
- Microsoft SQL Server
- Microsoft Yammer
- MySQL
- Oracle Database
- PostgreSQL

- Quip
- Salesforce
- ServiceNow
- Slack
- Zendesk

S3 バケットまたは S3 データソースにドキュメントを保存する場合、JSON メタデータファイルを使用してフィールドを指定します。詳細については、「S3 data source connector」を参照してください。

データソースフィールドをインデックスフィールドにマッピングするには、次の 3 つの手順を実行します。

1. インデックスを作成します。詳細については、インデックスの作成を参照してください。
2. インデックスを更新して、フィールドを追加します。
3. データソースを作成し、フィールドマッピングを含めて、予約済みフィールドとカスタムフィールドを Amazon Kendra インデックスフィールドにマッピングします。

インデックスを更新してカスタムフィールドを追加するには、 コンソールを使用してデータソースフィールドマッピングを編集し、カスタムフィールドを追加するか、 UpdateIndex API を使用します。合計 500 のカスタムフィールドをインデックスに追加できます。

データベースデータソースの場合、データベース列の名前が予約フィールドの名前と一致する場合、フィールドと列は自動的にマッピングされます。

UpdateIndex API では、 を使用して予約済みフィールドとカスタムフィールドを追加しますDocumentMetadataConfigurationUpdates。

次の JSON の例では、DocumentMetadataConfigurationUpdates を使用して「Department」というフィールドをインデックスに追加します。

```
"DocumentmetadataConfigurationUpdates": [
   {
       "Name": "Department",
       "Type": "STRING_VALUE"
   }
]
```

フィールドを作成するときに、検索でのフィールドの使用方法を設定するオプションがあります。次から選択できます。

- 表示可能 - クエリレスポンスでフィールドを返すかどうかを指定します。デフォルトは `true` です。
- ファセット可能 - フィールドを使用してファセットを作成できることを示します。デフォルトは `false` です。
- 検索可能 - 検索でフィールドを使用するかどうかを指定します。デフォルトは、文字列フィールドに対しては `true`、数値フィールドと日付フィールドに対しては `false` です。
- ソート可能 - フィールドを使用して検索結果をソートできることを示します。日付、数値、および文字列フィールドに対してのみ設定できます。文字列リストフィールドには設定できません。

次の JSON 例では、`DocumentMetadataConfigurationUpdates` を使用して「Department」というフィールドをインデックスに追加し、それをファセット可能としてマークします。

```
"DocumentMetadataConfigurationUpdates": [
   {
       "Name": "Department",
       "Type": "STRING_VALUE",
       "Search": {
           "Facetable": true
       }
   }
]
```

Amazon Kendra 予約済みまたは共通のドキュメントフィールドの使用

UpdateIndex API を使用すると、 を使用して予約済みフィールドまたは共通フィールドを作成し`DocumentMetadataConfigurationUpdates`、 Amazon Kendra 予約済みインデックスフィールド名を指定して、同等のドキュメント属性/フィールド名にマッピングできます。カスタムフィールドも作成できます。データソースコネクタを使用する場合、データソースドキュメントフィールドを Amazon Kendra インデックスフィールドにマッピングするフィールドマッピングがほとんど含まれます。コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。

`Search` オブジェクトを設定して、フィールドを表示可能、ファセット可能、検索可能、ソート可能のいずれかに設定できます。特定のフィールド値にマッピングされたブースト、新しさ、重要度の値に適用するフィールドのランク順序、ブースト期間、または期間を設定するように `Relevance` オブ

ジェクトを設定できます。コンソールを使用する場合は、ナビゲーションメニューのファセットオプションを選択して、フィールドの検索設定をセットできます。関連性調整を設定するには、ナビゲーションメニューでインデックスを検索するオプションを選択し、クエリを入力し、サイドパネルのオプションを使用して検索の関連性を調整します。フィールドを作成すると、フィールドタイプを変更することはできません。

Amazon Kendra には、次の予約済みまたは共通のドキュメントフィールドがあり、使用できます。

- _authors - ドキュメントの内容を担当する 1 人以上の作成者のリスト。
- _category - ドキュメントを特定のグループに配置するカテゴリ。
- _created_at - ドキュメントが作成された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _data_source_id - ドキュメントを含むデータソースの識別子。
- _document_body - ドキュメントのコンテンツ。
- _document_id - ドキュメントの一意の識別子。
- _document_title - ドキュメントのタイトル。
- _excerpt_page_number - ドキュメントの抜粋が表示される PDF ファイルのページ番号。2020 年 9 月 8 日より前にインデックスが作成された場合、この属性を使用する前に、ドキュメントのインデックスを再作成する必要があります。
- _faq_id - これが質疑応答タイプのドキュメント (よくある質問) の場合、よくある質問の固有識別子です。
- _file_type - pdf や doc など、ドキュメントのファイルタイプ。
- _last_updated_at - ドキュメントが最後に更新された ISO 8601 形式の日付と時刻。例えば、2012-03-25T12:30:10+01:00 は、中央ヨーロッパ時間の 2012 年 3 月 25 日午後 12 時 30 分 (プラス 10 秒) の ISO 8601 の日付/時刻形式です。
- _source_uri - ドキュメントが利用可能な URI。例えば、会社のウェブサイト上のドキュメントの URI などです。
- _version - ドキュメントの特定のバージョンの識別子。
- _view_count - ドキュメントが表示された回数。
- _language_code (文字列) - ドキュメントに適用される言語のコード。言語を指定しないと、デフォルトで英語になります。コードを含む、サポートされている言語の詳細については、英語以外の言語でドキュメントを追加するを参照してください。

カスタムフィールドの場合、予約フィールドまたは共通フィールドを作成する場合と同じように、UpdateIndex API で DocumentMetadataConfigurationUpdates を使用してこれらのフィールドを作成します。カスタムフィールドには適切なデータタイプを設定する必要があります。コンソールを使用する場合は、データソースを選択し、編集アクションを選択してから、フィールドマッピングセクションの横に進んでデータソースを設定して、フィールドを更新します。一部のデータソースは、新しいフィールドやカスタムフィールドの追加をサポートしていません。フィールドを作成すると、フィールドタイプを変更することはできません。

カスタムフィールドには以下のタイプを設定できます。

- 日付
- 数
- 文字列
- 文字列リスト

BatchPutDocument API を使用してインデックスにドキュメントを追加した場合、 はドキュメントのフィールド/属性をAttributes一覧表示し、 DocumentAttribute オブジェクトを使用してフィールドを作成します。

Amazon S3 データソースからインデックス作成されたドキュメントの場合、フィールド情報を含む JSON メタデータファイルを使用してフィールドを作成します。

サポートされているデータベースをデータソースとして使用する場合は、フィールドマッピングオプションを使用してフィールドを設定できます。

英語以外の言語でドキュメントを追加する

ドキュメントは、複数の言語でインデックス作成できます。言語を指定しない場合、 Amazon Kendra はデフォルトで英語でドキュメントをインデックス作成します。ドキュメントの言語コードをドキュメントメタデータにフィールドとして含めます。ドキュメントの _language_code フィールドの詳細については、「フィールドマッピング」と「カスタム属性」を参照してください。

を呼び出すときに、データソース内のすべてのドキュメントの言語コードを指定できますCreateDataSource。ドキュメントにメタデータフィールドで指定された言語コードがない場合、データソースレベルですべてのドキュメントに指定された言語コードを使用して、ドキュメントのインデックスが作成されます。コンソールでは、データソースレベルでのみ、サポートされている言語でドキュメントのインデックス作成ができます。[Data sources] (データソース) へ移動し、[Specify

data source details] (データソースの詳細を指定) ページで、[Language] (言語) のドロップダウンから言語を選択します。

サポートされている言語でドキュメントを検索またはクエリできます。詳細については、「各言語での検索」を参照してください。

以下の言語とそのコードがサポートされています (言語を指定しない場合、英語または en はデフォルトでサポートされています。) この表には、フルセマンティック検索で が Amazon Kendra サポートする言語と、単純なキーワード一致のみをサポートする言語が含まれています。次の表では、完全なセマンティック検索をサポートする言語にはアスタリスクが付いており、太字で示されています。英語 (デフォルト言語) は完全セマンティック検索でもサポートされています。

言語名	言語コード
アラビア語	ar
アルメニア語	hy
バスク語	eu
ベンガル語	bn
ブルガリア語	bg
カタロニア語	ca
中国語 - 簡体字と繁体字*	zh
チェコ語	cs
デンマーク語	da
オランダ語	nl
フィンランド語	fi
フランス語 - フランス語 (カナダ) を含む*	fr
ガリシア語	gl
ドイツ語*	de

言語名	言語コード
ギリシャ語	el
ヒンディー語	hi
ハンガリー語	hu
インドネシア語	id
アイルランド語	ga
イタリア語	it
日本語*	ja
韓国語*	ko
ラトビア語	lv
リトアニア語	lt
ノルウェー語	no
ペルシャ語	fa
ポルトガル語	pt
ポルトガル語 (ブラジル)*	pt-BR
ルーマニア語	ro
ロシア語	ru
ソラニ語	ckb
スペイン語 - スペイン語 (メキシコ) を含む*	es
スウェーデン語	sv
トルコ語	tr

*その言語ではセマンティック検索がサポートされています。

セマンティック検索をサポートする言語では、以下の機能がサポートされます。

- 単純なキーワードマッチングを超えるドキュメントの関連性。
- 単純なキーワードマッチング以外のよくある質問。
- Amazon Kendraの読み取り理解度に基づくドキュメントからの回答の抽出。
- 検索結果の信頼バケット (非常に高い、高い、中程度、低いなど) です。

セマンティック検索をサポートしていない言語では、ドキュメントの関連性やよくある質問に関する単純なキーワードマッチングがサポートされています。

シノニム (カスタムシノニムを含む)、増分学習とフィードバック、およびクエリの提案は、英語 (デフォルト言語) でのみサポートされています。

を使用する Amazon Kendra ための の設定 Amazon VPC

Amazon Kendra は、 で作成した Virtual Private Cloud (VPC) に接続 Amazon Virtual Private Cloud して、プライベートクラウドで実行されているデータソースに保存されているコンテンツのインデックスを作成できます。データソースコネクタを作成するときに、データソースを含むサブネットのセキュリティグループとサブネット識別子を指定できます。この情報を使用して、 は VPC 内のデータソースと安全に通信するために使用する Elastic Network Interface Amazon Kendra を作成します。

Amazon Kendra データソースコネクタを でセットアップするには Amazon VPC、 AWS Management Console または CreateDataSource API オペレーションを使用できます。コンソールを使用する場合は、コネクタ設定プロセス中に VPC を接続します。

> **Note**
>
> データソースコネクタを設定する場合、 Amazon VPC Amazon Kendra この機能はオプションです。パブリックインターネットからデータソースにアクセスできる場合は、 Amazon VPC この機能を有効にする必要はありません。すべての Amazon Kendra データソースコネクタが をサポートしているわけではありません Amazon VPC。

データソースが で実行されておらず Amazon VPC 、パブリックインターネットからアクセスできない場合は、まず仮想プライベートネットワーク (VPN) を使用してデータソースを VPC に接続し

ます。その後、 Amazon VPC と を組み合わせて Amazon Kendra データソースを に接続できます AWS Virtual Private Network。VPN の設定については、「」のAWS VPN ドキュメントを参照してください。

トピック

- Amazon Kendra コネクタ Amazon VPC のサポートの設定
- に接続するための Amazon Kendra データソースを設定する Amazon VPC
- VPC のデータベースに接続する
- VPC 接続の問題のトラブルシューティング

Amazon Kendra コネクタ Amazon VPC のサポートの設定

Amazon Kendra コネクタで使用する Amazon VPC ように を設定するには、次のステップを実行します。

ステップ

- ステップ 1。の Amazon VPC サブネットを作成する Amazon Kendra
- ステップ 2。のセキュリティ Amazon VPC グループの作成 Amazon Kendra
- ステップ 3。外部データソースと を設定する Amazon VPC

ステップ 1。の Amazon VPC サブネットを作成する Amazon Kendra

がデータソースへのアクセスに使用できる既存の Amazon VPC サブネット Amazon Kendra を作成または選択します。準備済みサブネットは、次のいずれかの AWS リージョン およびアベイラビリティーゾーンにある必要があります。

- 米国西部 (オレゴン)/us-west-2-usw2-az1, usw2-az2, usw2-az3
- 米国東部 (バージニア北部)/us-east-1-use1-az1, use1-az2, use1-az4
- 米国東部 (オハイオ)/us-east-2-use2-az1, use2-az2, use2-az3
- アジアパシフィック (東京)/ap-northeast-1-apne1-az1, apne1-az2, apne1-az4
- アジアパシフィック (ムンバイ)/ap-south-1-aps1-az1, aps1-az2, aps1-az3
- アジアパシフィック (シンガポール)/ap-southeast-1-apse1-az1, apse1-az2, apse1-az3
- アジアパシフィック (シドニー)/ap-southeast-2-apse2-az1, apse2-az2, apse2-az3

- カナダ (中部)/ca-central-1-cac1-az1, cac1-az2, cac1-az4
- 欧州 (アイルランド)/eu-west-1-euw1-az1, uew1-az2, euw1-az3
- 欧州 (ロンドン)/eu-west-2-usw2-az1, usw2-az2, usw2-az3

データソースは、 Amazon Kendra コネクタに指定したサブネットからアクセスできる必要があります。

Amazon VPC サブネットの設定方法の詳細については、「Amazon VPC ユーザーガイド」の「 のサブネット Amazon VPC」を参照してください。

Amazon Kendra が 2 つ以上のサブネット間で接続をルーティングする必要がある場合は、複数のサブネットを準備できます。例えば、データソースを含むサブネットが IP アドレス外です。その場合、十分な IP アドレスを持ち Amazon Kendra 、最初のサブネットに接続されている追加のサブネットを に提供できます。複数のサブネットを一覧表示する場合、サブネットは相互に通信できる必要があります。

ステップ 2。のセキュリティ Amazon VPC グループの作成 Amazon Kendra

Amazon Kendra データソースコネクタを に接続するには Amazon VPC、 に割り当てる VPC から 1 つ以上のセキュリティグループを準備する必要があります Amazon Kendra。セキュリティグループは、 によって作成された Elastic Network Interface に関連付けられます Amazon Kendra。このネットワークインターフェイスは、 Amazon VPC サブネットにアクセスする Amazon Kendra ときのインバウンドトラフィックとアウトバウンドトラフィックを制御します。

セキュリティグループのアウトバウンドルールで、 Amazon Kendra データソースコネクタからのトラフィックが、同期するサブネットとデータソースにアクセスできることを確認してください。例えば、 MySQL コネクタを使用してMySQLデータベースから同期できます。デフォルトのポートを使用している場合、セキュリティグループは、データベースを実行するホストのポート 3306 Amazon Kendra へのアクセスを に許可する必要があります。

が Amazon Kendra 使用するには、次の値を使用してデフォルトのセキュリティグループを設定することをお勧めします。

- インバウンドルール – これを空のままにすると、すべてのインバウンドトラフィックがブロックされます。
- アウトバウンドルール – がデータソースからの同期リクエスト Amazon Kendra を開始できるように、すべてのアウトバウンドトラフィックを許可するルールを 1 つ追加します。

- IP バージョン – IPv4
- タイプ – すべてのトラフィック
- プロトコル – すべてのトラフィック
- ポート範囲 – すべて
- 送信先 – 0.0.0.0/0

Amazon VPC セキュリティグループの設定方法の詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループルール」を参照してください。

ステップ 3。外部データソースと を設定する Amazon VPC

外部データソースに、 がアクセス Amazon Kendra するための正しいアクセス許可設定とネットワーク設定があることを確認します。データソースの設定方法の詳細については、各コネクタページの前提条件セクションを参照してください。

また、 Amazon VPC 設定を確認し、 に割り当てるサブネットから外部データソースにアクセスできることを確認します Amazon Kendra。これを行うには、同じセキュリティグループを持つ同じサブネットに Amazon EC2 インスタンスを作成し、この Amazon EC2 インスタンスからデータソースへのアクセスをテストすることをお勧めします。詳細については、Amazon VPC 「接続のトラブルシューティング」を参照してください。

に接続するための Amazon Kendra データソースを設定する Amazon VPC

に新しいデータソースを追加する場合 Amazon Kendra、選択したデータソースコネクタが Amazon VPC この機能をサポートしている場合は、この機能を使用できます。

AWS Management Console または Amazon Kendra API を使用して、 を有効にした新しい Amazon Kendra データソース Amazon VPC を設定できます。具体的には、 CreateDataSource API オペレーションを使用し、 `VpcConfiguration`パラメータを使用して次の情報を指定します。

- `SubnetIds` – Amazon VPC サブネットの識別子のリスト
- `SecurityGroupIds` — Amazon VPC セキュリティグループの識別子のリスト

コンソールを使用する場合は、コネクタの設定 Amazon VPC 時に必要な情報を入力します。コンソールを使用してコネクタの Amazon VPC 機能を有効にするには、まず Amazon VPC を選択します。次に、Amazon VPC サブネットの識別子と Amazon VPC セキュリティグループの識別子を指

定します。「Amazon VPC の設定」で作成した Amazon VPC サブネットと Amazon VPC セキュリティグループを選択するか、既存のものを使用できます。

トピック

- 識別子の表示 Amazon VPC
- データソース IAM ロールの確認

識別子の表示 Amazon VPC

サブネットとセキュリティグループの識別子は、 Amazon VPC コンソールで設定されます。識別子を表示するには、次の手順を使用します。

サブネット識別子を表示するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. ナビゲーションペインで、[Subnets] (サブネット) を選択します。
3. サブネット リストから、データベースサーバーを含むサブネットを選択します。
4. 詳細 タブで、サブネット ID フィールドの識別子を書き留めます。

セキュリティグループ識別子を表示するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. ナビゲーションペインで、セキュリティグループ を選択します。
3. セキュリティグループリストで、識別子の対象となるグループを選択します。
4. 詳細 タブで、セキュリティグループ ID フィールドの識別子をメモします。

データソース IAM ロールの確認

データソースコネクタ AWS Identity and Access Management IAM) ロールに へのアクセス許可が含まれていることを確認します Amazon VPC。

コンソールを使用して IAM ロールの新しいロールを作成する場合、 はユーザーに代わって IAM ロールに正しいアクセス許可 Amazon Kendra を自動的に追加します。API を使用するか、既存の IAM ロールを使用する場合は、ロールに へのアクセス許可が含まれていることを確認します Amazon

VPC。適切なアクセス許可があることを確認するには、IAM 「VPC の ロール」を参照してください。

別の Amazon VPC サブネットを使用するように既存のデータソースを変更できます。ただし、データソースの IAM ロールを確認し、必要に応じて Amazon Kendra 、データソースコネクタが正しく動作するように変更します。

VPC のデータベースに接続する

次の例は、Virtual Private Cloud (VPC) で実行されているMySQLデータベースを接続する方法を示しています。この例では、デフォルト VPC から開始し、MySQLデータベースを作成する必要があることを前提としています。VPC がすでに作成されている場合は、次に示すように設定されていることを確認します。データベースがある場合は、新しいMySQLデータベースを作成する代わりにそれを使用できます。

ステップ

- ステップ 1: VPC を設定する
- ステップ 2: セキュリティグループを作成して設定する
- ステップ 3: データベースを作成する
- ステップ 4: データソースコネクタを作成する

ステップ 1: VPC を設定する

サブネットで実行されているMySQLデータベースにアクセス Amazon Kendra するためのプライベートサブネットと のセキュリティグループを持つように VPC を設定します。VPC 設定で提供されるサブネットは、米国西部 (オレゴン) リージョン、米国東部 (バージニア北部) リージョン、または欧州 (アイルランド) リージョンにある必要があります。

を使用して VPC を設定するには Amazon VPC

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. ナビゲーションペインで [Route tables] (ルートテーブル) を選択して、[Create route table] (ルートテーブルの作成) を選択します。
3. 名前 フィールドに と入力します**`Private subnet route table`**。VPC ドロップダウンから VPC を選択し、ルートテーブルの作成を選択します。[Close] (閉じる) をクリックして、ルートテーブルのリストに戻ります。

4. ナビゲーションペインで NAT ゲートウェイ を選択し、NAT ゲートウェイの作成 を選択します。
5. サブネット ドロップダウンから、パブリックサブネットであるサブネットを選択します。サブネット ID を書き留めます。
6. Elastic IP アドレスを持っていない場合は、[Create New EIP] (新しい EIP の作成) で、[Create a NAT Gateway] (NAT ゲートウェイの作成) を選択してから、[Close] (閉じる) をクリックします。
7. ナビゲーションペインで、ルートテーブル を選択します。
8. ルートテーブルリストから、ステップ 3 で作成した [Private subnet route table] (プライベートサブネットルートテーブル) を選択します。アクション から、ルートの編集 を選択します。
9. [Add Rule] (ルートの追加) を選択します。送信先には、 と入力**0.0.0.0/0**して、インターネットへのすべての送信トラフィックを許可します。[ターゲット] で、[NAT ゲートウェイ] を選択し、次に、ステップ 4 で作成したゲートウェイを選択します。変更の保存 を選択し、 を閉じる を選択します。
10. [Actions] (アクション) から、[Edit subnet associations] (サブネットの関連付けの編集) を選択します。
11. プライベートにするサブネットを選択します。上記で書き留めた NAT ゲートウェイを持つサブネットを選択しないでください。完了したら、関連付けの保存を選択します。

ステップ 2: セキュリティグループを作成して設定する

次に、データベースのセキュリティグループを設定します。

セキュリティグループを作成して設定するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. VPC の説明から、IPv4 CIDR を書き留めます。
3. ナビゲーションペインでセキュリティグループ を選択し、セキュリティグループの作成 を選択します。
4. [Security group name] (セキュリティグループ名) に **DataSourceInboundSecurityGroup** と入力します。説明を入力し、リストから VPC を選択します。セキュリティグループの作成を選択し、 を閉じるを選択します。
5. [Inbound rules] (インバウンドルール) タブを開きます。
6. インバウンドルールの編集 を選択し、ルールの追加 を選択します。

7. データベースの場合は、ポート範囲 のポート番号を入力します。例えば、 MySQLの場合は **3306**、HTTPS の場合は です**443**。[Source] (ソース) に、VPC のクラスレスドメイン間ルーティング (CIDR) を入力します。[Save rules] (ルールの保存) を選択し、[Close] (閉じる) をクリックします。

セキュリティグループでは、VPC 内のすべてのユーザーがデータベースに接続でき、インターネットへのアウトバウンド接続が許可されます。

ステップ 3: データベースを作成する

ドキュメントを保持するデータベースを作成します。または、既存のデータベースを使用できます。

MySQL データベースの作成方法については、「」を参照してくださいMySQL。

ステップ 4: データソースコネクタを作成する

VPC を設定してデータベースを作成したら、データベースのデータソースコネクタを作成できます。が Amazon Kendra サポートするデータベースコネクタの詳細については、「サポートされているコネクタ」を参照してください。

データベースについては、VPC、VPC で作成したプライベートサブネット、VPC で作成したセキュリティグループを設定していることを確認してください。

VPC 接続の問題のトラブルシューティング

Virtual Private Cloud (VPC) 接続に問題がある場合は、アクセス IAM 許可、セキュリティグループ設定、サブネットのルートテーブルが正しく設定されていることを確認します。

データソースコネクタの同期に失敗する考えられる原因の 1 つは、 に割り当てたサブネットからデータソースにアクセスできない可能性があることです Amazon Kendra。この問題のトラブルシューティングを行うには、同じ Amazon VPC 設定で Amazon EC2 インスタンスを作成することをお勧めします。次に、REST API コールまたはその他のメソッド (データソースの特定のタイプに基づく) を使用して、この Amazon EC2 インスタンスからデータソースにアクセスしようとします。

作成した Amazon EC2 インスタンスからデータソースに正常にアクセスすると、このサブネットからデータソースにアクセスできることを意味します。したがって、同期の問題は、 がデータソースにアクセスできないことに関連しません Amazon VPC。

VPC 設定から Amazon EC2 インスタンスにアクセスできず、作成した Amazon EC2 インスタンスで検証できない場合は、さらにトラブルシューティングを行う必要があります。例えば、接続の問題

に関するエラーで同期に失敗した Amazon S3 コネクタがある場合、 Amazon S3 コネクタに割り当てたのと同じ Amazon VPC 設定で Amazon EC2 インスタンスを設定できます。次に、この Amazon EC2 インスタンスを使用して、 Amazon VPC が正しく設定されているかどうかをテストします。

以下は、 Amazon S3 データソースと Amazon VPC の接続をトラブルシューティングするために Amazon EC2 インスタンスを設定する例です。

トピック

ステップ 1: Amazon EC2 インスタンスを起動する

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. インスタンスの起動を選択します。
3. ネットワーク設定 を選択し、編集 を選択し、次の操作を行います。

   a. に割り当てたのと同じ VPC とサブネットを選択します Amazon Kendra。

   b. ファイアウォール (セキュリティグループ) で、既存のセキュリティグループを選択を選択します。次に、 に割り当てたセキュリティグループを選択します Amazon Kendra。

   > **Note**
   >
   > セキュリティグループは、 へのアウトバウンドトラフィックを許可する必要があります Amazon S3。

   c. パブリック IP の自動割り当てを無効に設定します。

   d. 詳細 で、次の操作を行います。

   - IAM インスタンスプロファイル で、新しい IAM プロファイルの作成 を選択し、 IAM インスタンスプロファイルを作成してインスタンスにアタッチします。プロファイルに へのアクセス許可があることを確認します Amazon S3。詳細については、「」の「 Amazon S3 バケットへのアクセス権を Amazon EC2 インスタンスに付与するにはどうすればよいですか？」を参照してください AWS re:Post。
   - その他の設定はすべてデフォルトのままにしておきます。

e. Amazon EC2 インスタンスを確認して起動します。

ステップ 2: Amazon EC2 インスタンスに接続する

Amazon EC2 インスタンスが実行されたら、インスタンスの詳細ページに移動し、インスタンスに接続します。これを行うには、Linux インスタンス用ユーザーガイドの「EC2 Instance Connect Endpoint を使用してパブリック IPv4 アドレスを必要とせずにインスタンスに接続する」の手順を使用します。 Amazon EC2

ステップ 3: アクセスをテスト Amazon S3 する

Amazon EC2 インスタンスターミナルに接続したら、 AWS CLI コマンドを実行して、このプライベートサブネットから Amazon S3 バケットへの接続をテストします。

Amazon S3 アクセスをテストするには、 AWS CLI で次のコマンドを入力します AWS CLI。 `aws s3 ls`

AWS CLI コマンドが実行されたら、以下を確認します。

- 必要な IAM アクセス許可を正しく設定し、 Amazon S3 設定が正しい場合は、 Amazon S3 バケットのリストが表示されます。
- などのアクセス許可エラーが表示された場合は`Access Denied`、yourVPC 設定が正しい可能性がありますが、アクセス IAM 許可または Amazon S3 バケットポリシーに問題がある可能性があります。

コマンドがタイムアウトしている場合、VPC のセットアップが正しくなく、Amazon EC2 インスタンスがサブネットから Amazon S3 にアクセスできないため、接続がタイムアウトしている可能性があります。VPC を再設定して、もう一度試してください。

インデックス、データソース、またはバッチアップロードされたドキュメントの削除

このセクションでは、インデックス、インデックス内のドキュメントのデータソースリポジトリ、またはバッチアップロードしたインデックス内のドキュメントを削除する方法を説明します。

トピック

インデックスを削除する

インデックスを使用しなくなったら、Amazon Kendra からインデックスを削除できます。例えば、次の場合にインデックスを削除します。

- インデックスを使用しなくなり、AWS アカウントへの請求料金を削減したい。Amazon Kendra インデックスは、インデックスに対してクエリを実行するかどうかにかかわらず、実行中は料金が発生します。
- Amazon Kendra の別のエディションのインデックスを再設定する。既存のインデックスを削除し、別のエディションで新しいインデックスを作成します。
- アカウントのインデックスの最大数に達したが、クォータは超えないようにしたい。既存のインデックスを削除し、新しいインデックスを追加します。作成できるインデックスの最大数の詳細については、「Quotas」を参照してください。

インデックスを削除するには、コンソール、AWS Command Line Interface、AWS CloudFormation スクリプト、または `DeleteIndex` API を使用します。インデックスを削除すると、インデックスと関連するすべてのデータソースとドキュメントデータが削除されます。インデックスを削除しても、元のドキュメントはストレージから削除されません。

インデックスの削除は、非同期演算です。インデックスの削除を開始すると、インデックスのステータスは `DELETING` に変わります。インデックスに関連する情報がすべて削除されるまで、ステータスは `DELETING` のままです。インデックスが削除されると、そのインデックスは、ListIndices API の呼び出しの結果に表示されなくなります。削除されたインデックスの識別子を使用して DescribeIndex API を呼び出すと、`ResourceNotFound` 例外を受け取ります。

インデックスを削除するには (コンソール)

1. AWS Management Console にサインインして Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. ナビゲーションペインで、[Indexes] (インデックス) を選択し、削除するインデックスを選択します。
3. [Delete] (削除) を選択して、選択したインデックスを削除します。

インデックスを削除するには (CLI)

- AWS CLI で次のコマンドを使用します。次のコマンドは、Linux と macOS 用にフォーマットされています。Windows を使用している場合、Unix 行連結記号 (\) をキャレット (^) に置き換えます。

```
aws kendra delete-index \
   --id index-id
```

データソースの削除

データソースに含まれる情報を Amazon Kendra インデックスから削除する場合は、データソースを削除します。例えば、以下の場合にデータソースを削除してください。

- データソースが誤って構成されている。データソースを削除し、データソースの削除が完了するのを待ってから、再作成します。
- あるデータソースから別のデータソースにドキュメントを移行した。元のデータソースを削除し、新しい場所に再作成します。
- インデックスのデータソースの制限に達した。既存のデータソースの 1 つを削除し、新しいデータソースを追加します。作成可能なデータソースの数については、クォータ を参照してください。

データソースを削除するには、コンソール、AWS Command Line Interface (AWS CLI)、`DeleteDataSource` API、または AWS CloudFormation スクリプトを使用します。データソースを削除すると、そのデータソースに関するすべての情報がインデックスから削除されます。データソースの同期のみを停止する場合は、データソースの同期スケジュールを「オンデマンドで実行」に変更します。

データソースの削除は、非同期演算です。データソースの削除を開始すると、データソースのステータスは DELETING に変わります。データソースに関連する情報が削除されるまで、ステータスは DELETING のままです。データソースが削除されると、そのデータソースは、ListDataSources API の呼び出しの結果に表示されなくなります。削除されたデータソースの識別子を使用して DescribeDataSource API を呼び出すと、ResourceNotFound 例外を受け取ります。

> **Note**
>
> データソース全体の削除、データソースから特定のドキュメントを削除した後のインデックスの再同期には、削除するドキュメントの数に応じて、最大で 1 時間以上かかる場合があります。

データソースを削除するには (コンソール)

1. AWS Management Console にサインインして Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. ナビゲーションペインで、[Indexes] (インデックス) を選択し、削除するデータソースを含むインデックスを選択します。
3. ナビゲーションペインで、[Data source] (データソース) をクリックします。
4. 削除するデータソースを選択します。
5. [Delete] (削除) をクリックして、データソースを削除します。

データソースを削除するには (CLI)

- AWS Command Line Interface で次のコマンドを使用します。次のコマンドは、Linux と macOS 用にフォーマットされています。Windows を使用している場合、Unix 行連結記号 (\) をキャレット (^) に置き換えます。

```
aws kendra delete-data-source \
   --id data-source-id \
   --index-id index-id
```

データソースを削除すると、Amazon Kendra はデータソースに関する保存された情報をすべて削除します。Amazon Kendra はインデックスに保存されているすべてのドキュメントデータ、および

データソースに関連付けられているすべての実行履歴とメトリクスを削除します。データソースを削除しても、元のドキュメントはストレージから削除されません。

Amazon Kendra がデータソースを削除している間、データソース内のドキュメントは、DescribeIndex API により返されたドキュメント数に含まれる可能性があります。Amazon Kendra がデータソースを削除している間、データソースのドキュメントが検索結果に表示されることがあります。

DeleteDataSource API を呼び出すか、コンソールでデータソースを削除するとすぐに、Amazon Kendra は、データソースのリソースをリリースします。データソースを削除してデータソースの数を制限以下に減らす場合は、すぐに新しいデータソースを作成できます。

データソースを削除し、ドキュメントデータに別のデータソースを作成する場合は、最初のデータソースが削除されるのを待ってから、新しいデータソースを同期します。

Amazon Kendra と同期プロセス中のデータソースを削除できます。同期が停止し、データソースが削除されます。データソースの削除時に同期を開始しようとすると、ConflictException 例外を取得します。

関連付けられているインデックスが DELETING ステータスの場合、データソースは削除できません。インデックスを削除すると、そのインデックスのすべてのデータソースが削除されます。インデックスの削除は、そのインデックスのデータソースが DELETING ステータスの間に開始できます。

同じ Amazon S3 バケットを参照している 2 つのデータソースなど、同じドキュメントを参照している 2 つのデータソースがある場合、一方のデータソースを削除すると、インデックス内のドキュメントが矛盾することがあります。2 つのデータソースが同じドキュメントを参照する場合、ドキュメントデータのコピーが 1 つだけインデックスに保存されます。1 つのデータソースを削除すると、ドキュメントのインデックスデータが削除されます。もう 1 つのデータソースは、ドキュメントが削除されたことを認識していないため、Amazon Kendra はドキュメントの次回同期時に正しくインデックスを再作成しません。同じドキュメントの場所を指すデータソースが 2 つある場合は、両方のデータソースを削除してから 1 つを再作成する必要があります。

バッチアップロードしたドキュメントの削除

BatchDeleteDocument API を使用して、ドキュメントをインデックスから直接削除できます。コンソールを使用してドキュメントを直接削除することはできません。コンソールを使用する場合は、データソースリポジトリから特定のドキュメントを削除してインデックスと再同期するか、データソースコネクタ全体を削除できます。

BatchDeleteDocument を使用したインデックスのドキュメント削除は、非同期演算です。BatchDeleteDocument API を呼び出した後、BatchGetDocumentStatus API を使用して、ドキュメント削除の進行状況をモニタリングします。ドキュメントがインデックスから削除されると、Amazon Kendra は、NOT_FOUND をステータスとして返します。

> **Note**
>
> BatchDeleteDocument を使用してインデックスからドキュメントを削除する場合、削除するドキュメントの数に応じて、1 時間以上かかることがあります。

インデックス (CLI) にバッチアップロードしたドキュメントの削除方法。

- AWS Command Line Interface で次のコマンドを使用します。次のコマンドは、Linux と macOS 用にフォーマットされています。Windows を使用している場合、Unix 行連結記号 (\) をキャレット (^) に置き換えます。

```
aws kendra batch-delete-document \
   --index-id index-id \
   --document-id-list 'doc-id-1' 'doc-id-2'
```

取り込み中のドキュメントの強化

ドキュメントの取り込みプロセス中に、コンテンツおよびドキュメントのメタデータフィールドまたは属性を変更できます。Amazon Kendra の Custom Document Enrichment 機能を使用すると、Amazon Kendra にドキュメントを取り込むときに、ドキュメントの属性とコンテンツを作成、変更、または削除できます。つまり、必要に応じて、データを操作して取り込むことができます。

この機能を使用すると、ドキュメントをどのように処理し、Amazon Kendra に取り込むかを制御できます。例えば、Amazon Kendra にドキュメントを取り込むときに、ドキュメントのメタデータ内の個人を特定できる情報をスクラブできます。

この機能のもう 1 つの用途は、AWS Lambda で Lambda 関数を呼び出して、画像に対する光学文字認識 (OCR)、テキストの翻訳、および検索または分析のためのデータの準備に関するその他のタスクを実行します。例えば、関数を呼び出すと、画像上で OCR を実行できます。この関数は、画像のテキストを解釈し、各画像をテキスト文書として扱うことができます。郵送された顧客調査を受け取り、これらのアンケートを画像として保存する企業は、これらの画像をテキストドキュメントとして Amazon Kendra に取り込むことができます。その後、企業は Amazon Kendra で貴重な顧客調査情報を検索できます。

基本的な操作を使用してデータの最初の解析として適用し、次に、Lambda 関数を使用して、より複雑な操作をデータに適用できます。例えば、基本的な操作を使用してドキュメントメタデータフィールド 「Customer_ID」のすべての値を削除し、Lambda 関数を適用してドキュメント内のテキストの画像からテキストを抽出することができます。

Custom Document Enrichment の仕組み

Custom Document Enrichment の全体のプロセスは次のとおりです。

1. Custom Document Enrichment は、データソースを作成または更新するとき、またはドキュメントを Amazon Kendra に直接インデックス作成するときに設定します。
2. Amazon Kendra は、インライン構成または基本ロジックを適用してデータを変更します。詳細については、「the section called “メタデータを変更する基本操作”」を参照してください。
3. 高度なデータ操作を構成する場合、Amazon Kendra は、元の未加工のドキュメント、または構造化された解析済みドキュメントにこれを適用できます。詳細については、「the section called “Lambda 関数:メタデータまたはコンテンツの抽出と変更”」を参照してください。
4. 変更されたドキュメントは Amazon Kendra に取り込まれます。

このプロセスのどの時点でも、構成が有効でない場合、Amazon Kendra はエラーをスローします。

CreateDataSource、UpdateDataSource、または BatchPutDocument API を呼び出す場合、Custom Document Enrichment 構成を指定します。`BatchPutDocument` を呼び出すと、各リクエストで Custom Document Enrichment を設定する必要があります。コンソールを使用する場合は、インデックスを選択し、[Document enrichments] (ドキュメントのエンリッチメント) をクリックして、Custom Document Enrichment を構成します。

コンソールで[ドキュメントのエンリッチメント] を使用する場合、API を使用する場合と同様に、基本操作のみ、Lambda 関数のみ、あるいはその両方を設定できます。コンソールステップで [次へ] を選択して、基本操作を設定せず、Lambda 関数のみを設定するよう選択できます。これには、元の (抽出前) データに適用するか構造化データ (抽出後) に適用するかなどが含まれます。設定を保存するには、コンソールのすべてのステップを完了する必要があります。すべての手順を完了しないと、ドキュメントの設定は保存されません。

メタデータを変更する基本操作

基本ロジックを使用して、ドキュメントのフィールド、およびコンテンツを操作できます。これには、フィールド内の値の削除、条件を使用したフィールドの値の変更、またはフィールドの作成が含まれます。基本ロジックを使用して操作できる範囲を超える高度な操作については、Lambda 関数を呼び出します。詳細については、「the section called “Lambda 関数:メタデータまたはコンテンツの抽出と変更”」を参照してください。

基本ロジックを適用するには、DocumentAttributeTarget オブジェクトを使用して操作するターゲットフィールドを指定します。属性キーを指定します。例えば、「Department」キーは、ドキュメントに関連付けられているすべての部門名を保持するフィールドまたは属性です。特定の条件が満たされた場合に、ターゲットフィールドで使用する値を指定することもできます。条件の設定は、DocumentAttributeCondition オブジェクトを使用します。例えば、「Source_URI」フィールドに URI 値に「financial」が含まれている場合に、ターゲットフィールド「Department」にドキュメントのターゲット値「Finance」を事前に入力するという条件を設定します。ターゲットドキュメント属性の値を削除することもできます。

コンソールを使用して基本ロジックを適用するには、インデックスを選択し、ナビゲーションメニューの [Document enrichments] (ドキュメントのエンリッチメント) を選択します。[基本的な演算の設定] をクリックして、ドキュメントのフィールドとコンテンツに基本的な演算を適用します。

次に、基本ロジックを使用して「Customer_ID」というドキュメントフィールドのすべての顧客識別番号を削除する例を示します。

例 1: ドキュメントに関連付けられている顧客識別番号の削除

基本操作が適用される前のデータ。

Document_ID	Body_Text	Customer_ID
1	Lorem Ipsum。	CID1234
2	Lorem Ipsum。	CID1235
3	Lorem Ipsum。	CID1236

基本操作が適用された後のデータ。

Document_ID	Body_Text	Customer_ID
1	Lorem Ipsum。	
2	Lorem Ipsum。	
3	Lorem Ipsum。	

次に、基本ロジックを使用して「Department」というフィールドを作成し、「source_URI」フィールドの情報に基づいてこのフィールドに部門名を事前に入力する例を示します。これは、「Source_URI」フィールドに URI 値に「financial」が含まれている場合に、ターゲットフィールド「Department」にドキュメントのターゲット値「Finance」を事前に入力するという条件を使用します。

例 2:「Department」フィールドを作成し、条件を使用してドキュメントに関連付けられた部門名を事前に入力します。

基本操作が適用される前のデータ。

Document_ID	Body_Text	Source_URI
1	Lorem Ipsum。	financial/1
2	Lorem Ipsum。	financial/2

Document_ID	Body_Text	Source_URI
3	Lorem Ipsum。	financial/3

基本操作が適用された後のデータ。

Document_ID	Body_Text	Source_URI	Department
1	Lorem Ipsum。	financial/1	財務
2	Lorem Ipsum。	financial/2	財務
3	Lorem Ipsum。	financial/3	財務

> **Note**
>
> Amazon Kendra では、ターゲットドキュメントフィールドがインデックスフィールドとしてまだ作成されていない場合、それを作成することはできません。インデックスフィールドを作成したら、`DocumentAttributeTarget` を使用してドキュメントフィールドを作成できます。Amazon Kendra はその後、新しく作成したドキュメントのメタデータフィールドをインデックスフィールドにマッピングします。

次のコードは、ドキュメントに関連付けられている顧客識別番号を削除するための基本的なデータ操作を設定する例です。

Console

顧客識別番号を削除するための基本的なデータ操作を構成するには

1. 左側のナビゲーションペインの [Indexes] (インデックス) で、[Document enrichments] (ドキュメントのエンリッチメント) を選択して、[Add document enrichment] (ドキュメントのエンリッチメントを追加) を選択します。
2. [基本的な操作の設定] ページで、ドキュメントのフィールドとコンテンツを変更するデータソースをドロップダウンから選択します。次に、ドロップダウンからドキュメントフィールド名「Customer_ID」を選択し、ドロップダウンからインデックスフィールド名

「Customer_ID」を選択し、ドロップダウンからターゲットアクション [Delete] (削除) を選択します。次に、[Add basic operation] (基本的な演算の追加) を選択します。

CLI

顧客識別番号を削除するための基本的なデータ操作を構成するには

```
aws kendra create-data-source \
 --name data-source-name \
 --index-id index-id \
 --role-arn arn:aws:iam::account-id:role/role-name \
 --type S3 \
 --configuration '{"S3Configuration":{"BucketName":"S3-bucket-name"}}' \
 --custom-document-enrichment-configuration '{"InlineConfigurations":[{"Target":
{"TargetDocumentAttributeKey":"Customer_ID", "TargetDocumentAttributeValueDeletion":
 true}}]}'
```

Python

顧客識別番号を削除するための基本的なデータ操作を構成するには

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Create a data source with customizations")

# Provide the name of the data source
name = "data-source-name"
# Provide the index ID for the data source
index_id = "index-id"
# Provide the IAM role ARN required for data sources
role_arn = "arn:aws:iam::${account-id}:role/${role-name}"
# Provide the data source connection information
data_source_type = "S3"
S3_bucket_name = "S3-bucket-name"
# Configure the data source with Custom Document Enrichment
configuration = {"S3Configuration":
        {
            "BucketName": S3_bucket_name
```

```
        }
    }
custom_document_enrichment_configuration = {"InlineConfigurations":[
        {
            "Target":{"TargetDocumentAttributeKey":"Customer_ID",
                      "TargetDocumentAttributeValueDeletion": True}
        }]
    }

try:
    data_source_response = kendra.create_data_source(
        Name = name,
        IndexId = index_id,
        RoleArn = role_arn,
        Type = data_source_type
        Configuration = configuration
        CustomDocumentEnrichmentConfiguration =
 custom_document_enrichment_configuration
    )

    pprint.pprint(data_source_response)

    data_source_id = data_source_response["Id"]

    print("Wait for Amazon Kendra to create the data source with your
 customizations.")

    while True:
        # Get the details of the data source, such as the status
        data_source_description = kendra.describe_data_source(
            Id = data_source_id,
            IndexId = index_id
        )
        status = data_source_description["Status"]
        print(" Creating data source. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break

    print("Synchronize the data source.")

    sync_response = kendra.start_data_source_sync_job(
        Id = data_source_id,
        IndexId = index_id
```

```
    )

    pprint.pprint(sync_response)

    print("Wait for the data source to sync with the index.")

    while True:

        jobs = kendra.list_data_source_sync_jobs(
            Id= data_source_id,
            IndexId= index_id
        )

        # For this example, there should be one job
        status = jobs["History"][0]["Status"]

        print(" Syncing data source. Status: "+status)
        time.sleep(60)
        if status != "SYNCING":
            break

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

顧客識別番号を削除するための基本的なデータ操作を構成するには

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.CreateIndexRequest;
import software.amazon.awssdk.services.kendra.model.CreateIndexResponse;
import software.amazon.awssdk.services.kendra.model.DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.DataSourceStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJob;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJobStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceType;
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceRequest;
```

```
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.DescribeIndexRequest;
import software.amazon.awssdk.services.kendra.model.DescribeIndexResponse;
import software.amazon.awssdk.services.kendra.model.IndexStatus;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsRequest;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsResponse;
import software.amazon.awssdk.services.kendra.model.S3DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobRequest;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobResponse;

public class CreateDataSourceWithCustomizationsExample {

    public static void main(String[] args) throws InterruptedException {
        System.out.println("Create a data source with customizations");

        String dataSourceName = "data-source-name";
        String indexId = "index-id";
        String dataSourceRoleArn = "arn:aws:iam::account-id:role/role-name";
        String s3BucketName = "S3-bucket-name"

        KendraClient kendra = KendraClient.builder().build();

        CreateDataSourceRequest createDataSourceRequest = CreateDataSourceRequest
            .builder()
            .name(dataSourceName)
            .description(experienceDescription)
            .roleArn(experienceRoleArn)
            .type(DataSourceType.S3)
            .configuration(
                DataSourceConfiguration
                    .builder()
                    .s3Configuration(
                        S3DataSourceConfiguration
                            .builder()
                            .bucketName(s3BucketName)
                            .build()
                    ).build()
            )
            .customDocumentEnrichmentConfiguration(
                CustomDocumentEnrichmentConfiguration
                    .builder()
                    .inlineConfigurations(Arrays.asList(
                        InlineCustomDocumentEnrichmentConfiguration
                            .builder()
```

```
                            .target(
                                DocumentAttributeTarget
                                    .builder()
                                    .targetDocumentAttributeKey("Customer_ID")
                                    .targetDocumentAttributeValueDeletion(true)
                                    .build())
                            .build()
                    )).build();

        CreateDataSourceResponse createDataSourceResponse =
kendra.createDataSource(createDataSourceRequest);
        System.out.println(String.format("Response of creating data source: %s",
createDataSourceResponse));

        String dataSourceId = createDataSourceResponse.id();
        System.out.println(String.format("Waiting for Kendra to create the data
source %s", dataSourceId));
        DescribeDataSourceRequest describeDataSourceRequest =
DescribeDataSourceRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

        while (true) {
            DescribeDataSourceResponse describeDataSourceResponse =
kendra.describeDataSource(describeDataSourceRequest);

            DataSourceStatus status = describeDataSourceResponse.status();
            System.out.println(String.format("Creating data source. Status: %s",
status));
            TimeUnit.SECONDS.sleep(60);
            if (status != DataSourceStatus.CREATING) {
                break;
            }
        }

        System.out.println(String.format("Synchronize the data source %s",
dataSourceId));
        StartDataSourceSyncJobRequest startDataSourceSyncJobRequest =
StartDataSourceSyncJobRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
```

```
            .build();
        StartDataSourceSyncJobResponse startDataSourceSyncJobResponse =
 kendra.startDataSourceSyncJob(startDataSourceSyncJobRequest);
        System.out.println(String.format("Waiting for the data
 source to sync with the index %s for execution ID %s", indexId,
 startDataSourceSyncJobResponse.executionId()));

        // For this example, there should be one job
        ListDataSourceSyncJobsRequest listDataSourceSyncJobsRequest =
 ListDataSourceSyncJobsRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

        while (true) {
            ListDataSourceSyncJobsResponse listDataSourceSyncJobsResponse =
 kendra.listDataSourceSyncJobs(listDataSourceSyncJobsRequest);
            DataSourceSyncJob job = listDataSourceSyncJobsResponse.history().get(0);
            System.out.println(String.format("Syncing data source. Status: %s",
 job.status()));

            TimeUnit.SECONDS.sleep(60);
            if (job.status() != DataSourceSyncJobStatus.SYNCING) {
                break;
            }

        }

        System.out.println("Data source creation with customizations is complete");
    }
}
```

Lambda 関数:メタデータまたはコンテンツの抽出と変更

Lambda 関数を使用して、ドキュメントのフィールド、およびコンテンツを操作できます。これは、基本ロジックを超えて高度なデータ操作を適用する場合に便利です。例えば、画像上のテキストを解釈し、各画像をテキストドキュメントとして扱う、光学文字認識 (OCR) を使用します。または、特定のタイムゾーンで現在の日付/時間を取得し、日付フィールドに空の値がある日付/時刻を挿入します。

まず基本ロジックを適用し、Lambda 関数を使用してデータをさらに操作でき、その逆も可能です。Lambda 関数のみを適用することもできます。

Amazon Kendra は Lambda 関数を呼び出して、CustomDocumentEnrichmentConfiguration の一部として取り込みプロセス中に高度なデータ操作を適用します。Lambda 関数を実行し、Amazon S3 バケットにアクセスしてデータ操作の出力を保存する許可を含むロールを指定します。「IAM アクセスロール」を参照してください。

Amazon Kendra では、元のドキュメント、または構造化された解析済みドキュメントに Lambda 関数を適用できます。PreExtractionHookConfiguration を使用して、元のデータまたは raw データを受け取り、データ操作を適用する Lambda 関数を構成できます。また、PostExtractionHookConfiguration を使用して構造化ドキュメントを受け取り、データ操作を適用する Lambda 関数を設定することもできます。Amazon Kendra はドキュメントのメタデータとテキストを抽出してドキュメントを構造化します。Lambda 関数は、必須のリクエストとレスポンスの構造に従う必要があります。詳細については、「the section called “Lambda 関数のデータ制約”」を参照してください。

コンソールで Lambda 関数を設定するには、インデックスを選択し、ナビゲーションメニューの [Document enrichments] (ドキュメントのエンリッチメント) を選択します。[Configure Lambda functions] (Lambda 関数の設定) をクリックして、Lambda 関数を設定します。

`PreExtractionHookConfiguration` に設定できる Lambda 関数は 1 つのみ、`PostExtractionHookConfiguration` に設定できる Lambda 関数も 1 つのみです。ただし、Lambda 関数は必要とする他の関数を呼び出すことができます。`PreExtractionHookConfiguration` および `PostExtractionHookConfiguration` の両方またはいずれか 1 つを設定できます。`PreExtractionHookConfiguration` の Lambda 関数の実行時間が 5 分を超えないよう、また、`PostExtractionHookConfiguration` の Lambda 関数の実行時間が 1 分を超えないようにしてください。Custom Document Enrichment の設定は、これを設定しない場合よりも、Amazon Kendra にドキュメントを取り込むのに時間がかかります。

条件が満たされた場合にのみ Lambda 関数を呼び出すように Amazon Kendra を設定できます。例えば、空の日付/時間値がある場合、Amazon Kendra は現在の日付/時間を挿入する関数を呼び出す必要があるという条件を指定できます。

次に、Lambda 関数を使用して OCR を実行して画像からテキストを解釈し、このテキストを「Document_Image_Text」というフィールドに保存する例を示します。

例 1: 画像からテキストを抽出してテキストドキュメントを作成する

高度な操作が適用される前のデータ。

Document_ID	Document_Image
1	image_1.png
2	image_2.png
3	image_3.png

高度な操作が適用された後のデータ。

Document_ID	Document_Image	Document_Image_Text
1	image_1.png	メールでのアンケートの回答
2	image_2.png	メールでのアンケートの回答
3	image_3.png	メールでのアンケートの回答

次に、Lambda 関数を使用して、空の日付値に対する現在の日付/時間を挿入する例を示します。これは、日付フィールドの値が「null」の場合、これを現在の日付/時刻に置き換えるという条件を使用します。

例 2: Last_Updated フィールドの空の値を、現在の日付/時間で置き換える。

高度な操作が適用される前のデータ。

Document_ID	Body_Text	Last_Updated
1	Lorem Ipsum。	2020 年 1 月 1 日
2	Lorem Ipsum。	
3	Lorem Ipsum。	2020 年 7 月 1 日

高度な操作が適用された後のデータ。

Document_ID	Body_Text	Last_Updated
1	Lorem Ipsum。	2020 年 1 月 1 日
2	Lorem Ipsum。	2021 年 12 月 1 日
3	Lorem Ipsum。	2020 年 7 月 1 日

次のコードは、raw の元のデータに対して高度なデータ操作のための Lambda 関数を設定する例です。

Console

raw の元のデータに対して高度なデータ操作のための Lambda 関数を設定するには

1. 左側のナビゲーションペインの [Indexes] (インデックス) で、[Document enrichments] (ドキュメントのエンリッチメント) を選択して、[Add document enrichment] (ドキュメントのエンリッチメントを追加) を選択します。
2. [Lambda 関数の設定] ページの [事前抽出用の Lambda] セクションで、ドロップダウンから Lambda 関数 ARN と Amazon S3 バケットを選択します。ドロップダウンから、新しいロールを作成するオプションを選択して、IAM アクセスロールを追加します。これにより、Amazon Kendra ドキュメントエンリッチメントの作成に必要なアクセス許可が作成されます。

CLI

raw の元のデータに対して高度なデータ操作のための Lambda 関数を設定するには

```
aws kendra create-data-source \
 --name data-source-name \
 --index-id index-id \
 --role-arn arn:aws:iam::account-id:role/role-name \
 --type S3 \
 --configuration '{"S3Configuration":{"BucketName":"S3-bucket-name"}}' \
 --custom-document-enrichment-configuration '{"PreExtractionHookConfiguration":
{"LambdaArn":"arn:aws:iam::account-id:function/function-name", "S3Bucket":"S3-
bucket-name"}, "RoleArn": "arn:aws:iam:account-id:role/cde-role-name"}'
```

Python

raw の元のデータに対して高度なデータ操作のための Lambda 関数を設定するには

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Create a data source with customizations.")

# Provide the name of the data source
name = "data-source-name"
# Provide the index ID for the data source
index_id = "index-id"
# Provide the IAM role ARN required for data sources
role_arn = "arn:aws:iam::${account-id}:role/${role-name}"
# Provide the data source connection information
data_source_type = "S3"
S3_bucket_name = "S3-bucket-name"
# Configure the data source with Custom Document Enrichment
configuration = {"S3Configuration":
        {
            "BucketName": S3_bucket_name
        }
    }
custom_document_enrichment_configuration = {"PreExtractionHookConfiguration":
        {
            "LambdaArn":"arn:aws:iam::account-id:function/function-name",
            "S3Bucket":"S3-bucket-name"
        }
    "RoleArn":"arn:aws:iam::account-id:role/cde-role-name"
    }

try:
    data_source_response = kendra.create_data_source(
        Name = name,
        IndexId = index_id,
        RoleArn = role_arn,
        Type = data_source_type
        Configuration = configuration
```

```
        CustomDocumentEnrichmentConfiguration =
custom_document_enrichment_configuration
    )

    pprint.pprint(data_source_response)

    data_source_id = data_source_response["Id"]

    print("Wait for Amazon Kendra to create the data source with your
customizations.")

    while True:
        # Get the details of the data source, such as the status
        data_source_description = kendra.describe_data_source(
            Id = data_source_id,
            IndexId = index_id
        )
        status = data_source_description["Status"]
        print(" Creating data source. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break

    print("Synchronize the data source.")

    sync_response = kendra.start_data_source_sync_job(
        Id = data_source_id,
        IndexId = index_id
    )

    pprint.pprint(sync_response)

    print("Wait for the data source to sync with the index.")

    while True:

        jobs = kendra.list_data_source_sync_jobs(
            Id = data_source_id,
            IndexId = index_id
        )

        # For this example, there should be one job
        status = jobs["History"][0]["Status"]
```

```
        print(" Syncing data source. Status: "+status)
        time.sleep(60)
        if status != "SYNCING":
            break

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

raw の元のデータに対して高度なデータ操作のための Lambda 関数を設定するには

```
package com.amazonaws.kendra;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.CreateDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.CreateIndexRequest;
import software.amazon.awssdk.services.kendra.model.CreateIndexResponse;
import software.amazon.awssdk.services.kendra.model.DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.DataSourceStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJob;
import software.amazon.awssdk.services.kendra.model.DataSourceSyncJobStatus;
import software.amazon.awssdk.services.kendra.model.DataSourceType;
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceRequest;
import software.amazon.awssdk.services.kendra.model.DescribeDataSourceResponse;
import software.amazon.awssdk.services.kendra.model.DescribeIndexRequest;
import software.amazon.awssdk.services.kendra.model.DescribeIndexResponse;
import software.amazon.awssdk.services.kendra.model.IndexStatus;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsRequest;
import software.amazon.awssdk.services.kendra.model.ListDataSourceSyncJobsResponse;
import software.amazon.awssdk.services.kendra.model.S3DataSourceConfiguration;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobRequest;
import software.amazon.awssdk.services.kendra.model.StartDataSourceSyncJobResponse;


public class CreateDataSourceWithCustomizationsExample {

    public static void main(String[] args) throws InterruptedException {
        System.out.println("Create a data source with customizations");
```

```
        String dataSourceName = "data-source-name";
        String indexId = "index-id";
        String dataSourceRoleArn = "arn:aws:iam::account-id:role/role-name";
        String s3BucketName = "S3-bucket-name"

        KendraClient kendra = KendraClient.builder().build();

        CreateDataSourceRequest createDataSourceRequest = CreateDataSourceRequest
            .builder()
            .name(dataSourceName)
            .description(experienceDescription)
            .roleArn(experienceRoleArn)
            .type(DataSourceType.S3)
            .configuration(
                DataSourceConfiguration
                    .builder()
                    .s3Configuration(
                        S3DataSourceConfiguration
                            .builder()
                            .bucketName(s3BucketName)
                            .build()
                    ).build()
            )
            .customDocumentEnrichmentConfiguration(
                CustomDocumentEnrichmentConfiguration
                    .builder()
                    .preExtractionHookConfiguration(
                        HookConfiguration
                            .builder()
                            .lambdaArn("arn:aws:iam::account-id:function/function-
name")
                            .s3Bucket("S3-bucket-name")
                            .build())
                    .roleArn("arn:aws:iam::account-id:role/cde-role-name")
                    .build();

        CreateDataSourceResponse createDataSourceResponse =
 kendra.createDataSource(createDataSourceRequest);
        System.out.println(String.format("Response of creating data source: %s",
 createDataSourceResponse));

        String dataSourceId = createDataSourceResponse.id();
        System.out.println(String.format("Waiting for Kendra to create the data
 source %s", dataSourceId));
```

```
        DescribeDataSourceRequest describeDataSourceRequest =
DescribeDataSourceRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

        while (true) {
            DescribeDataSourceResponse describeDataSourceResponse =
kendra.describeDataSource(describeDataSourceRequest);

            DataSourceStatus status = describeDataSourceResponse.status();
            System.out.println(String.format("Creating data source. Status: %s",
status));
            TimeUnit.SECONDS.sleep(60);
            if (status != DataSourceStatus.CREATING) {
                break;
            }
        }

        System.out.println(String.format("Synchronize the data source %s",
dataSourceId));
        StartDataSourceSyncJobRequest startDataSourceSyncJobRequest =
StartDataSourceSyncJobRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();
        StartDataSourceSyncJobResponse startDataSourceSyncJobResponse =
kendra.startDataSourceSyncJob(startDataSourceSyncJobRequest);
        System.out.println(String.format("Waiting for the data
source to sync with the index %s for execution ID %s", indexId,
startDataSourceSyncJobResponse.executionId()));

        // For this example, there should be one job
        ListDataSourceSyncJobsRequest listDataSourceSyncJobsRequest =
ListDataSourceSyncJobsRequest
            .builder()
            .indexId(indexId)
            .id(dataSourceId)
            .build();

        while (true) {
```

```
            ListDataSourceSyncJobsResponse listDataSourceSyncJobsResponse =
 kendra.listDataSourceSyncJobs(listDataSourceSyncJobsRequest);
            DataSourceSyncJob job = listDataSourceSyncJobsResponse.history().get(0);
            System.out.println(String.format("Syncing data source. Status: %s",
 job.status()));

            TimeUnit.SECONDS.sleep(60);
            if (job.status() != DataSourceSyncJobStatus.SYNCING) {
                break;
            }

        }

        System.out.println("Data source creation with customizations is complete");
    }
}
```

Lambda 関数のデータ制約

高度なデータ操作のための Lambda 関数は、Amazon Kendra データコントラクトと連携します。制約は、Lambda 関数の必須のリクエストとレスポンスの構造です。Lambda 関数がこれらの構造に従わない場合、Amazon Kendra はエラーをスローします。

PreExtractionHookConfiguration の Lambda 関数は、次のリクエスト構造を期待します。

```
{
    "version": <str>,
    "dataBlobStringEncodedInBase64": <str>, //In the case of a data blob
    "s3Bucket": <str>, //In the case of an S3 bucket
    "s3ObjectKey": <str>, //In the case of an S3 bucket
    "metadata": <Metadata>
}
```

metadata 構造。これには、つぎの CustomDocumentAttribute 構造を含みます。

```
{
    "attributes": [<CustomDocumentAttribute<]
}

CustomDocumentAttribute
{
```

```
    "name": <str>,
    "value": <CustomDocumentAttributeValue>
}

CustomDocumentAttributeValue
{
    "stringValue": <str>,
    "integerValue": <int>,
    "longValue": <long>,
    "stringListValue": list<str>,
    "dateValue": <str>
}
```

PreExtractionHookConfiguration の Lambda 関数は、以下のレスポンス構造に従う必要があります。

```
{
    "version": <str>,
    "dataBlobStringEncodedInBase64": <str>, //In the case of a data blob
    "s3ObjectKey": <str>, //In the case of an S3 bucket
    "metadataUpdates": [<CustomDocumentAttribute>]
}
```

PostExtractionHookConfiguration の Lambda 関数は、次のリクエスト構造を期待します。

```
{
    "version": <str>,
    "s3Bucket": <str>,
    "s3ObjectKey": <str>,
    "metadata": <Metadata>
}
```

PostExtractionHookConfiguration の Lambda 関数は、以下のレスポンス構造に従う必要があります。

```
PostExtractionHookConfiguration Lambda Response
{
    "version": <str>,
    "s3ObjectKey": <str>,
    "metadataUpdates": [<CustomDocumentAttribute>]
}
```

変更されたドキュメントが Amazon S3 バケットにアップロードされます。変更されたドキュメントは、the section called “構造化ドキュメントの形式” に示す形式に従う必要があります。

構造化ドキュメントの形式

Amazon Kendra は、構造化ドキュメントを指定された Amazon S3 バケットにアップロードします。構造化ドキュメントは、次の形式に従います。

```
Kendra document

{
   "textContent": <TextContent>
}

TextContent
{
  "documentBodyText": <str>
}
```

データ制約に準拠する Lambda 関数の例

次の Python コードは、メタデータフィールド _authors、_document_title および raw の元のドキュメントの本文の高度な操作を適用する Lambda 関数の例です。

本文が Amazon S3 バケットに存在する場合

```
import json
import boto3

s3 = boto3.client("s3")

# Lambda function for advanced data manipulation
def lambda_handler(event, context):
    # Get the value of "S3Bucket" key name or item from the given event input
    s3_bucket = event.get("s3Bucket")
    # Get the value of "S3ObjectKey" key name or item from the given event input
    s3_object_key = event.get("s3ObjectKey")

    content_object_before_CDE = s3.get_object(Bucket = s3_bucket, Key = s3_object_key)
    content_before_CDE = content_object_before_CDE["Body"].read().decode("utf-8");
    content_after_CDE = "CDEInvolved " + content_before_CDE
```

```
    # Get the value of "metadata" key name or item from the given event input
    metadata = event.get("metadata")
    # Get the document "attributes" from the metadata
    document_attributes = metadata.get("attributes")

    s3.put_object(Bucket = s3_bucket, Key = "dummy_updated_kendra_document",
 Body=json.dumps(content_after_CDE))
    return {
        "version": "v0",
        "s3ObjectKey": "dummy_updated_kendra_document",
        "metadataUpdates": [
            {"name":"_document_title", "value":
{"stringValue":"title_from_pre_extraction_lambda"}},
            {"name":"_authors", "value":{"stringListValue":["author1", "author2"]}}
        ]
    }
```

本文がデータ BLOB に存在する場合

```
import json
import boto3
import base64

# Lambda function for advanced data manipulation
def lambda_handler(event, context):

    # Get the value of "dataBlobStringEncodedInBase64" key name or item from the given
 event input
    data_blob_string_encoded_in_base64 = event.get("dataBlobStringEncodedInBase64")
    # Decode the data blob string in UTF-8
    data_blob_string =
 base64.b64decode(data_blob_string_encoded_in_base64).decode("utf-8")
    # Get the value of "metadata" key name or item from the given event input
    metadata = event.get("metadata")
    # Get the document "attributes" from the metadata
    document_attributes = metadata.get("attributes")

    new_data_blob = "This should be the modified data in the document by pre processing
 lambda ".encode("utf-8")
    return {
        "version": "v0",
        "dataBlobStringEncodedInBase64":
 base64.b64encode(new_data_blob).decode("utf-8"),
```

```
        "metadataUpdates": [
            {"name":"_document_title", "value":
{"stringValue":"title_from_pre_extraction_lambda"}},
            {"name":"_authors", "value":{"stringListValue":["author1", "author2"]}}
        ]
    }
```

次の Python コードは、メタデータフィールド _authors、_document_title および構造化されたドキュメントまたは解析済みのドキュメントの本文の高度な操作を適用する Lambda 関数の例です。

```
import json
import boto3
import time

s3 = boto3.client("s3")

# Lambda function for advanced data manipulation
def lambda_handler(event, context):

    # Get the value of "S3Bucket" key name or item from the given event input
    s3_bucket = event.get("s3Bucket")
    # Get the value of "S3ObjectKey" key name or item from the given event input
    s3_key = event.get("s3ObjectKey")
    # Get the value of "metadata" key name or item from the given event input
    metadata = event.get("metadata")
    # Get the document "attributes" from the metadata
    document_attributes = metadata.get("attributes")

    kendra_document_object = s3.get_object(Bucket = s3_bucket, Key = s3_key)
    kendra_document_string = kendra_document_object['Body'].read().decode('utf-8')
    kendra_document = json.loads(kendra_document_string)
    kendra_document["textContent"]["documentBodyText"] = "Changing document body to a
 short sentence."

    s3.put_object(Bucket = s3_bucket, Key = "dummy_updated_kendra_document",
 Body=json.dumps(kendra_document))

    return {
        "version" : "v0",
        "s3ObjectKey": "dummy_updated_kendra_document",
        "metadataUpdates": [
```

```
            {"name": "_document_title", "value":{"stringValue":
 "title_from_post_extraction_lambda"}},
            {"name": "_authors", "value":{"stringListValue":["author1", "author2"]}}
        ]
    }
```

インデックスの検索

Amazon Kendra インデックスを検索するには Query API を使用します。Query API は、アプリケーションで使用するインデックスが作成されたドキュメントに関する情報を返します。このセクションでは、クエリの作成、フィルタリングの実行、および Query API からのレスポンスの解釈方法について説明します。

Amazon Kendra インデックスに登録したドキュメントを検索するには Amazon Lex、AMAZON を使用してください。 KendraSearchIntent。 Amazon Kendra での設定の例については Amazon Lex、「Amazon Kendra インデックス用の FAQ ボットの作成」を参照してください。

トピック

インデックスのクエリ

インデックスを検索すると、 Amazon Kendra ドキュメントに関して提供されたすべての情報を使用して、入力された検索用語に最も関連性の高いドキュメントが決定されます。 Amazon Kendra 考慮される項目には次のようなものがあります。

- ドキュメントのテキスト (本文)。
- ドキュメントのタイトル。
- 検索可能としてマークしたカスタムテキストフィールド。

- 指定した日付フィールドは、ドキュメントの「鮮度」を決定するために使用する必要があります。
- 関連情報を提供可能なその他のフィールド。

Amazon Kendra また、検索に設定したフィールド/属性フィルターに基づいてレスポンスをフィルターすることもできます。例えば、「department」というカスタムフィールドがある場合、「legal」という部門からのドキュメントのみを返すようにレスポンスをフィルタリングできます。詳細については、「Custom fields or attributes」を参照してください。

返された検索結果は、 Amazon Kendra ドキュメントごとに決定される関連性によってソートされます。結果はページ割りされ、ユーザーにページを一度に表示できます。

インデックスに登録したドキュメントを検索するには Amazon Lex、 Amazon Kendra AMAZON を使用してください。 KendraSearchIntent。 Amazon Kendra での設定の例については Amazon Lex、「Amazon Kendra インデックス用の FAQ ボットの作成」を参照してください。

次の例は、インデックスを検索する方法を示しています。 Amazon Kendra クエリに最も適した検索結果のタイプ (回答、文書、質問と回答) を決定します。特定のタイプの検索応答 (回答、文書、質問と回答) Amazon Kendra をクエリに返すようには設定できません。

クエリのレスポンスの詳細については、クエリレスポンスとレスポンスタイプ を参照してください。

前提条件

クエリ API を使用してインデックスをクエリする前に、次の操作を行います。

- インデックスに必要なアクセス許可を設定して、データソースに接続するか、ドキュメントを一括でアップロードします。詳細については、「IAM roles」を参照してください。インデックスとデータソースコネクタを作成するために API を呼び出す場合、ドキュメントを一括でアップロードする場合は、ロールの Amazon リソースネームを使用します。
- AWS Command Line Interface、SDK をセットアップするか、コンソールにアクセスしてください。 Amazon Kendra 詳細については、「セットアップ Amazon Kendra」を参照してください。
- インデックスを作成してドキュメントのデータソースに接続するか、ドキュメントを一括でアップロードします。詳細については、「Creating an index」および「Creating a data source connector」を参照してください。

インデックスの検索 (コンソール)

Amazon Kendra コンソールを使用してインデックスを検索してテストできます。クエリを作成して結果を確認できます。

コンソールでインデックスを検索するには

1. AWS Management Console にサインインし、http://console.aws.amazon.com/kendra/ Amazon Kendra でコンソールを開きます。
2. ナビゲーションペインで、[Indexes] (インデックス) をクリックします。
3. インデックスを選択します。
4. ナビゲーションメニューで、インデックスを検索するオプションを選択します。
5. テキストボックスにクエリを入力し、Enter キーを押します。
6. Amazon Kendra 検索の結果を返します。

また、サイドパネルの電球アイコンを選択すると、検索のクエリ ID を取得できます。

インデックスの検索 (SDK)

Python または Java でインデックスを検索するには

- 次の例では、インデックスを検索します。`query` の値を検索クエリに、`index_id` または `indexId` を検索するインデックスのインデックス識別子に変更します。

  クエリ API を呼び出すと、レスポンス要素の一部として検索のクエリ ID を取得することもできます。

  Python

```
import boto3
import pprint

kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the query text
query = "query text"
```

```
response = kendra.query(
        QueryText = query,
        IndexId = index_id)

print("\nSearch results for query: " + query + "\n")

for query_result in response["ResultItems"]:

    print("------------------")
    print("Type: " + str(query_result["Type"]))

    if query_result["Type"]=="ANSWER" or
 query_result["Type"]=="QUESTION_ANSWER":
        answer_text = query_result["DocumentExcerpt"]["Text"]
        print(answer_text)

    if query_result["Type"]=="DOCUMENT":
        if "DocumentTitle" in query_result:
            document_title = query_result["DocumentTitle"]["Text"]
            print("Title: " + document_title)
        document_text = query_result["DocumentExcerpt"]["Text"]
        print(document_text)

    print("------------------\n\n")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.QueryRequest;
import software.amazon.awssdk.services.kendra.model.QueryResponse;
import software.amazon.awssdk.services.kendra.model.QueryResultItem;

public class SearchIndexExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String query = "query text";
        String indexId = "index-id";

        QueryRequest queryRequest = QueryRequest
```

```
            .builder()
            .queryText(query)
            .indexId(indexId)
            .build();

        QueryResponse queryResponse = kendra.query(queryRequest);

        System.out.println(String.format("\nSearch results for query: %s",
 query));
        for(QueryResultItem item: queryResponse.resultItems()) {
            System.out.println("----------------------");
            System.out.println(String.format("Type: %s", item.type()));

            switch(item.type()) {
                case QUESTION_ANSWER:
                case ANSWER:
                    String answerText = item.documentExcerpt().text();
                    System.out.println(answerText);
                    break;
                case DOCUMENT:
                    String documentTitle = item.documentTitle().text();
                    System.out.println(String.format("Title: %s",
 documentTitle));
                    String documentExcerpt = item.documentExcerpt().text();
                    System.out.println(String.format("Excerpt: %s",
 documentExcerpt));
                    break;
                default:
                    System.out.println(String.format("Unknown query result type:
 %s", item.type()));

            }

            System.out.println("----------------------\n");
        }
    }
}
```

インデックスの検索 (Postman)

Postman Amazon Kendra を使用してインデックスのクエリとテストを行うことができます。

Postman を使用してインデックスを検索するには

1. Postman で新しいコレクションを作成し、リクエストタイプを [POST] に設定します。
2. エンドポイントの URL を入力します。例えば、https://kendra.<region>.amazonaws.com などです。
3. [認証] タブを選択し、次の情報を入力します。

   - [タイプ] - [AWS 署名] を選択します。
   - AccessKey IAM —ユーザーを作成したときに生成されたアクセスキーを入力します。
   - SecretKey IAM —ユーザを作成したときに生成されたシークレットキーを入力します。
   - AWS 地域 — インデックスの地域を入力します。例えば、us-west-2 などです。
   - [サービス名] - [kendra] と入力します。大文字と小文字を区別するため、小文字にする必要があります。

   > **Warning**
   >
   > サービス名を正しく入力しなかった場合や、小文字を使用しなかった場合は、[送信] を選択してリクエストを送信すると、「認証情報の範囲は、正しいサービス「kendra」に限定する必要があります」というエラーが表示されます。
   > また、正しいアクセスキーとシークレットキーが入力されたことを確認する必要があります。

4. [ヘッダー] タブを選択し、次のキーと値の情報を入力します。

   - キー: X-Amz-Target

     値:com.amazonaws.kendra。 AWSKendraFrontendService.Query
   - キー: Content-Encoding

     値: amz-1.0
5. [本文] タブを選択し、次の操作を行います。

   - リクエストの本文の 未加工の JSON タイプを選択します。
   - インデックス ID とクエリテキストを含む JSON を入力します。

```
{
    "IndexId": "index-id",
    "QueryText": "enter a query here"
```

```
}
```

> **Warning**
>
> JSON が正しいインデントを使用していない場合、「」というエラーが表示されます。SerializationExceptionJSON のインデントを確認してください。

6. [送信] (右上付近にあります) を選択します。

高度なクエリ構文による検索

高度なクエリ構文や演算子を使用すると、単純なキーワードクエリや自然言語クエリよりも具体的なクエリを作成できます。これには、範囲、ブール値、ワイルドカードなどが含まれます。演算子を使用すると、クエリにコンテキストを追加して、検索結果をさらに絞り込むことができます。

Amazon Kendra 以下の演算子をサポートします。

- ブール値: 検索を限定または拡大するロジック。例えば、`amazon AND sports` は、両方の語句を含むドキュメントのみを検索するように検索を限定します。
- かっこ: ネストされたクエリ語句を優先順に読み取ります。例えば、`(amazon AND sports) NOT rainforest` は、`NOT rainforest` の前に `(amazon AND sports)` を読み取ります。
- 範囲: 日付または数値の範囲値。範囲は、包含、除外、制限なしのいずれでも構いません。例えば、最終更新日が 2020 年 1 月 1 日から 2020 年 12 月 31 日まで (両端の日付を含む) のドキュメントを検索できます。
- フィールド: 特定のフィールドを使用して、検索対象を絞り込みます。例えば、「場所」のフィールドに「米国」が含まれるドキュメントを検索できます。
- ワイルドカード: テキストの文字列との部分一致。たとえば、`Cloud*`一致する可能性がありますCloudFormation。 Amazon Kendra 現在のところ、末尾のワイルドカードのみがサポートされています。
- 完全一致引用符: テキストの文字列との完全一致。例: `"Amazon Kendra" "pricing"` を含むドキュメント。

上のいずれかの演算子を組み合わせて使用できます。

演算子や非常に複雑なクエリを過度に使用すると、クエリのレイテンシーに影響を与える可能性があるので注意してください。ワイルドカードは、レイテンシーの点で非常にコストのかかる演算子の

1 つです。一般的に、使用する語句や演算子が多いほど、レイテンシーへの影響が大きくなります。待ち時間に影響するその他の要因には、インデックスに登録されるドキュメントの平均サイズ、インデックスのサイズ、検索結果のフィルタリング、インデックスにかかる全体的な負荷などがあります。 Amazon Kendra

ブール値

ブール値演算子の AND、OR、NOT を使用して、単語を組み合わせたり、除外したりすることができます。

ブール演算子を使用した例を次に示します。

amazon AND sports

Amazon Prime Video のスポーツまたはその他の類似コンテンツなど、テキストに「amazon」と「sports」の両方の語句を含む検索結果を返します。

sports OR recreation

テキストに「sports」や「recreation」、または両方の語句を含む検索結果を返します。

amazon NOT rainforest

テキストに「amazon」の語句を含むが、「rainforest」の語句は含まない検索結果を返します。これは、アマゾンの熱帯雨林に関するドキュメントではなく、Amazon という会社に関するドキュメントを検索するためのものです。

括弧

かっこを使うと、ネストされた単語を優先順に検索できます。 Amazon Kendra 括弧はクエリの読み方を示しています。

かっこ演算子を使用した例を次に示します。

(amazon AND sports) NOT rainforest

テキストに「amazon」と「sports」の両方の語句を含むが、「rainforest」の語句は含まないドキュメントを返します。これは Amazon Prime Video スポーツまたはその他の類似コンテンツを検索するためのもので、アマゾンの熱帯雨林でのアドベンチャースポーツを検索するためのものではありません。かっこは、NOT rainforest の前に amazon AND sports を読み取る必要があることを示す

のに役立ちます。このクエリは amazon AND (sports NOT rainforest) と読み取るものではありません。

(amazon AND (sports OR recreation)) NOT rainforest

「sports」や「recreation」、または両方の語句を含む検索結果を返します。ただし、「rainforest」の語句は含まれません。これは Amazon Prime Video スポーツまたはレクリエーションを検索するためのもので、アマゾンの熱帯雨林でのアドベンチャースポーツを検索するためのものではありません。かっこは、sports OR recreation は「amazon」と組み合わせる前に読み取る必要があり、NOT rainforest の前に読み取る必要があることを示すのに役立ちます。このクエリは amazon AND (sports OR (recreation NOT rainforest)) と読み取るものではありません。

範囲

値の範囲を使用して、検索結果を絞り込むことができます。属性と範囲の値を指定します。日付タイプでも数値タイプでも構いません。

日付範囲は、次の形式になります。

- Epoch
- YYYY
- YYYY-mm
- YYYY-mm-dd
- YYYY-mm-dd'T'HH

また、範囲の下限値と上限値を含めるかどうかを指定できます。

範囲演算子を使用した例を次に示します。

_processed_date:>2019-12-31 AND _processed_date:<2021-01-01

2020 年 (2019 年 12 月 31 日より後で、かつ 2021 年 1 月 1 日より前) に処理されたドキュメントを返します。

_processed_date:>=2020-01-01 AND _processed_date:<=2020-12-31

2020 年 (2020 年 1 月 1 日から 2020 年 12 月 31 日まで (両端の日付を含む)) に処理されたドキュメントを返します。

_document_likes:<1

「いいね」がゼロまたはユーザーからのフィードバックがない、つまり「いいね」が 1 件未満のドキュメントを返します。

ある範囲を、指定した範囲の値を含むものとして扱うのか、含まないものとして扱うのかを指定できます。

包含

_last_updated_at:[2020-01-01 TO 2020-12-31]

最終更新日が 2020 年のドキュメントを返します。これには 2020 年 12 月 1 日と 2020 年 12 月 31 日が含まれます。

除外

_last_updated_at:{2019-12-31 TO 2021-01-01}

最終更新日が 2020 年のドキュメントを返します。これには 2019 年 12 月 31 日と 2021 年 1 月 1 日は含まれません。

包含または除外ではなく、制限なしの範囲の場合は、< and > 演算子を使用します。例えば、次のようになります: `_last_updated_at:>2019-12-31 AND _last_updated_at:<2021-01-01`

フィールド

特定のフィールドの値と一致するドキュメントのみを返すように検索を限定できます。フィールドは、任意のタイプに指定できます。

フィールドレベルのコンテキスト演算子の使用例を次に示します。

status:"Incomplete" AND financial_year:2021

会計年度が 2021 年のドキュメントで、ステータスが未完了のものを返します。

(sports OR recreation) AND country:"United States" AND level:"professional"

米国内のプロスポーツまたはレクリエーションに関するドキュメントを返します。

ワイルドカード

ワイルドカード演算子を使用すると、単語やフレーズのバリエーションを考慮して検索範囲を広げることができます。これは名前のバリエーションを検索する場合に便利です。 Amazon Kendra 現在の

ところ、末尾のワイルドカードのみがサポートされています。末尾のワイルドカードのプレフィックス文字の数は 2 文字以上である必要があります。

ワイルドカード演算子を使用した例を次に示します。

Cloud*

CloudFormation やなどのバリアントを含むドキュメントを返します。 CloudWatch

kendra*aws

kendra.amazonaws などのバリアントを含むドキュメントを返します。

kendra*aws*

kendra.amazonaws.com などのバリアントを含むドキュメントを返します。

完全一致引用符

引用符を使用すると、テキストの一部と完全に一致するものを検索できます。

引用符の使用例を次に示します。

"Amazon Kendra" "pricing"

「Amazon Kendra」と「pricing」の両方の語句を含むドキュメントを返します。結果を返すには、ドキュメント「Amazon Kendra」と「pricing」の両方が含まれている必要があります。

"Amazon Kendra" "pricing" cost

「Amazon Kendra」と「pricing」の両方の語句を含み、オプションで「cost」の語句を含むドキュメントを返します。結果を返すには、ドキュメント「Amazon Kendra」と「pricing」の両方が含まれている必要がありますが、「cost」は含まれていない場合もあります。

無効なクエリ構文

Amazon Kendra クエリ構文に問題がある場合や、クエリが現在でサポートされていない場合は警告が表示されます Amazon Kendra。詳細については、「API documentation for query warnings」を参照してください。

次のクエリは、無効なクエリ構文の例です。

_last_updated_at:<2021-12-32

無効な日付です。32 日は、 Amazon Kendraが使用しているグレゴリオ暦には存在しません。

`_view_count:ten`

無効な数値です。数値を表すには、数字を使用する必要があります。

`nonExistentField:123`

無効なフィールド検索です。フィールド検索を使用するには、そのフィールドが存在している必要があります。

`Product:[A TO D]`

無効な範囲です。範囲には、数値または日付を使用する必要があります。

`OR Hello`

無効なブール値です。演算子は語句と一緒に使用し、語句の間に記述する必要があります。

各言語での検索

サポートされている言語でドキュメントを検索できます。に言語コードを渡すと、AttributeFilterフィルタされたドキュメントが選択した言語で返されます。クエリは、サポートされている言語で入力できます。

言語を指定しない場合、 Amazon Kendra デフォルトで英語でドキュメントをクエリします。コードを含む、サポートされている言語の詳細については、英語以外の言語でドキュメントを追加するを参照してください。

サポートされている言語のドキュメントをコンソールで検索するには、インデックスを選択し、ナビゲーションメニューから、インデックスを検索するオプションを選択します。[検索設定] を選択して、ドキュメントを返す言語を選択します。次にドロップダウンの [言語] から言語を選択します。

次の例では、スペイン語でドキュメントを検索する方法を示しています。

コンソールでスペイン語のインデックスを検索するには

1. AWS Management Console にサインインし、http://console.aws.amazon.com/kendra/ Amazon Kendra のコンソールを開きます。
2. ナビゲーションメニューで、[インデックス] を選択し、目的のインデックスを選択します。
3. ナビゲーションメニューで、インデックスを検索するオプションを選択します。

4. [検索設定] で、[言語] ドロップダウンを選択し、スペイン語を選択します。
5. テキストボックスにクエリを入力し、Enter キーを押します。
6. Amazon Kendra 検索結果をスペイン語で返します。

CLI、Python または Java を使用してインデックスをスペイン語で検索するには

- 次の例では、インデックスをスペイン語で検索します。searchString の値を検索クエリに、indexID の値を検索するインデックスのインデックス識別子に変更します。スペイン語の言語コードは es です。これを独自の言語コードに置き換えることができます。

  CLI

```
{
  "EqualsTo":{
    "Key": "_language_code",
    "Value": {
    "StringValue": "es"
    }
  }
}
```

  Python

```
import boto3
import pprint

kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the query text
query = "search-string"

# Includes the index ID, query text, and language attribute filter
response = kendra.query(
        QueryText = query,
        IndexId = index_id,
        AttributeFilter = {
            "EqualsTo": {
                "Key": "_language_code",
                "Value": {
```

```
                    "StringValue": "es"
                    }
                }
            })

print ("\nSearch results|Resultados de la búsqueda: " + query + "\n")

for query_result in response["ResultItems"]:

    print("-------------------")
    print("Type: " + str(query_result["Type"]))

    if query_result["Type"]=="ANSWER" or
 query_result["Type"]=="QUESTION_ANSWER":
        answer_text = query_result["DocumentExcerpt"]["Text"]
        print(answer_text)

    if query_result["Type"]=="DOCUMENT":
        if "DocumentTitle" in query_result:
            document_title = query_result["DocumentTitle"]["Text"]
            print("Title: " + document_title)
        document_text = query_result["DocumentExcerpt"]["Text"]
        print(document_text)

    print("------------------\n\n")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.QueryRequest;
import software.amazon.awssdk.services.kendra.model.QueryResponse;
import software.amazon.awssdk.services.kendra.model.QueryResultItem;

public class SearchIndexExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String query = "searchString";
        String indexId = "indexID";
```

```
        QueryRequest queryRequest = QueryRequest.builder()
            .queryText(query)
            .indexId(indexId)
            .attributeFilter(
                 AttributeFilter.builder()
                     .withEqualsTo(
                         DocumentAttribute.builder()
                             .withKey("_language_code")
                             .withValue("es")
                             .build())
                     .build())
            .build();

        QueryResponse queryResponse = kendra.query(queryRequest);

        System.out.println(String.format("\nSearch results|
                                          Resultados de la búsqueda: %s",
query));
        for(QueryResultItem item: queryResponse.resultItems()) {
            System.out.println("----------------------");
            System.out.println(String.format("Type: %s", item.type()));

            switch(item.type()) {
                case QUESTION_ANSWER:
                case ANSWER:
                    String answerText = item.documentExcerpt().text();
                    System.out.println(answerText);
                    break;
                case DOCUMENT:
                    String documentTitle = item.documentTitle().text();
                    System.out.println(String.format("Title: %s",
documentTitle));
                    String documentExcerpt = item.documentExcerpt().text();
                    System.out.println(String.format("Excerpt: %s",
documentExcerpt));
                    break;
                default:
                    System.out.println(String.format("Unknown query result type:
%s", item.type()));

            }

            System.out.println("----------------------\n");
```

```
        }
    }
}
```

パッセージを取得する

Retrieve API を、検索拡張生成 (RAG) システムのレトリバーとして使用できます。

RAG システムは、生成系人工知能を使用して質問応答アプリケーションを構築します。RAG システムは、レトリバーと大規模言語モデル (LLM) で構成されています。クエリが指定されると、レトリバーはドキュメントのコーパスから最も関連性が強いテキストのコーパスを特定し、それを LLM に送り、最も有用な答えを提供します。次に、LLM は関連するテキストチャンクまたはパッセージを分析し、クエリに対する包括的なレスポンスを生成します。

Retrieve API は、パッセージと呼ばれるテキストのチャンクまたは抜粋を検索し、最も関連性の強い最上位のパッセージをクエリに返します。

Query API と同様に、Retrieve API もセマンティック検索を使用して関連情報を検索します。セマンティック検索では、検索クエリのコンテキストに加えて、インデックスが作成されたドキュメントから入手可能なすべての情報が考慮されます。ただし、デフォルトでは Query API は最大 100 個のトークンワードの抜粋のみを返します。Retrieve API を使用すると、最大 200 個のトークンワードの長いパッセージと、最大 100 個のセマンティックに関連するパッセージを取得できます。これには、インデックスからの質問応答や FAQ タイプの回答は含まれません。パッセージは、複数のドキュメントや同じドキュメントの複数の部分からセマンティックに抽出できるテキストの抜粋です。極端なケースで、Retrieve API を使用してドキュメントからパッセージが生成されない場合は、代わりに Query API とそのレスポンスタイプを使用できます。

Retrieve API を使用すると、次のことを実行できます。

- インデックスレベルでのオーバーライドブースト
- ドキュメントフィールドまたは属性に基づくフィルタリング
- ユーザーまたはグループのドキュメントへのアクセスに基づいたフィルタリング
- 取得したドキュメントの結果の信頼度スコアバケットを表示します。信頼度バケットは、 Amazon Kendra がそのレスポンスがクエリに関連している信頼度を示す相対的なランク付けを提供します。

> **Note**
>
> 信頼スコアバケットは現在、英語でのみ利用可能です。

有用な追加情報が得られる可能性のある特定のフィールドをレスポンスに含めることもできます。

Retrieve API では現在、Query API がサポートしているすべての機能をサポートしているわけではありません。高度なクエリ構文を使用したクエリ、クエリのスペル修正の提案、ファセット、クエリの提案による検索クエリの自動入力、増分学習はサポートされていません。API にすべての機能が適用されるわけではないことに注意してください。RetrieveRetrieveAPI のfuture リリースは、このガイドに記載されます。

Retrieve API は、インデックスに設定したクエリ容量ユニットの数を共有します。1 つのキャパシティーユニットに含まれる内容と、インデックスのデフォルトの基本容量の詳細については、「Adjusting capacity」を参照してください。

> **Note**
>
> Amazon Kendra Developer Edition を使用している場合は容量を追加できません。容量を追加できるのは Amazon Kendra Enterprise Edition を使用する場合のみです。Developer Edition と Enterprise Edition に含まれる内容の詳細については、「Amazon Kendra Editions」を参照してください。

次の例では、Retrieve API を使用してクエリ "how does amazon kendra work?" のインデックス内のドキュメントから最も関連性の高い上位 100 件のドキュメントを取得しています。

Python

```
import boto3
import pprint

kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the query text
query = "how does amazon kendra work?"
```

```
# You can retrieve up to 100 relevant passages
# You can paginate 100 passages across 10 pages, for example
page_size = 10
page_number = 10

result = kendra.retrieve(
        IndexId = index_id,
        QueryText = query,
        PageSize = page_size,
        PageNumber = page_number)

print("\nRetrieved passage results for query: " + query + "\n")

for retrieve_result in result["ResultItems"]:

    print("-------------------")
    print("Title: " + str(retrieve_result["DocumentTitle"]))
    print("URI: " + str(retrieve_result["DocumentURI"]))
    print("Passage content: " + str(retrieve_result["Content"]))
    print("------------------\n\n")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.RetrieveRequest;
import software.amazon.awssdk.services.kendra.model.RetrieveResult;
import software.amazon.awssdk.services.kendra.model.RetrieveResultItem;

public class RetrievePassageExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String indxId = "index-id";
        String query = "how does amazon kendra work?";
        Integer pgSize = 10;
        Integer pgNumber = 10;

        RetrieveRequest retrieveRequest = retrieveRequest
            .builder()
            .indexId(indxId)
            .queryText(query)
```

```
            .pageSize(pgSize)
            .pageNumber(pgNumber)
            .build();

        RetrieveResult retrieveResult = kendra.retrieve(retrieveRequest);

        System.out.println(String.format("\nRetrieved passage results for query:
 %s", query));
        for(RetrieveResultItem item: retrieveResult.resultItems()) {
            System.out.println("----------------------");
            System.out.println(String.format("Title: %s", documentTitle));
            System.out.println(String.format("URI: %s", documentURI));
            System.out.println(String.format("Passage content: %s", content));
            System.out.println("----------------------\n");
        }
    }
}
```

インデックスの閲覧

検索クエリを入力しなくても、属性やファセット別にドキュメントをブラウズできます。 Amazon Kendra Index Browse を使用すると、ユーザーは特定のクエリに関係なく、自由にインデックスを閲覧してドキュメントを検索できます。これは、ユーザーが検索の出発点として、インデックスを幅広く閲覧するのにも役立ちます。

Index Browse は、ソートタイプを持つドキュメント属性またはファセットによる検索にのみ使用できます。Index Browse を使用して、インデックス全体を検索することはできません。クエリテキストが見つからない場合は、ドキュメント属性フィルタまたはファセット、 Amazon Kendra およびソートタイプを要求します。

Query API を使用してインデックスを参照できるようにするには、AttributeFilterまたはファセット、およびを含める必要があります。SortingConfigurationコンソールでインデックスを閲覧できるようにするには、ナビゲーションメニューの [インデックス] でインデックスを選択し、インデックスを検索するオプションを選択します。検索ボックスで、Enter キーを 2 回押します。ドロップダウンから [検索結果のフィルタリング] を選択して [フィルター] を選択し、さらにドロップダウンから [ソート] を選択してソートの種類を選択します。

次の例では、スペイン語のドキュメントのインデックスをドキュメント作成日の降順で閲覧しています。

CLI

```
aws kendra query \
--index-id "index-id" \
--attribute-filter '{
    "EqualsTo":{
      "Key": "_language_code",
      "Value": {
        "StringValue": "es"
      }
    }
  }' \
--sorting-configuration '{
    "DocumentAttributeKey": "_created_at",
    "SortOrder": "DESC"
  }'
```

Python

```
import boto3

kendra = boto3.client("kendra")

# Must include the index ID, the attribute filter, and sorting configuration
response = kendra.query(
        IndexId = "index-id",
        AttributeFilter = {
            "EqualsTo": {
                "Key": "_language_code",
                "Value": {
                    "StringValue": "es"
                    }
                }
            },
        SortingConfiguration = {
            "DocumentAttributeKey": "_created_at",
            "SortOrder": "DESC"})

print("\nSearch results|Resultados de la búsqueda: \n")

for query_result in response["ResultItems"]:

    print("-------------------")
```

```
    print("Type: " + str(query_result["Type"]))

    if query_result["Type"]=="ANSWER" or query_result["Type"]=="QUESTION_ANSWER":
        answer_text = query_result["DocumentExcerpt"]["Text"]
        print(answer_text)

    if query_result["Type"]=="DOCUMENT":
        if "DocumentTitle" in query_result:
            document_title = query_result["DocumentTitle"]["Text"]
            print("Title: " + document_title)
        document_text = query_result["DocumentExcerpt"]["Text"]
        print(document_text)

    print("------------------\n\n")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.QueryRequest;
import software.amazon.awssdk.services.kendra.model.QueryResult;
import software.amazon.awssdk.services.kendra.model.QueryResultItem;

public class SearchIndexExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();
        QueryRequest queryRequest = QueryRequest.builder()
            .withIndexId("index-id")
            .withAttributeFilter(AttributeFilter.builder()
                .withEqualsTo(DocumentAttribute.builder()
                    .withKey("_language_code")
                    .withValue(DocumentAttributeValue.builder()
                        .withStringValue("es")
                        .build())
                    .build())
                .build())
            .withSortingConfiguration(SortingConfiguration.builder()
                .withDocumentAttributeKey("_created_at")
                .withSortOrder("DESC")
                .build())
            .build());
```

```
        QueryResult queryResult = kendra.query(queryRequest);
        for (QueryResultItem item : queryResult.getResultItems()) {
            System.out.println("----------------------");
            System.out.println(String.format("Type: %s", item.getType()));

            switch (item.getType()) {
                case QueryResultType.QUESTION_ANSWER:
                case QueryResultType.ANSWER:
                    String answerText = item.getDocumentExcerpt().getText();
                    System.out.println(answerText);
                    break;
                case QueryResultType.DOCUMENT:
                    String documentTitle = item.getDocumentTitle().getText();
                    System.out.println(String.format("Title: %s", documentTitle));
                    String documentExcerpt = item.getDocumentExcerpt().getText();
                    System.out.println(String.format("Excerpt: %s",
 documentExcerpt));
                    break;
                default:
                    System.out.println(String.format("Unknown query result type:
 %s", item.getType()));
            }
            System.out.println("----------------------\n");
        }
    }
}
```

検索結果を目立たせる

ユーザーが特定のクエリを発行した際に、検索結果に特定のドキュメントを目立たせることができます。これにより、ユーザーにとって検索結果がより見やすく、目立つようになります。注目の検索結果は、通常の検索結果リストとは別に、検索ページの上部に表示されます。クエリごとに別々のドキュメントを目立たせることや、特定のドキュメントにふさわしい視認性を確保できます。

特定のクエリを特定のドキュメントにマッピングして、その結果に反映できます。クエリに完全に一致するものが含まれている場合、1 つ以上の特定のドキュメントが検索結果に表示されます。

例えば、ユーザーが「new products 2023」というクエリを発行した場合に、「What's new」と「Coming soon」というタイトルのドキュメントを選択して検索結果ページの上部に目立たせるように指定できます。これにより、新製品に関するこれらのドキュメントが、目的に合った視認性を得られるようになります。

Amazon Kendra 検索結果ページの上部に表示する結果が既に選択されている場合、検索結果は重複しません。注目の検索結果が他の検索結果よりも既に上位に表示されている場合、それが再び最初の結果としてランク付けされることはありません。

特定の検索結果を目立たせるためには、クエリに含まれるキーワードやフレーズを使用したクエリの部分一致ではなく、テキスト全文のクエリの完全一致を指定する必要があります。例えば、注目の結果セットに「Kendra」というクエリのみを指定すると、「How does Kendra semantically rank results?」などのクエリになります。注目の結果はレンダリングされません。おすすめの検索結果は、範囲が広すぎるクエリではなく、特定のクエリを対象としています。 Amazon Kendra キーワードタイプのクエリを自然に処理して、検索結果で最も有用なドキュメントをランク付けします。これにより、単純なキーワードに基づいて結果が過度に注目されるのを防ぐことができます。

ユーザーが頻繁に使用する特定のクエリがある場合は、それらのクエリを指定して注目の結果に指定できます。例えば、Amazon Kendra Analytics を使用して上位のクエリを調べて、「How does kendra semantically rank results?」など、特定のクエリが見つかったとします。また、「kendra セマンティック検索」が頻繁に使用される場合、これらのクエリは「search 101」というタイトルのドキュメントを特集する場合に便利です。Amazon Kendra

Amazon Kendra 注目結果のクエリは大文字と小文字を区別しないものとして扱います。 Amazon Kendra クエリを小文字に変換し、末尾の空白文字を 1 つのスペースに置き換えます。 Amazon Kendra 主要結果のクエリを指定すると、他のすべての文字がそのまま照合されます。

CreateFeaturedResultsSetAPI を使用して特定のクエリにマッピングする主な結果のセットを作成します。コンソールを使用する場合は、インデックスを選択し、ナビゲーションメニューで [注目の結果] を選択して注目の結果セットを作成します。1 つのインデックスにつき最大 50 セットの注目の結果、1 セットにつき 4 つの主要ドキュメント、および注目の結果セットごとに最大 49 のクエリテキストを作成できます。これらの制限は、サポートにリクエストして引き上げることができます。

複数の注目の結果セットから同じドキュメントを選択できます。ただし、複数のセットで同じ完全一致クエリテキストを使用しないでください。注目の結果に対して指定するクエリは、インデックスの注目の結果セットごとに一意である必要があります。

注目のドキュメントの選択が 4 つまでの場合は、ドキュメントの順序を調整できます。API を使用する場合、注目のドキュメントを一覧表示する順序は、注目の結果に表示される順序と同じになります。コンソールを使用すると、結果に表示するドキュメントを選択するときに、ドキュメントの順序をドラッグアンドドロップするだけで済みます。

特定のユーザーやグループが特定のドキュメントにアクセスでき、他のユーザーやグループはアクセスできないアクセスコントロールも、注目の結果を設定する際にも引き続き適用されます。これは、ユーザーコンテキストのフィルタリングにも当てはまります。例えば、ユーザー A は「Interns」企

業グループに属しているため、企業秘密に関するドキュメントにはアクセスできないはずです。ユーザー A が企業秘密文書を含むクエリを入力しても、ユーザー A の結果にはこのドキュメントは表示されません。検索結果ページの他の結果も同様です。タグを使用して、アクセスをコントロールできる Amazon Kendra リソースである注目の結果セットへのアクセスをコントロールすることもできます。

次の例では、「new products 2023」、「new products available」というクエリを「What's new」(doc-id-1) と「Coming soon」(doc-id-2) というタイトルのドキュメントにマッピングして、注目の結果セットを作成しています。

CLI

```
aws kendra create-featured-results-set \
 --featured-results-set-name 'New product docs to feature' \
 --description "Featuring What's new and Coming soon docs" \
 --index-id index-id \
 --query-texts 'new products 2023' 'new products available' \
 --featured-documents '{"Id":"doc-id-1", "Id":"doc-id-2"}'
```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Create a featured results set.")

# Provide a name for the featured results set
featured_results_name = "New product docs to feature"
# Provide an optional decription for the featured results set
description = "Featuring What's new and Coming soon docs"
# Provide the index ID for the featured results set
index = "index-id"
# Provide a list of query texts for the featured results set
queries = ['new products 2023', 'new products available']
# Provide a list of document IDs for the featured results set
featured_doc_ids = [{"Id":"doc-id-1"}, {"Id":"doc-id-2"}]

try:
```

```
    featured_results_set_response = kendra.create_featured_results_set(
        FeaturedResultsSetName = featured_results_name,
        Decription = description,
        Index = index,
        QueryTexts = queries,
        FeaturedDocuments = featured_doc_ids
    )

    pprint.pprint(featured_results_set_response)

    featured_results_set_id = featured_results_set_response["FeaturedResultsSetId"]

    while True:
        # Get the details of the featured results set, such as the status
        featured_results_set_description = kendra.describe_featured_results_set(
            Id = featured_results_set_id
        )
        status = featured_results_set_description["Status"]
        print(" Featured results set status: "+status)

except  ClientError as e:
        print("%s" % e)

print("Program ends.")
```

HTML の表形式検索

Amazon Kendraの表形式検索機能では、HTML ドキュメントに埋め込まれた表から回答を検索して抽出できます。インデックスを検索すると、 Amazon Kendra クエリに関連する場合は表からの抜粋が含まれ、有用な情報が表示されます。

Amazon Kendra テーブル内の有用な情報を含め、文書の本文に含まれるすべての情報を調べます。例えば、インデックスには、運用コスト、収入、その他の財務情報に関する表を含むビジネスレポートが含まれています。「2020 年から 2022 年までの年間運用コストはいくらですか?」というクエリの場合 では、関連するテーブル列「業務 (百万米ドル)」と「会計年度」、および 2020 年、2021 年、2022 Amazon Kendra 年の収益値を含むテーブル行を含むテーブルからの抜粋を返すことができます。結果には、表の抜粋とドキュメントのタイトル、ドキュメント全文へのリンク、選択したドキュメントフィールドが含まれます。

表の抜粋は、情報が表の 1 つのセルにあるか、複数のセルにあるかにかかわらず、検索結果に表示できます。たとえば、 Amazon Kendra 次の各種類のクエリに合わせた表の抜粋を表示できます。

- 「highest interest rate credit card in 2020」
- 「highest interest rate credit card from 2020-2022」
- 「top 3 highest interest rate credit cards in 2020-2022」
- 「credit cards with interest rates less than 10%」
- 「all available low interest credit cards」

Amazon Kendra クエリに最も関連性のある 1 つまたは複数のテーブルセルを強調表示します。最も関連性の高いセルと、それに対応する行、列、列名が検索結果に表示されます。表の抜粋には、クエリに関連する表セルの数と元の表で利用可能な列の数に応じて、最大 5 つの列と 3 つの行が表示されます。表の抜粋では、最も関連性の高いセルが、その次に関連性の高いセルとともに表示されます。

レスポンスには、表の回答がクエリにどの程度関連しているかを示す信頼バケット (MEDIUM、HIGH、VERY_HIGH) が含まれます。表のセルの値の信頼度が VERY_HIGH の場合、その値が「最上位の回答」になり、強調表示されます。表のセルの値の信頼度が HIGH の場合、その値が強調表示されます。表のセルの値の信頼度が MEDIUM の場合、その値は強調表示されません。表の回答に対する全体的な信頼度は、レスポンスで返されます。例えば、表に信頼度 HIGH の表のセルのほとんどが含まれている場合、表の回答で返される全体的な信頼度は HIGH の信頼度です。

デフォルトでは、表にドキュメントの他の構成要素よりも重要度や重みが付けられることはありません。文書内では、テーブルがクエリと少しだけ関連しているけれども、関連性の高い段落がある場合は、 Amazon Kendra その段落の抜粋を返します。検索結果には、同じドキュメントまたは他のドキュメント内の、最良に近いの回答と最も有用な情報を提供するコンテンツが表示されます。表の信頼度が MEDIUM の信頼度を下回ると、そのデーブルの抜粋はレスポンスに返されません。

既存のインデックスで表形式検索を使用するには、コンテンツのインデックスを再作成する必要があります。

Amazon Kendra 表形式検索では、シノニム (カスタムシノニムを含む) がサポートされます。Amazon Kendra table タグ内の HTML テーブルを含む英語のドキュメントのみをサポートします。

次の例は、クエリ結果に含まれる表の抜粋を示しています。表の抜粋を含む、クエリレスポンスを含むサンプル JSON を表示するには、「Query responses and types」を参照してください。

Python

```
import boto3
import pprint
```

```
kendra = boto3.client("kendra")

# Provide the index ID
index_id = <index-id>
# Provide the query text
query = "search string"

response = kendra.query(
        QueryText = query,
        IndexId = index_id)

print("\nSearch results for query: " + query + "\n")

for query_result in response["ResultItems"]:

    print("------------------")
    print("Type: " + str(query_result["Type"]))
    print("Type: " + str(query_result["Format"]))

    if query_result["Type"]=="ANSWER" and query_result["Format"]=="TABLE":
        answer_table = query_result["TableExcerpt"]
        print(answer_table)

    if query_result["Type"]=="ANSWER" and query_result["Format"]=="TEXT":
        answer_text = query_result["DocumentExcerpt"]
        print(answer_text)

    if query_result["Type"]=="QUESTION_ANSWER":
        question_answer_text = query_result["DocumentExcerpt"]["Text"]
        print(question_answer_text)

    if query_result["Type"]=="DOCUMENT":
        if "DocumentTitle" in query_result:
            document_title = query_result["DocumentTitle"]["Text"]
            print("Title: " + document_title)
        document_text = query_result["DocumentExcerpt"]["Text"]
        print(document_text)

    print("------------------\n\n")
```

Java

```
package com.amazonaws.kendra;
```

```
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.QueryRequest;
import software.amazon.awssdk.services.kendra.model.QueryResponse;
import software.amazon.awssdk.services.kendra.model.QueryResultItem;

public class SearchIndexExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String query = "search string";
        String indexId = "index-id";

        QueryRequest queryRequest = QueryRequest
            .builder()
            .queryText(query)
            .indexId(indexId)
            .build();

        QueryResponse queryResponse = kendra.query(queryRequest);

        System.out.println(String.format("\nSearch results for query: %s", query));
        for(QueryResultItem item: queryResponse.resultItems()) {
            System.out.println("----------------------");
            System.out.println(String.format("Type: %s", item.type()));
            System.out.println(String.format("Format: %s", item.format()));

            switch(item.format()) {
                case TABLE:
                    String answerTable = item.TableExcerpt();
                    System.out.println(answerTable);
                    break;
            }

            switch(item.format()) {
                case TEXT:
                    String answerText = item.DocumentExcerpt();
                    System.out.println(answerText);
                    break;
            }

            switch(item.type()) {
                case QUESTION_ANSWER:
                    String questionAnswerText = item.documentExcerpt().text();
```

```
                    System.out.println(questionAnswerText);
                    break;
                case DOCUMENT:
                    String documentTitle = item.documentTitle().text();
                    System.out.println(String.format("Title: %s", documentTitle));
                    String documentExcerpt = item.documentExcerpt().text();
                    System.out.println(String.format("Excerpt: %s",
 documentExcerpt));
                    break;
                default:
                    System.out.println(String.format("Unknown query result type:
 %s", item.type()));

            }

            System.out.println("----------------------\n");
        }
    }
}
```

クエリの提案

Amazon Kendra の [クエリの提案] は、ユーザーが検索クエリをより速く入力し、検索をガイドするのに役立ちます。

Amazon Kendra 以下のいずれかに基づいて、ユーザーに関連するクエリを提案します。

- クエリ履歴またはクエリログでよく使われるクエリ
- ドキュメントフィールド/属性の内容

SuggestionTypes を QUERY または DOCUMENT_ATTRIBUTES のいずれかを設定し、GetQuerySuggestions を呼び出すことにより、クエリ履歴またはドキュメントフィールドを使用する際の優先順位を設定できます。デフォルトでは、 Amazon Kendra クエリ履歴に基づいて候補が表示されます。UpdateQuerySuggestionsConfig呼び出し時にクエリ履歴とドキュメントフィールドの両方がアクティブになっていて、SuggestionTypesドキュメントフィールドを使用するプリファレンスを設定していない場合は、 Amazon Kendra クエリ履歴が使用されます。

コンソールを使用すると、クエリの提案をクエリ履歴またはドキュメントフィールドのいずれかに基づくことができます。最初にインデックスを選択し、ナビゲーションメニューの [エンリッチメント] で [クエリの提案] を選択します。次に、[クエリの提案の設定] を選択します。クエリの提案を設

定すると、検索コンソールが表示され、右側のパネルで [クエリ履歴] または [ドキュメントフィールド] のいずれかを選択すると、検索バーに検索クエリを入力できます。

デフォルトでは、クエリ履歴とドキュメントフィールドを使用したクエリの提案、はどちらも追加料金なしで有効化されています。これらのタイプのクエリの提案は、UpdateQuerySuggestionsConfig API を使用していつでも無効化できます。クエリ履歴に基づくクエリの提案を無効にするには、UpdateQuerySuggestionsConfig を呼び出す際に Mode を DISABLED に設定します。ドキュメントフィールドに基づくクエリの提案を無効にするには、ドキュメントフィールド設定で AttributeSuggestionsMode を INACTIVE に設定し、次に UpdateQuerySuggestionsConfig> を呼び出します。コンソールを使用している場合は、[クエリの提案の設定] でクエリの提案を無効にできます。

クエリ候補では大文字と小文字は区別されません。 Amazon Kendra クエリプレフィックスと推奨クエリを小文字に変換し、一重引用符と二重引用符をすべて無視し、複数の空白文字を 1 つのスペースに置き換えます。 Amazon Kendra 他のすべての特殊文字はそのままマッチします。 Amazon Kendra ユーザーが 2 文字未満または 60 文字以上を入力しても、候補は表示されません。

トピック

クエリ履歴を使用したクエリの提案

トピック

クエリ履歴またはクエリログでよく使用されるクエリに基づいて、ユーザーに関連するクエリを提案するように選択できます。 Amazon Kendra ユーザーが検索し、そのクエリから学習したすべてのクエリを使用して、ユーザーへの提案を行います。 Amazon Kendra ユーザーがクエリを入力し始めると、よく使われるクエリを提案します。 Amazon Kendra クエリのプレフィックスまたは最初の数文字が、ユーザーがクエリとして入力し始めたものと一致する場合、クエリを提案します。

例えば、ユーザーが「upcoming events」というクエリの入力を開始したとします。 Amazon Kendra はクエリ履歴から、多くのユーザーが「upcoming events 2050」を何度も検索していることを学習します。検索バーの真下に「upcoming events 2050」が表示され、検索クエリが自動入力されます。ユーザーがこのクエリの提案を選択すると、検索結果に「New events: What's happening in 2050」というドキュメントが表示されます。

Amazon Kendra 適格なクエリをどのように選択してユーザーに提案するかを指定できます。たとえば、候補となるクエリは、10 人以上のユニークユーザー (デフォルトは 3 人) が検索したこと、過去 30 日以内に検索されたこと、禁止リストにある単語や語句が含まれていないことを指定できます。Amazon Kendra クエリには少なくとも 1 つの検索結果があり、4 文字を超える単語が 1 つ以上含まれている必要があります。

提案用のクエリを選択するための設定

UpdateQuerySuggestionsConfig API を使用することにより、次の設定を構成して、提案用のクエリを選択できます。

- モード - クエリ履歴を使用するクエリの低名は、ENABLED または LEARN_ONLY のいずれかです。Amazon Kendra は、デフォルトでクエリの提案を有効にします。LEARN_ONLY は、クエリの提案を無効にします。オフにすると、 Amazon Kendra 候補は引き続き学習されますが、ユーザーへのクエリの候補は表示されません。
- クエリログの期間 - クエリログの期間内で表示された最新のクエリ。タイムウィンドウは、現在の日から過去の日までの日数の整数値です。
- ユーザー情報のないクエリ - TRUE に設定し、すべてのクエリを含めるか、FALSE に設定してユーザー情報を含むクエリのみを含めます。この設定は、ユーザーがクエリを発行したときに、検索アプリケーションにユーザー ID などのユーザー情報が含まれている場合に使用できます。この設定は、デフォルトではクエリに関連する特定のユーザー情報がない場合、クエリをフィルタリングしません。ただし、この設定を使用して、ユーザー情報を含むクエリに基づく提案のみを行うことができます。
- 一意のユーザー - ユーザーに的確な提案をするために、クエリを検索する必要がある一意のユーザーの最小数。この数値は整数値です。
- クエリ数 - ユーザーに的確な提案をするために、クエリを検索する必要がある最小検索回数。この数値は整数値です。

これらの設定は、ユーザーに提案する一般的なクエリとしてクエリを選択する方法に影響します。設定の調整方法は、特定のニーズによって異なります。以下に例を挙げます。

- 通常、ユーザーが平均月 1 回検索する場合、クエリログ期間の日数を 30 に設定できます。この設定を使用すると、期間の中で陳腐化する前に、ユーザーの最近のクエリのほとんどをキャプチャできます。
- ユーザー情報を含むクエリの数が少なく、少ないサンプル数に基づいてクエリを提案しない場合は、すべてのユーザーを含むようにクエリを設定できます。
- [popular queries] (人気のあるクエリ) を 10 人以上のユニークユーザーに 100 回以上検索されたものと定義する場合は、ユニークユーザーを 10、クエリ数を 100 に設定します。

> **Warning**
>
> 設定の変更はすぐには反映されない場合があります。DescribeQuerySuggestionsConfig API を使用して、設定の変更を追跡できます。更新した設定が有効になるまでの時間は、更新した内容やインデックス内の検索クエリの数によって異なります。 Amazon Kendra は、設定を変更した後、またはブロックリストを適用した後、24 時間ごとに自動的に提案を更新します。

CLI

クエリの提案を取得するには

```
aws kendra get-query-suggestions \
 --index-id index-id \
 --query-text "query-text" \
 --suggestion-types '["QUERY"]' \
 --max-suggestions-count 1 // If you want to limit the number of suggestions
```

クエリの提案を更新するには

例えば、クエリログタイムウィンドウと、クエリを検索する必要がある最小回数を変更するには、以下の操作を行います。

```
aws kendra update-query-suggestions-config \
 --index-id index-id \
 --query-log-look-back-window-in-days 30 \
 --minimum-query-count 100
```

Python

クエリの提案を取得するには

```
import boto3
from botocore.exceptions import ClientError

kendra = boto3.client("kendra")

print("Get query suggestions.")

# Provide the index ID
index_id = "index-id"

# Provide the query text
query_text = "query"

# Provide the query suggestions type
query_suggestions_type = "QUERY"


# If you want to limit the number of suggestions
num_suggestions = 1

try:
    query_suggestions_response = kendra.get_query_suggestions(
        IndexId = index_id,
        QueryText = query_text,
        SuggestionTypes = query_suggestions_type,
        MaxSuggestionsCount = num_suggestions
    )

    # Print out the suggestions you received
    if ("Suggestions" in query_suggestions_response.keys()) {
        for (suggestion: query_suggestions_response["Suggestions"]) {
            print(suggestion["Value"]["Text"]["Text"]);
        }
    }

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

クエリの提案を更新するには

例えば、クエリログタイムウィンドウと、クエリを検索する必要がある最小回数を変更するには、以下の操作を行います。

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Updating query suggestions settings/configuration for an index.")

# Provide the index ID
index_id = "index-id"

# Configure the settings you want to update
minimum_query_count = 100
query_log_look_back_window_in_days = 30

try:
    kendra.update_query_suggestions_config(
        IndexId = index_id,
        MinimumQueryCount = minimum_query_count,
        QueryLogLookBackWindowInDays = query_log_look_back_window_in_days
    )

    print("Wait for Amazon Kendra to update the query suggestions.")

    while True:
        # Get query suggestions description of settings/configuration
        query_sugg_config_response = kendra.describe_query_suggestions_config(
            IndexId = index_id
        )

        # If status is not UPDATING, then quit
        status = query_sugg_config_response["Status"]
        print(" Updating query suggestions config. Status: " + status)
        if status != "UPDATING":
            break
        time.sleep(60)

except ClientError as e:
```

```
        print("%s" % e)

print("Program ends.")
```

クエリ履歴を保持したままで提案を消去する

ClearQuerySuggestions API を使用して、クエリの提案を消去できます。提案をクリアすると、既存のクエリ提案のみが削除されます。クエリ履歴内のクエリは削除されません。候補を消去すると、Amazon Kendra 候補をクリアした時点からクエリログに追加された新しいクエリに基づいて新しい候補が学習されます。

CLI

クエリの提案を消去するには

```
aws  kendra clear-query-suggestions \
 --index-id index-id
```

Python

クエリの提案を消去するには

```
import boto3
from botocore.exceptions import ClientError

kendra = boto3.client("kendra")

print("Clearing out query suggestions for an index.")

# Provide the index ID
index_id = "index-id"

try:
    kendra.clear_query_suggestions(
        IndexId = index_id
    )

    # Confirm last cleared date-time and that there are no suggestions
    query_sugg_config_response = kendra.describe_query_suggestions_config(
        IndexId = index_id
    )
```

```
    print("Query Suggestions last cleared at: " +
 str(query_sugg_config_response["LastClearTime"]));
    print("Number of suggestions available from the time of clearing: " +
 str(query_sugg_config_response["TotalSuggestionsCount"]));

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

提案がありません。

クエリの提案が表示されない場合は、次のいずれかの理由が考えられます。

- インデックスには学習に必要なクエリが足りません。 Amazon Kendra
- クエリの提案の設定が厳しすぎるため、ほとんどのクエリが提案から除外されています。
- Amazon Kendra 最近候補をクリアしたのに、新しいクエリが蓄積されて新しい提案を学習するにはまだ時間が必要です。

DescribeQuerySuggestionsConfig API を使用して、現在の設定を確認できます。

ドキュメントフィールドを使用したクエリの提案

トピック

- 提案用のフィールドを選択するための設定
- ドキュメントフィールドのユーザーコントロール

ドキュメントフィールドの内容に基づいて、ユーザーに関連するクエリの提案を選択できます。クエリ履歴を使って他の一般的な関連クエリを提案する代わりに、クエリのオートコンプリートに役立つドキュメントフィールド内の情報を使用できます。 Amazon Kendra Suggestableとに設定されたフィールドから、ユーザーのクエリとほぼ一致する関連コンテンツを探します。次に、ユーザーがクエリの入力を開始したときに、 Amazon Kendra このコンテンツを提案します。

たとえば、候補のベースとなるタイトルフィールドを指定し、ユーザーが「How amazon ken... 'では、最も関連性の高いタイトル「How Amazon Kendra works」が提案され、検索がオートコンプリートされます。検索バーのすぐ下に「How Amazon Kendra works」が表示され、検索クエリが

オートコンプリートされます。ユーザーがこのクエリ候補を選択すると、「How Amazon Kendra works」というドキュメントが検索結果に返されます。

クエリの提案のためのフィールド設定の一部として、フィールドを Suggestable に設定することにより、String および StringList のタイプのドキュメントフィールドの内容を使用してクエリを提案できます。ブロックリストを使用して、特定の語句が含まれるドキュメントフィールドの提案をユーザーに表示しないようにすることもできます。ブロックリストは 1 つしか使用できません。ブロックリストは、クエリの提案にクエリ履歴、またはドキュメント文書フィールドのどちらを使用するように設定したかにかかわらず適用されます。

提案用のフィールドを選択するための設定

次の設定を構成することにより、AttributeSuggestionsConfig を使用して提案用のドキュメントフィールドを選択し、UpdateQuerySuggestionsConfig API を呼び出してインデックスレベルで設定を更新できます。

- フィールド/属性提案モード - ドキュメントフィールドを使用するクエリの提案は、ACTIVE または INACTIVE のいずれかです。 Amazon Kendra では、デフォルトでクエリの提案を有効にします。
- 提案が可能なフィールド/属性 - 提案の基になるフィールド名またはフィールドキー。これらのフィールドは、フィールド設定の一部として Suggestable の場合は TRUE に設定する必要があります。インデックスレベルでは設定を維持したまま、クエリレベルではフィールドの設定をオーバーライドできます。GetQuerySuggestionsAPI AttributeSuggestionConfig を使用してクエリレベルで変更します。このクエリレベルの設定は、インデックスレベルで設定を更新しなくても、異なるドキュメントフィールドをすばやく試用するのに便利です。
- 追加フィールド/属性 - クエリの提案のレスポンスに含める追加フィールド。これらのフィールドは、レスポンスに追加情報を提供するために使用されますが、提案のベースには使用されません。

> **Warning**
>
> 設定の変更はすぐには反映されない場合があります。DescribeQuerySuggestionsConfig API を使用して、設定の変更を追跡できます。更新した設定が有効になるまでの時間は、行う更新によって異なります。 Amazon Kendra 設定を変更した後、または禁止リストを適用した後に、24 時間ごとに候補が自動的に更新されます。

CLI

インデックスレベルで設定を変更する代わりに、クエリレベルでクエリの提案を取得してドキュメントフィールドの設定をオーバーライドできます。

```
aws kendra get-query-suggestions \
 --index-id index-id \
 --query-text "query-text" \
 --suggestion-types '["DOCUMENT_ATTRIBUTES"]' \
 --attribute-suggestions-config '{"SuggestionAttributes":'["field/attribute key
 1", "field/attribute key 2"]', "AdditionalResponseAttributes":'["response field/
attribute key 1", "response field/attribute key 2"]'}' \
 --max-suggestions-count 1 // If you want to limit the number of suggestions
```

クエリの提案を更新するには

例えば、ドキュメントフィールドの設定をインデックスレベルで変更するには:

```
aws kendra update-query-suggestions-config \
 --index-id index-id \
 --attribute-suggestions-config '{"SuggestableConfigList": '[{"SuggestableConfig":
 "_document_title", "Suggestable": true}]', "AttributeSuggestionsMode": "ACTIVE"}'
```

Python

インデックスレベルで設定を変更する代わりに、クエリレベルでクエリの提案を取得してドキュメントフィールドの設定をオーバーライドできます。

```
import boto3
from botocore.exceptions import ClientError

kendra = boto3.client("kendra")

print("Get query suggestions.")

# Provide the index ID
index_id = "index-id"

# Provide the query text
query_text = "query"

# Provide the query suggestions type
```

```
query_suggestions_type = "DOCUMENT_ATTRIBUTES"

# Override fields/attributes configuration at query level
configuration = {"SuggestionAttributes":
    '["field/attribute key 1", "field/attribute key 2"]',
      "AdditionalResponseAttributes":
          '["response field/attribute key 1", "response field/attribute key 2"]'
          }

# If you want to limit the number of suggestions
num_suggestions = 1

try:
    query_suggestions_response = kendra.get_query_suggestions(
        IndexId = index_id,
        QueryText = query_text,
        SuggestionTypes = [query_suggestions_type],
        AttributeSuggestionsConfig = configuration,
        MaxSuggestionsCount = num_suggestions
    )

    # Print out the suggestions you received
    if ("Suggestions" in query_suggestions_response.keys()) {
        for (suggestion: query_suggestions_response["Suggestions"]) {
            print(suggestion["Value"]["Text"]["Text"]);
        }
    }

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

クエリの提案を更新するには

例えば、ドキュメントフィールドの設定をインデックスレベルで変更するには:

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")
```

```
print("Updating query suggestions settings/configuration for an index.")

# Provide the index ID
index_id = "index-id"

# Configure the settings you want to update at the index level
configuration = {"SuggestableConfigList":
    '[{"SuggestableConfig": "_document_title", "Suggestable": true}]',
       "AttributeSuggestionsMode": "ACTIVE"
       }

try:
    kendra.update_query_suggestions_config(
        IndexId = index_id,
        AttributeSuggestionsConfig = configuration
    )

    print("Wait for Amazon Kendra to update the query suggestions.")

    while True:
        # Get query suggestions description of settings/configuration
        query_sugg_config_response = kendra.describe_query_suggestions_config(
            IndexId = index_id
        )

        # If status is not UPDATING, then quit
        status = query_sugg_config_response["Status"]
        print(" Updating query suggestions config. Status: " + status)
        if status != "UPDATING":
            break
        time.sleep(60)

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

ドキュメントフィールドのユーザーコントロール

クエリの提案のベースにするドキュメントフィールドに、ユーザーコンテキストフィルタリングを適用できます。これにより、ユーザーまたはグループのドキュメントへのアクセスに基づいて、ドキュメントフィールド情報をフィルタリングします。例えば、インターンが会社のポータルを検索した

が、会社の極秘文書にはアクセスできなかったとします。そのため、極秘文書のタイトルや、その他の提案可能なフィールドに基づいて提案されたクエリは、インターンには表示されません。

アクセスコントロールリスト (ACL) を使用してドキュメントにインデックスを付け、ドキュメントへのアクセスを指定するユーザーやグループを定義でます。その後、クエリの提案のドキュメントフィールドにユーザーコンテキストフィルタリングを適用できます。現在、インデックスに設定されているユーザーコンテキストフィルタリングは、クエリの提案のドキュメントフィールド設定に適用されているユーザーコンテキストフィルタリングと同じです。ユーザーコンテキストフィルタリングは、ドキュメントフィールド設定の一部です。AttributeSuggestionsGetConfig を使用して、GetQuerySuggestions を呼び出します。

特定のクエリやドキュメントフィールドの内容を提案からブロックする

ブロックリストでは、 Amazon Kendra 特定のクエリがユーザーに提案されなくなります。ブロックリストは、クエリの候補から除外したい単語や語句のリストです。 Amazon Kendra ブロックリスト内の単語または語句と完全に一致するクエリを除外します。

ブロックリストを使用すると、クエリ履歴またはドキュメントフィールドに一般的に表示される、Amazon Kendra が提案として選択する可能性がある不快な単語や語句から保護できます。禁止リストを使用すると、 Amazon Kendra 公開または発表される準備が整っていない情報を含むクエリが提案されるのを防ぐこともできます。例えば、新製品がリリースされる可能性があり、ユーザーが頻繁に問い合わせているとします。ただし、まだリリースする準備ができていないために製品を提案しない場合は、製品名やその製品情報を含むクエリを提案からブロックできます。

CreateQuerySuggestionsBlockList API を使用して、クエリのブロックリストを作成できます。各単語または語句はテキストファイルで個別の行に配置します。次に、テキストファイルを Amazon S3 バケットにアップロードし、ファイルへのパスまたは場所を指定します Amazon S3。 Amazon Kendra 現在、ブロックリストは 1 つしか作成できません。

Amazon S3 バケット内のブロックされた単語やフレーズのテキストファイルを置き換えることができます。でブロックリストを更新するには Amazon Kendra、UpdateQuerySuggestionsBlockListAPI を使用します。

DescribeQuerySuggestionsBlockList API を使用してブロックリストのステータスを取得します。また、`DescribeQuerySuggestionsBlockList` は、次のようなその他の有用な情報を提供できます。

- ブロックリストの最終更新日時
- 現在のブロックリストにある語句の数

- ブロックリストを作成する際に役立つエラーメッセージ

また、ListQuerySuggestionsBlockLists API を使用して、インデックスに対するブロックリストの概要のリストを取得できます。

ブロックリストを削除するには DeleteQuerySuggestionsBlockListAPI を使用します。

ブロックリストへの更新はすぐには反映されない場合があります。DescribeQuerySuggestionsBlockList API を使用すると、更新を追跡できます。

CLI

ブロックリストを作成するには

```
aws kendra create-query-suggestions-block-list \
 --index-id index-id \
 --name "block-list-name" \
 --description "block-list-description" \
 --source-s3-path "Bucket=bucket-name,Key=query-suggestions/block_list.txt" \
 --role-arn role-arn
```

ブロックリストを更新するには

```
aws kendra update-query-suggestions-block-list \
 --index-id index-id \
 --name "new-block-list-name" \
 --description "new-block-list-description" \
 --source-s3-path "Bucket=bucket-name,Key=query-suggestions/new_block_list.txt" \
 --role-arn role-arn
```

ブロックリストを削除するには

```
aws kendra delete-query-suggestions-block-list \
 --index-id index-id \
 --id block-list-id
```

Python

ブロックリストを作成するには

```
import boto3
from botocore.exceptions import ClientError
```

```
import pprint
import time

kendra = boto3.client("kendra")

print("Create a query suggestions block list.")

# Provide a name for the block list
block_list_name = "block-list-name"
# Provide an optional description for the block list
block_list_description = "block-list-description"
# Provide the IAM role ARN required for query suggestions block lists
block_list_role_arn = "role-arn"

# Provide the index ID
index_id = "index-id"

s3_bucket_name = "bucket-name"
s3_key = "query-suggestions/block_list.txt"
source_s3_path = {
    'Bucket': s3_bucket_name,
    'Key': s3_key
}

try:
    block_list_response = kendra.create_query_suggestions_block_list(
        Description = block_list_description,
        Name = block_list_name,
        RoleArn = block_list_role_arn,
        IndexId = index_id,
        SourceS3Path = source_s3_path
    )

    print(block_list_response)

    block_list_id = block_list_response["Id"]

    print("Wait for Amazon Kendra to create the block list.")

    while True:
        # Get block list description
        block_list_description = kendra.describe_query_suggestions_block_list(
            Id = block_list_id,
            IndexId = index_id
```

```
        )
        # If status is not CREATING, then quit
        status = block_list_description["Status"]
        print("Creating block list. Status: " + status)
        if status != "CREATING":
            break
        time.sleep(60)

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

ブロックリストを更新するには

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Update a block list for query suggestions.")

# Provide the block list name you want to update
block_list_name = "new-block-list-name"
# Provide the block list description you want to update
block_list_description = "new-block-list-description"
# Provide the IAM role ARN required for query suggestions block lists
block_list_role_arn = "role-arn"

# Provide the block list ID
block_list_id = "block-list-id"
# Provide the index ID
index_id = "index-id"

s3_bucket_name = "bucket-name"
s3_key = "query-suggestions/new_block_list.txt"
source_s3_path = {
'Bucket': s3_bucket_name,
'Key': s3_key
}
```

```
try:
    kendra.update_query_suggestions_block_list(
        Id = block_list_id,
        IndexId = index_id,
        Description = block_list_description,
        Name = block_list_name,
        RoleArn = block_list_role_arn,
        SourceS3Path = source_s3_path
    )

    print("Wait for Amazon Kendra to update the block list.")

    while True:
        # Get block list description
        block_list_description = kendra.describe_query_suggestions_block_list(
            Id = block_list_id,
            IndexId = index_id
        )
        # If status is not UPDATING, then the update has finished
        status = block_list_description["Status"]
        print("Updating block list. Status: " + status)
        if status != "UPDATING":
            break
        time.sleep(60)

except ClientError as e:
print("%s" % e)

print("Program ends.")
```

ブロックリストを削除するには

```
import boto3
from botocore.exceptions import ClientError

kendra = boto3.client("kendra")

print("Delete a block list for query suggestions.")

# provide the block list ID
query_suggestions_block_list_id = "query-suggestions-block-list-id"
# Provide the index ID
index_id = "index-id"
```

```
try:
    kendra.delete_query_suggestions_block_list(
        Id = query_suggestions_block_list_id,
        IndexId = index_id
    )

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

クエリスペルチェッカー

Amazon Kendra スペルチェッカーにより、クエリのスペル修正を提案します。これにより、ゼロ検索結果になるのを最小限に抑え、関連する結果を返すことができます。クエリのスペルミスによるゼロ検索結果によって、一致する結果が返されなかったり、ドキュメントが返されなかったりする場合があります。また、ユーザーがスペルミスによって無関係の検索結果が返される場合があります。

スペルチェッカーは、インデックスが作成されたドキュメントに出現する単語と、修正された単語がスペルミスの単語とどの程度一致するかに基づき、スペルミスの単語の修正を提案するように設計されています。例えば、インデックスが作成されたドキュメントに「statements」という単語が含まれている場合、「year-end financial statments」というクエリの中のスペルミスの単語の「statments」とほぼ一致する可能性があります。

スペルチェッカーは、元のクエリテキスト内のスペルミスのある単語の代わりに、意図した単語または修正された単語を返します。例えば、「depoying kendre search」は「deploying Kendra search」が返される可能性があります。また、API で提供されているオフセット位置を使用して、フロントエンドアプリケーションのクエリで返された修正済みの単語を強調表示またはイタリックにできます。コンソールでは、修正済みの単語はデフォルトで強調表示またはイタリックで表示されます。例えば、「deploying Kendra search」などです。

インデックスに登録されたドキュメントにビジネス固有の用語や特殊な用語が含まれている場合、スペルチェッカーは、これらの語句をクエリのスペルミスとして誤って判断することはありません。例えば、「amazon macie」は「amazon mace」に修正されません。

「year-end」のようにハイフンでつながれた単語については、スペルチェッカーはそれらを個別の単語として扱い、修正を提案します。例えば、「yaer-end」の修正案は「year-end」の可能性があります。

DOCUMENT および QUESTION_ANSWER クエリのレスポンスタイプでは、スペルチェッカーはドキュメント本文の単語に基づいて、スペルミスのある単語の修正を提案します。スペルミスのある単語にほぼ一致する修正を提案する場合、タイトルよりもドキュメント本文の方が信頼できます。ANSWER クエリレスポンスタイプの場合、スペルチェッカーはインデックス内のデフォルトの質問と回答のドキュメントに含まれる単語に基づいて修正を提案します。

オブジェクトを使用してスペルチェッカーを起動できます。SpellCorrectionConfigurationIncludeQuerySpellCheckSuggestions を TRUE に設定します。スペルチェッカーは、コンソールではデフォルトで有効になっています。デフォルトでは、コンソールに組み込まれています。

また、スペルチェッカーは、英語だけでなく複数の言語でのクエリのスペル修正を提案できます。スペルチェッカーでサポートされている言語のリストについては、「Amazon Kendra supported languages」を参照してください。

クエリスペルチェッカーをデフォルトの制限付きで使用する

スペルチェッカーは、特定のデフォルト値または制限を設定して設計されています。次のリストは、スペル修正の提案を有効にした際に適用される現在の制限です。

- 3 文字未満または 30 文字を超える単語については、スペル修正の提案を返すことはできません。30 文字を超える、または 3 文字未満を有効にするには、サポートにお問い合わせください。
- スペル修正の提案では、ユーザーアクセスコントロールやユーザーコンテキストフィルタリング用のアクセスコントロールリストに基づいて候補を制限することはできません。スペル修正は、特定のユーザーに限定されているかどうかに関係なく、インデックスに登録されたドキュメント内のすべての単語に基づいて行われます。クエリのスペル修正の提案に特定の単語が表示されないようにする場合は、SpellCorrectionConfiguration を有効にしないでください。
- 数字を含む単語については、スペル修正の提案を返すことはできません。例えば、「how 2 not br8k ubun2」などです。
- スペル修正の提案には、インデックスが作成されたドキュメントにない単語は使用できません。
- スペル修正の提案には、インデックスが作成されたドキュメント内で頻度が 0.01% 未満の単語は使用できません。0.01% のしきい値を変更するには、サポートにお問い合わせください。

フィルタリングとファセット検索

フィルターを使用すると、クエリ API からの検索結果またはレスポンスを改善できます。フィルターにより、レスポンス内のドキュメントはクエリに直接適用するドキュメントに制限されます。

ファセット検索候補を作成するには、ブール論理を使用して、特定の条件に一致しないレスポンスまたはドキュメントから、特定のドキュメント属性をフィルタリングします。Query API Facets のパラメータを使用して、ファセットを指定できます。

インデックスに登録したドキュメントを検索するには Amazon Lex、 Amazon Kendra AMAZON を使用してください。 KendraSearchIntent。 Amazon Kendra での設定の例については Amazon Lex、「Amazon Kendra インデックス用の FAQ ボットの作成」を参照してください。を使用してレスポンスにフィルターを設定することもできますAttributeFilter。これは、AMAZON.KendraSearchIntent の設定時の JSON のクエリフィルターです。コンソールで検索インテントを設定する際に属性フィルターを指定するには、インテントエディタに移動し、[Amazon Kendra query] を選択して JSON でクエリフィルターを指定します。AMAZON.KendraSearchIntent の詳細については、「Amazon Lex ドキュメントガイド」を参照してください。

ファセット

ファセットは、一連の検索結果の対象範囲内のビューです。例えば、世界中の都市の検索結果を提供できます。この場合、ドキュメントは関連する特定の都市でフィルタリングされます。または、ファセットを作成して、特定の作成者の結果を表示できます。

ドキュメントに関連付けられているドキュメント属性またはメタデータフィールドをファセットとして使用すると、ユーザーはそのファセット内のカテゴリまたは値で検索できます。また、検索結果にネストされたファセットを表示すると、ユーザーはカテゴリやフィールドだけでなく、サブカテゴリやサブフィールドでも検索できます。

次の例は、「City」というカスタム属性のファセット情報を取得する方法を示しています。

```
response=kendra.query(
        QueryText = query,
        IndexId = index,
        Facets = [
            {
                "DocumentAttributeKey" : "City"
            }
        ]
        )
```

ネストされたファセットを使用すると、検索をさらに絞り込むことができます。例えば、ドキュメント属性またはファセット「City」には、「Seattle」という値が含まれています。さらに、ドキュメント属性またはファセット "CityRegion" には、「Seattle」に割り当てられたドキュメントの「North」

と「South」の値が含まれます。検索結果にはネストされたファセットとその数を表示できるため、都市だけでなく都市内の地域でもドキュメントを検索できます。

ネストされたファセットは、クエリのレイテンシーに影響する可能性があることに注意してください。一般的に、ネストされたファセットの使用が多いほど、レイテンシーへの影響が大きくなります。レイテンシーに影響を与えるその他の要因には、インデックスが作成されたドキュメントの平均サイズ、インデックスのサイズ、非常に複雑なクエリ、 Amazon Kendra インデックスの全体的な負荷などがあります。

次の例は、"CityRegion" カスタム属性のファセット情報を「City」内のネストされたファセットとして取得する方法を示しています。

```
response=kendra.query(
        QueryText = query,
        IndexId = index,
        Facets = [
            {
                "DocumentAttributeKey" : "City",
                "Facets": [
                    {
                        "DocumentAttributeKey" : "CityRegion"
                    }
                ]
            }
        ]
        )
```

ドキュメント数などのファセット情報は、FacetResults レスポンスの配列で返されます。コンテンツを使用して、アプリケーションでファセット検索提案を表示します。例えば、ドキュメント属性「City」に検索を適用できる都市が含まれている場合、その情報を使用して都市検索のリストを表示します。ユーザーは、都市を選択して検索結果をフィルタリングできます。ファセット検索を行うには、クエリ API を呼び出し、選択したドキュメント属性を使用して結果をフィルタリングします。

1 つのクエリでは、ファセットごとに最大 10 個のファセット値を表示でき、ファセット内には 1 つのネストされたファセットのみを表示できます。これらの制限を引き上げる場合は、サポートにお問い合わせください。ファセットあたりのファセット値の数を 10 未満に制限する場合は、これを Facet オブジェクト内で指定できます。

次の JSON レスポンスのサンプルは、対象範囲が「City」というドキュメント属性であるファセットを示しています。レスポンスには、ファセット値のドキュメント数が含まれます。

```
{
    'FacetResults': [
        {
            'DocumentAttributeKey': 'City',
            'DocumentAttributeValueCountPairs': [
                {
                    'Count': 3,
                    'DocumentAttributeValue': {
                        'StringValue': 'Dubai'
                    }
                },
                {
                    'Count': 3,
                    'DocumentAttributeValue': {
                        'StringValue': 'Seattle'
                    }
                },
                {
                    'Count': 1,
                    'DocumentAttributeValue': {
                        'StringValue': 'Paris'
                    }
                }
            ]
        }
    ]
```

また、都市内の地域などのネストされたファセットのファセット情報を表示して、検索結果をさらに絞り込むことができます。

次の JSON レスポンスのサンプルは、"CityRegion" ドキュメント属性にスコープされたファセットを「City」内のネストされたファセットとして示しています。レスポンスには、ネストされたファセット値のドキュメント数が含まれます。

```
{
    'FacetResults': [
        {
            'DocumentAttributeKey': 'City',
            'DocumentAttributeValueCountPairs': [
                {
                    'Count': 3,
                    'DocumentAttributeValue': {
```

```
                    'StringValue': 'Dubai'
                },
                'FacetResults': [
                    {
                        'DocumentAttributeKey': 'CityRegion',
                        'DocumentAttributeValueCountPairs': [
                            {
                                'Count': 2,
                                'DocumentAttributeValue': {
                                    'StringValue': 'Bur Dubai'
                                }
                            },
                            {
                                'Count': 1,
                                'DocumentAttributeValue': {
                                    'StringValue': 'Deira'
                                }
                            }
                        ]
                    }
                ]
            },
            {
                'Count': 3,
                'DocumentAttributeValue': {
                    'StringValue': 'Seattle'
                },
                'FacetResults': [
                    {
                        'DocumentAttributeKey': 'CityRegion',
                        'DocumentAttributeValueCountPairs': [
                            {
                                'Count': 1,
                                'DocumentAttributeValue': {
                                    'StringValue': 'North'
                                }
                            },
                            {
                                'Count': 2,
                                'DocumentAttributeValue': {
                                    'StringValue': 'South'
                                }
                            }
                        ]
```

```
                        }
                    ]
                },
                {
                    'Count': 1,
                    'DocumentAttributeValue': {
                        'StringValue': 'Paris'
                    },
                    'FacetResults': [
                        {
                            'DocumentAttributeKey': 'CityRegion',
                            'DocumentAttributeValueCountPairs': [
                                {
                                    'Count': 1,
                                    'DocumentAttributeValue': {
                                        'StringValue': 'City center'
                                    }
                                }
                            ]
                        }
                    ]
                }
            }
        ]
}
```

文字列リストフィールドを使用してファセットを作成する場合、返されるファセット結果は文字列リストの内容に基づきます。例えば、「dachshund」、「sausage dog」というリストと、「husky」という値を持つ 2 つの項目を含む文字列リストフィールドがある場合、3 つのファセットを持つ FacetResults を取得します。

詳細については、「クエリレスポンスとレスポンスタイプ」を参照してください。

ドキュメント属性を使用した検索結果のフィルタリング

デフォルトでは、Query はすべての検索結果を返します。レスポンスをフィルタリングするには、ドキュメント属性に対して論理演算を実行できます。例えば、特定の都市のドキュメントのみが必要な場合は、「City」および「State」のカスタムドキュメント属性でフィルタリングできます。AttributeFilterを使用して、指定したフィルターに対してブール演算を作成します。

ほとんどの属性は、すべての [response types] (レスポンスタイプ) のレスポンスをフィルタリングできます。ただし、レスポンスをフィルタリングする場合、_excerpt_page_number 属性は ANSWER レスポンスタイプにのみ適用されます。

次の例では、特定の州、ワシントン の特定の都市、シアトル でフィルタリングして論理 AND 演算を実行する方法を示します。

```
response=kendra.query(
        QueryText = query,
        IndexId = index,
        AttributeFilter = {'AndAllFilters':
            [
                {"EqualsTo": {"Key": "City","Value": {"StringValue": "Seattle"}}},
                {"EqualsTo": {"Key": "State","Value": {"StringValue": "Washington"}}}
            ]
            }
        )
```

次の例では、Fileformat、Author、または SourceURI キーが指定された値と一致する場合に論理 OR 演算を実行する方法を示します。

```
response=kendra.query(
        QueryText = query,
        IndexId = index,
        AttributeFilter = {'OrAllFilters':
            [
                {"EqualsTo": {"Key": "Fileformat","Value": {"StringValue":
 "AUTO_DETECT"}}},
                {"EqualsTo": {"Key": "Author","Value": {"StringValue": "Ana
 Carolina"}}},
                {"EqualsTo": {"Key": "SourceURI","Value": {"StringValue": "https://
aws.amazonaws.com/234234242342"}}}
            ]
            }
        )
```

StringList フィールドの場合、ContainsAny または ContainsAll 属性フィルターを使用して、指定した文字列を含むドキュメントを返します。次の例は、Locations カスタム属性に「Seattle」または「Portland」という値を持つすべてのドキュメントを返す方法を示しています。

```
response=kendra.query(
```

```
        QueryText = query,
        IndexId = index,
        AttributeFilter = {
                "ContainsAny": { "Key": "Locations", "Value": { "StringListValue":
 [ "Seattle", "Portland"] }}
            }
        )
```

検索結果内の各ドキュメント属性のフィルタリング

Amazon Kendra 検索結果の各ドキュメントのドキュメント属性を返します。検索結果の一部として、レスポンスに含める特定のドキュメント属性をフィルタリングできます。デフォルトでは、ドキュメントに割り当てられたすべてのドキュメント属性がレスポンスに返されます。

次の例では、_source_uri および _author ドキュメント属性は、ドキュメントのレスポンスに含まれます。

```
response=kendra.query(
        QueryText = query,
        IndexId = index,
        RequestedDocumentAttributes = ["_source_uri", "_author"]
        )
```

ユーザーコンテキストでのフィルタリング

ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、ユーザーの検索結果をフィルタリングできます。ユーザートークン、ユーザー ID、またはユーザー属性を使用して、ドキュメントをフィルタリングできます。 Amazon Kendra は、ユーザーをグループにマッピングすることもできます。ID ストア/ソースとして AWS IAM Identity Center を使用することを選択できます。

ユーザーコンテキストフィルタリングは、ドキュメントへのアクセスをコントロールできるという利点を持つ、パーソナライズされた検索の一種です。例えば、企業ポータルで情報を検索するすべてのチームが会社の極秘文書にアクセスする必要があるわけではなく、これらの文書がすべてのユーザーに関連しているわけでもありません。極秘文書へのアクセス許可を与えられた特定のユーザーまたはチームグループのみが、検索結果でこれらの文書を参照できます。

ドキュメントをインデックスに登録すると Amazon Kendra、ほとんどのドキュメントに対応するアクセス制御リスト (ACL) が取り込まれます。ACL は、ドキュメントへのアクセスを許可または拒否するユーザー名とグループ名を指定します。ACL のないドキュメントはパブリックドキュメントです。

Amazon Kendra ほとんどのデータソースについて、各ドキュメントに関連するユーザーまたはグループの情報を抽出できます。例えば、Quip のドキュメントには、そのドキュメントへのアクセス許可を持つ特定のユーザーの「共有」リストを含めることができます。S3 バケットをデータソースとして使用する場合は、ACL 用の JSON ファイルを提供し、このファイルへの S3 パスをデータソース設定の一部として含めます。ドキュメントをインデックスに直接追加する場合は、BatchPutDocumentAPI のドキュメントオブジェクトの一部として Principal オブジェクトの ACL を指定します。

CreateAccessControlConfigurationAPI を使用すると、すべてのドキュメントを再度インデックスしなくても、既存のドキュメントレベルのアクセス制御を再設定できます。例えば、インデックスには、特定の従業員またはユーザーだけがアクセスできる会社の極秘文書が含まれています。これらのユーザーのうちの 1 人が会社を退職したり、極秘文書へのアクセスをブロックすべきチームに異動したりします。文書が前にインデックスが作成されたときにアクセス許可を持っていたため、ユーザーは引き続き極秘文書にアクセスできます。アクセスを拒否するユーザーに対して、特定のアクセスコントロール設定を作成できます。ユーザーが会社に戻って「極秘」チームに再び加わった場合にアクセスを許可するように、後でアクセスコントロール設定を更新できます。状況の変化に応じて、ドキュメントのアクセスコントロールを再設定できます。

アクセス制御設定を特定のドキュメントに適用するには、BatchPutDocumentAccessControlConfigurationIdドキュメントオブジェクトに含まれているを使用して API を呼び出します。S3 バケットをデータソースとして使用する場合は、.metadata.jsonAccessControlConfigurationIdを使用してを更新し、データソースを同期します。 Amazon Kendra 現在、API を使用してインデックス化された S3 データソースとドキュメントのアクセスコントロール設定のみをサポートしています。BatchPutDocument

ユーザートークンによるフィルタリング

インデックスをクエリする際、ユーザートークンを使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングできます。クエリを発行すると、Amazon Kendra トークンを抽出して検証し、ユーザーとグループの情報を取得して確認し、クエリを実行します。パブリックドキュメントを含め、ユーザーがアクセスできるすべてのドキュメントが返されます。詳細については、「Token-based user access control」を参照してください。

UserContextユーザートークンをオブジェクトに指定し、これを Query API に渡します。

次に、ユーザートークンを含める方法を示します。

```
response = kendra.query(
    QueryText = query,
    IndexId = index,
```

```
    UserToken = {
        Token = "token"
    })
```

ユーザーをグループにマッピングできます。ユーザーコンテキストフィルタリングを使用する場合、クエリを発行するときに、ユーザーが属しているグループをすべて含める必要はありません。PutPrincipalMappingAPI を使用すると、ユーザーをグループにマップできます。PutPrincipalMapping API を使用しない場合は、クエリを発行するときに、ユーザー名とユーザーが属するすべてのグループを指定する必要があります。オブジェクトを使用して IAM Identity Center ID ソース内のグループとユーザーのアクセスレベルを取得することもできます。UserGroupResolutionConfiguration

ユーザー ID とグループによるフィルタリング

インデックスをクエリする際、ユーザー ID およびグループを使用して、ユーザーまたはそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングできます。クエリを発行すると、 Amazon Kendra ユーザーとグループの情報を確認し、クエリを実行します。パブリックドキュメントを含め、ユーザーがアクセスできるクエリに関連するすべてのドキュメントが返されます。

また、ユーザーおよびグループがアクセスできるデータソースで検索結果をフィルタリングできます。データソースの指定は、グループが複数のデータソースに関連付けられていても、特定のデータソースのドキュメントにのみアクセスできるようにする場合に便利です。例えば、「リサーチ」、「エンジニアリング」、および「セールスおよびマーケティング」のグループはすべて、Confluence および Salesforce のデータソースに保存されている企業のドキュメントに関連付けられています。ただし、「セールスおよびマーケティング」チームでは、Salesforce に保存されている顧客関連ドキュメントへのアクセスのみが必要です。そのため、営業やマーケティングに携わるユーザーが顧客関連のドキュメントを検索すると、その結果に Salesforce のドキュメントが表示されます。営業およびマーケティングで作業していないユーザーには、検索結果に Salesforce ドキュメントは表示されません。

ユーザー、グループ、UserContextデータソースの情報をオブジェクトに入力し、その情報を Query API に渡します。ユーザー ID、およびグループおよびデータソースのリストは、プリンシパルオブジェクトで指定した名前と一致する必要があり、ユーザー、グループ、およびデータソースを識別します。Principal オブジェクトを使用すると、ドキュメントにアクセスするための許可リストまたは拒否リストに、ユーザー、グループ、またはデータソースを追加できます。

次のいずれかを提供する必要があります。

- ユーザーとグループの情報、および (オプション) データソース情報。

- PutPrincipalMappingAPI を使用してユーザーをグループやデータソースにマッピングする場合は、ユーザー情報のみになります。オブジェクトを使用して IAM Identity Center ID ソース内のグループとユーザーのアクセスレベルを取得することもできます。UserGroupResolutionConfiguration

この情報がクエリに含まれていない場合は、 Amazon Kendra すべてのドキュメントが返されます。この情報を指定すると、ユーザー ID、グループ、およびデータソースが一致するドキュメントのみが返されます。

次に、ユーザー ID、グループ、およびデータソースを含める方法を示します。

```
response = kendra.query(
    QueryText = query,
    IndexId = index,
    UserId = {
        UserId = "user1"
    },
    Groups = {
        Groups = ["Sales and Marketing"]
    },
    DataSourceGroups = {
        DataSourceGroups = [{"DataSourceId" : "SalesforceCustomerDocsGroup", "GroupId":
 "Sales and Marketing"}]
    })
```

ユーザー属性でフィルタリングする

インデックスをクエリする際、組み込み属性の _user_id および _group_id を使用して、ユーザーおよびそのグループのドキュメントへのアクセスに基づいて、検索結果をフィルタリングできます。最大 100 個のグループ識別子を設定できます。クエリを発行すると、 Amazon Kendra ユーザーとグループの情報を確認し、クエリを実行します。パブリックドキュメントを含め、ユーザーがアクセスできるクエリに関連するすべてのドキュメントが返されます。

AttributeFilterオブジェクトにユーザー属性とグループ属性を指定し、これを Query API に渡します。

次の例は、ユーザー ID とユーザーが属するグループ「HR」および「IT」に基づいてクエリレスポンスをフィルタリングするリクエストを示しています。このクエリは、許可リストにユーザー、または「HR」または「IT」のグループを含むすべてのドキュメントを返します。ユーザーまたはいずれかのグループがドキュメントの拒否リストに含まれている場合、ドキュメントは返されません。

```
response = kendra.query(
        QueryText = query,
        IndexId = index,
        AttributeFilter = {
            "OrAllFilters": [
                {
                    "EqualsTo": {
                        "Key": "_user_id",
                        "Value": {
                            "StringValue": "user1"
                        }
                     }
                },
                {
                    "EqualsTo": {
                        "Key": "_group_ids",
                        "Value": {
                            "StringListValue": ["HR", "IT"]
                        }
                    }
                }
            ]
        }
        )
```

グループにアクセスできるデータソースは、Principal オブジェクトにアクセスできます。

> **Note**
>
> ユーザーコンテキストフィルタリングは、コンテンツの認証または認可コントロールではありません。Query API に送信されたユーザーおよびグループに対するユーザー認証は行いません。Query API に送信されたユーザーおよびグループの情報が認証および認可されるか否かは、アプリケーションによって異なります。

各データソースのユーザコンテキストフィルタリングの実装があります。以下のセクションでは、各実装について説明します。

トピック

- インデックスに直接追加されたドキュメントのユーザーコンテキストフィルタリング
- よくある質問に対するユーザーコンテキストのフィルタリング

- データソースのユーザーコンテキストフィルタリング

インデックスに直接追加されたドキュメントのユーザーコンテキストフィルタリング

BatchPutDocumentAPI を使用してドキュメントをインデックスに直接追加すると、 Amazon Kendra AccessControlListドキュメントのフィールドからユーザーとグループの情報が取得されます。ドキュメントのアクセスコントロールリスト (ACL) を指定し、その ACL をドキュメントに取り込みます。

ACL は、BatchPutDocument API のドキュメントオブジェクトの一部として、プリンシパルオブジェクトに指定します。次の情報を指定します。

- ユーザーまたはグループが持つ必要があるアクセス許可。ALLOW または DENY の返答が可能です。
- エンティティのタイプ。USER または GROUP の返答が可能です。
- ユーザーまたはグループの名前。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

よくある質問に対するユーザーコンテキストのフィルタリング

FAQ をインデックスに追加すると、FAQ JSON Amazon Kendra AccessControlList ファイルのオブジェクト/フィールドからユーザーとグループの情報を取得します。アクセスコントロール用のカスタムフィールドまたは属性を含む FAQ CSV ファイルを使用することもできます。

次の情報を指定します。

- ユーザーまたはグループが持つ必要があるアクセス許可。ALLOW または DENY の返答が可能です。
- エンティティのタイプ。USER または GROUP の返答が可能です。
- ユーザーまたはグループの名前。

詳細については、「FAQ files」を参照してください。

データソースのユーザーコンテキストフィルタリング

Amazon Kendra また、サポートされているデータソースコネクタからユーザーおよびグループのアクセス制御リスト (ACL) 情報をクロールします。これは、ドキュメントへのユーザーまたはグループのアクセス許可に基づいて検索結果をフィルタリングする場合の、ユーザーコンテキストフィルタリングの設定に役立ちます。

トピック

- Microsoft SharePoint データソースのユーザーコンテキストフィルタリング
- Microsoft SQL Server データソースのユーザーコンテキストフィルタリング
- Microsoft Teams データソースのユーザーコンテキストフィルタリング
- Microsoft Yammer データソースのユーザーコンテキストフィルタリング
- MySQL データソースのユーザーコンテキストフィルタリング
- Oracle Database データソースのユーザーコンテキストフィルタリング
- PostgreSQL データソースのユーザーコンテキストフィルタリング
- Quip データソースのユーザーコンテキストフィルタリング
- Salesforce データソースのユーザーコンテキストフィルタリング
- ServiceNow データソースのユーザーコンテキストフィルタリング
- Slack データソースのユーザーコンテキストフィルタリング
- Zendesk データソースのユーザーコンテキストフィルタリング

Adobe Experience Manager データソースのユーザーコンテキストフィルタリング

Adobe Experience Manager データソースを使用する場合、Adobe Experience Manager Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

グループ ID とユーザー ID は、次のようにマッピングされます。

- `_group_ids` - グループ ID は、アクセス許可が設定されている Adobe Experience Manager コンテンツに存在します。これらは、Adobe Experience Manager のグループの名前からマッピングされます。
- `_user_id` - ユーザー ID は、アクセス許可が設定されている Adobe Experience Manager コンテンツに存在します。これらは、Adobe Experience Manager のユーザーの E メールからマッピングされます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Alfresco データソースのユーザーコンテキストフィルタリング

Alfresco データソースを使用する場合、Alfresco Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

グループ ID とユーザー ID は、次のようにマッピングされます。

- `_group_ids` - グループ ID は、Alfresco のアクセス許可が設定されているファイルに存在します。Alfresco のグループ (表示名ではない) のシステム名からマッピングされます。
- `_user_id` - ユーザー ID は、Alfresco のアクセス許可が設定されているファイルに存在します。これらは Alfresco の ID としてユーザーの E メールからマッピングされます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Aurora (MySQL) データソースのユーザーコンテキストフィルタリング

Aurora (MySQL) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Aurora (MySQL) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Aurora (PostgreSQL) データソースのユーザーコンテキストフィルタリング

Aurora (PostgreSQL) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、API TemplateConfigurationの一部としてオブジェクトを使用します。CreateDataSource

Aurora (PostgreSQL) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon FSx データソースのユーザーコンテキストフィルタリング

Amazon FSx データソースを使用すると、 Amazon Kendra Amazon FSx インスタンスのディレクトリサービスからユーザーとグループの情報を取得します。

Amazon FSx グループ ID とユーザー ID は次のようにマッピングされます。

- `_group_ids` - グループ ID は、 Amazon FSx のアクセス許可が設定されているファイルに存在します。これらはのディレクトリサービスのシステムグループ名からマッピングされます。 Amazon FSx
- `_user_id`—ユーザー ID は、 Amazon FSx アクセス権限が設定されているファイルに存在します。これらは、のディレクトリサービスのシステムユーザ名からマップされます。 Amazon FSx

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

データベースデータソースのユーザーコンテキストフィルタリング

Amazon Aurora PostgreSQLなどのデータベースデータソースを使用すると、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報が取得されます。この列は CreateDataSourceAPI AclConfigurationDatabaseConfigurationのオブジェクトの一部としてオブジェクト内で指定します。

データベースデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon RDS (Microsoft SQL Server) データソースのユーザーコンテキストフィルタリング

Amazon RDS (Microsoft SQL Server) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Amazon RDS (Microsoft SQL Server) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon RDS (MySQL) データソースのユーザーコンテキストフィルタリング

Amazon RDS (MySQL) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Amazon RDS (MySQL) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon RDS (Oracle) データソースのユーザーコンテキストフィルタリング

Amazon RDS (Oracle) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Amazon RDS (Oracle) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon RDS (PostgreSQL) データソースのユーザーコンテキストフィルタリング

Amazon RDS (PostgreSQL) データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、API [TemplateConfiguration](#)の一部としてオブジェクトを使用します。[CreateDataSource](#)

Amazon RDS (PostgreSQL) データベースのデータソースには以下の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Amazon S3 データソースのユーザーコンテキストフィルタリング

ドキュメントに関連付けられたメタデータファイルを使用して、 Amazon S3 データソース内のドキュメントにユーザーコンテキストフィルターを追加します。情報を JSON ドキュメント内の `AccessControlList` フィールドに追加します。 Amazon S3 データソースからインデックスが作成されたドキュメントにメタデータを追加する方法の詳細については、「[S3 document metadata](#)」を参照してください。

次の 3 つの情報を提供します。

- エンティティが持つべきアクセス。`ALLOW` または `DENY` の返答が可能です。
- エンティティのタイプ。`USER` または `GROUP` の返答が可能です。
- エンティティの名前。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Amazon WorkDocs データソースのユーザーコンテキストフィルタリング

Amazon WorkDocs データソースを使用すると、 Amazon Kendra Amazon WorkDocs インスタンスからユーザーとグループの情報を取得します。

Amazon WorkDocs グループ ID とユーザー ID は次のようにマッピングされます。

- `_group_ids`—グループ ID は、 Amazon WorkDocs アクセス権限が設定されているファイルに存在します。内のグループの名前からマッピングされます。 Amazon WorkDocs

- _user_id—ユーザー ID は、 Amazon WorkDocs アクセス権限が設定されているファイルに存在します。のユーザー名からマッピングされます。 Amazon WorkDocs

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Box データソースのユーザーコンテキストフィルタリング

Box データソースを使用する場合、Box Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

Box グループ ID とユーザー ID は、次のようにマッピングされます。

- _group_ids - グループ ID は、Box のアクセス許可が設定されているファイルに存在します。これらは Box のグループの名前からマッピングされます。
- _user_id - ユーザー ID は、Box のアクセス許可が設定されているファイルに存在します。これらは Box のユーザー ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Confluence データソースのユーザーコンテキストフィルタリング

Confluence データソースを使用する場合、Confluence Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

[スペースアクセス許可] ページを使用して、スペースへのユーザーおよびグループアクセスを構成します。ページとブログの場合は、[制限] ページを使用します。スペースのアクセス許可の詳細については、Confluence Support ウェブサイトのスペースアクセス許可の概要を参照してください。ページおよびブログの制限の詳細については、Confluence Support ウェブサイトのページの制限を参照してください。

Confluence グループ名とユーザー名は、次のようにマッピングされます。

- _group_ids - グループ名は、制限のあるスペース、ページ、ブログに存在します。これらは Confluence のグループの名前からマッピングされます。グループ名は常に小文字です。

- _user_id - ユーザー名は、制限のあるスペース、ページ、ブログに存在します。これらは、使用している Confluence インスタンスのタイプに応じてマッピングされます。

  Confluence Connector v1.0 向け

- サーバー - `_user_id` はユーザー名です。ユーザーネームは常に小文字です。
- クラウド - `_user_id` はユーザーのアカウント ID です。

Confluence Connector v2.0 向け

- サーバー - `_user_id` はユーザー名です。ユーザーネームは常に小文字です。
- クラウド - `_user_id` はユーザーの E メール ID です。

> **Important**
>
> Confluence コネクタでユーザーコンテキストフィルタリングを正しく機能させるには、Confluence ページへのアクセスを許可されたユーザーの可視性が [全員] に設定されていることを確認する必要があります。詳細については、Atlassian デベロッパードキュメントの「Set your email visibility」を参照してください。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Dropbox データソースのユーザーコンテキストフィルタリング

Dropbox データソースを使用する場合、Dropbox Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

グループ ID とユーザー ID は、次のようにマッピングされます。

- `_group_ids` - グループ ID は、Dropbox のアクセス許可が設定されているファイルに存在します。これらは Dropbox のグループの名前からマッピングされます。
- `_user_id` - ユーザー ID は、Dropbox のアクセス許可が設定されているファイルに存在します。これらは Dropbox の ID としてユーザーの E メールからマッピングされます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Drupal データソースのユーザーコンテキストフィルタリング

Drupal データソースを使用する場合、Drupal Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

グループ ID とユーザー ID は、次のようにマッピングされます。

- _group_ids - グループ ID は、Drupal のアクセス許可が設定されているファイルに存在します。これらは Drupal のグループの名前からマッピングされます。
- _user_id - ユーザー ID は、Drupal のアクセス許可が設定されているファイルに存在します。これらは Drupal の ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

データソースのユーザーコンテキストフィルタリング GitHub

GitHub データソースを使用すると、 Amazon Kendra GitHub インスタンスからユーザー情報を取得します。

GitHub ユーザー ID は次のようにマッピングされます。

- _user_id—ユーザー ID は、 GitHub アクセス権限が設定されているファイルに存在します。ユーザー E メールから ID としてマッピングされます。 GitHub

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Gmail データソースのユーザーコンテキストフィルタリング

Gmail データソースを使用する場合、Gmail Amazon Kendra インスタンスからユーザー情報を取得します。

ユーザー ID は、次のようにマッピングされます。

- _user_id - ユーザー ID は、Gmail のアクセス許可が設定されているファイルに存在します。これらは Gmail の ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Google Drive データソースのユーザーコンテキストフィルタリング

Google Workspace Drive データソースは、Google Drive のユーザーおよびグループのユーザー情報とグループ情報を返します。グループおよびドメインメンバーシップは、_group_ids インデックスフィールドにマッピングされます。Google Drive のユーザー名は、_user_id フィールドにマッピングされます。

Query API で 1 つ以上のユーザーの E メールアドレスを指定する場合、それらの E メールアドレスと共有されているドキュメントのみが返されます。以下の AttributeFilter パラメータは、「martha@example.com」と共有されているドキュメントのみを返します。

```
"AttributeFilter": {
                "EqualsTo":{
                   "Key": "_user_id",
                   "Value": {
                       "StringValue": "martha@example.com"
                   }
               }
           }
```

クエリで 1 つ以上のグループの E メールアドレスを指定すると、グループと共有されているドキュメントのみが返されます。以下の AttributeFilter パラメータは、「hr@example.com」グループと共有されているドキュメントのみを返します。

```
"AttributeFilter": {
                "EqualsTo":{
                   "Key": "_group_ids",
                   "Value": {
                       "StringListValue": ["hr@example.com"]
                   }
               }
           }
```

クエリでドメインを指定すると、ドメインで共有されているすべてのドキュメントが返されます。以下の AttributeFilter パラメータは、「example.com」ドメインと共有されているドキュメントのみを返します。

```
"AttributeFilter": {
                "EqualsTo":{
                   "Key": "_group_ids",
                   "Value": {
                       "StringListValue": ["example.com"]
                   }
               }
           }
```

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

IBM DB2 データソースのユーザーコンテキストフィルタリング

IBM DB2 データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

IBM DB2 データベースデータソースには、次の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Jira データソースのユーザーコンテキストフィルタリング

Jira データソースを使用すると、Jira Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

Jira ユーザー ID は、次のようにマッピングされます。

- `_user_id` - ユーザー ID は、Jira のアクセス許可が設定されているファイルに存在します。これらは Jira のユーザー ID としてユーザーの E メールからマッピングされます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Microsoft Exchange データソースのユーザーコンテキストフィルタリング

Microsoft Exchange データソースを使用する場合、Microsoft Exchange Amazon Kendra インスタンスからユーザー情報を取得します。

Microsoft Exchange のユーザ ID は次のようにマッピングされています。

- `_user_id`—Microsoft Exchange の権限には、ユーザが特定のコンテンツにアクセスするためのユーザ ID があります。これらはユーザー名から Microsoft Exchange の ID としてマッピングされます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Microsoft OneDrive データソースのユーザーコンテキストフィルタリング

Amazon Kendra は、 OneDrive サイト上のドキュメントにインデックスを付けるときに、Microsoft からユーザーおよびグループの情報を取得します。ユーザーとグループの情報は、ホストしている基盤となる Microsoft SharePoint OneDrive サイトから取得されます。

OneDrive ユーザーまたはグループを使用して検索結果を絞り込む場合は、次のように ID を計算してください。

1. サイト名を取得します。例えば、`https://host.onmicrosoft.com/sites/siteName.`。
2. サイト名の MD5 ハッシュを取ります。例えば `430a6b90503eef95c89295c8999c7981` です。
3. MD5 ハッシュを縦棒 (|) と ID で連結して、ユーザーの E メールまたはグループ ID を作成します。たとえば、グループ名が "localGroupName" の場合、グループ ID は次のようになります。

   `"430a6b90503eef95c89295c8999c7981 | localGroupName"`

   > **Note**
   >
   > 縦棒の前後にスペースを入れてください。垂直バーは MD5 `localGroupName` ハッシュで識別するために使用されます。

   ユーザー名が「someone@host.onmicrosoft.com」の場合、ユーザー ID は次のようになります。

   `"430a6b90503eef95c89295c8999c7981 | someone@host.onmicrosoft.com"`

Query API を呼び出すときに、ユーザー ID `_user_id` またはグループ ID Amazon Kendra `_group_id` をまたは属性としてに送信します。たとえば、 AWS CLI グループを使用して検索結果をフィルタリングするコマンドは以下のようになります。

```
aws kendra  query \
                --index-id index ID
                --query-text "query text"
                --attribute-filter '{
                   "EqualsTo":{
                     "Key": "_group_id",
                     "Value": {"StringValue": "430a6b90503eef95c89295c8999c7981 |
 localGroupName"}
                  }}'
```

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Microsoft OneDrive v2.0 データソースのユーザーコンテキストフィルタリング

Microsoft OneDrive v2.0 データソースは、 OneDrive アクセス制御リスト (ACL) エンティティからセクションとページの情報を返します。 Amazon Kendra OneDrive OneDrive テナントドメインを使用してインスタンスに接続し、セクションやファイル名へのユーザーまたはグループのアクセスに基づいて検索結果をフィルタリングできます。

標準オブジェクトについては、_user_id および _group_id が次のように使用されます。

- _user_id— Microsoft OneDrive ユーザーの電子メール ID _user_id がフィールドにマップされます。
- _group_id— Microsoft OneDrive _group_id グループのメールがフィールドにマップされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Microsoft SharePoint データソースのユーザーコンテキストフィルタリング

Amazon Kendra SharePoint サイトドキュメントのインデックスを作成するときに、Microsoft からユーザーおよびグループの情報を取得します。ユーザーまたはグループのアクセスに基づいて検索結果をフィルタリングするには、API を呼び出すときにユーザーとグループの情報を入力します。Query

ユーザー名を使用してフィルタリングするには、ユーザーの E メールアドレスを使用します。例えば、johnstiles@example.com。

SharePoint グループを使用して検索結果を絞り込む場合は、次のようにグループ ID を計算します。

ローカルグループ用

1. サイト名を取得します。例えば、https://host.onmicrosoft.com/sites/siteName.。
2. サイト名の SHA256 ハッシュを取ります。例えば 430a6b90503eef95c89295c8999c7981 です。
3. SHA256 ハッシュを縦棒 (|) と ID で連結して、グループ ID を作成します。たとえば、グループ名が "localGroupName" の場合、グループ ID は次のようになります。

   "430a6b90503eef95c89295c8999c7981 | localGroupName"

> **Note**
>
> 縦棒の前後にスペースを入れてください。縦棒は SHA256 `localGroupName` ハッシュで識別するために使用されます。

Query API を呼び出すときに、グループ ID Amazon Kendra `_group_id` を属性としてに送信します。たとえば、 AWS CLI コマンドは以下のようになります。

```
aws kendra  query \
                --index-id index ID
                --query-text "query text"
                --attribute-filter '{
                   "EqualsTo":{
                     "Key": "_group_id",
                     "Value": {"StringValue": "430a6b90503eef95c89295c8999c7981 |
 localGroupName"}
                  }}'
```

AD グループ用

1. AD グループ ID を使用して検索結果のフィルタリングを設定します。

Query API を呼び出すときに、グループ ID Amazon Kendra `_group_id` を属性としてに送信します。たとえば、 AWS CLI コマンドは以下のようになります。

```
aws kendra  query \
                --index-id index ID
                --query-text "query text"
                --attribute-filter '{
                   "EqualsTo":{
                     "Key": "_group_id",
                     "Value": {"StringValue": "AD group"}
                  }}'
```

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Microsoft SQL Server データソースのユーザーコンテキストフィルタリング

Microsoft SQL Server データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Microsoft SQL Server データベースデータソースには、次の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Microsoft Teams データソースのユーザーコンテキストフィルタリング

Amazon Kendra ドキュメントにインデックスを付けるときに Microsoft Teams からユーザー情報を取得します。ユーザー情報は、基盤となるMicrosoft Teamsインスタンスから取得されます。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

Microsoft Yammer データソースのユーザーコンテキストフィルタリング

Amazon Kendra ドキュメントにインデックスを付けるときに Microsoft Yammer からユーザー情報を取得します。ユーザーとグループの情報は、基盤となる Microsoft Yammer インスタンスから取得されます。

Microsoft Yammer ユーザー ID は、次のようにマッピングされます。

- `_email_id`— `_user_id` フィールドにマップされている Microsoft 電子メール ID。

`AccessControlList` フィールドでは最大 200 個のエントリを追加できます。

MySQL データソースのユーザーコンテキストフィルタリング

MySQL データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

MySQL データベースデータソースには、次の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Oracle Database データソースのユーザーコンテキストフィルタリング

Oracle Database データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、CreateDataSourceAPI TemplateConfigurationの一部としてオブジェクトを使用します。

Oracle Database データベースデータソースには、次の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

PostgreSQL データソースのユーザーコンテキストフィルタリング

PostgreSQL データソースを使用する場合、 Amazon Kendra ソーステーブルの列からユーザーとグループの情報を取得します。この列はコンソールで指定するか、API TemplateConfigurationの一部としてオブジェクトを使用します。CreateDataSource

PostgreSQL データベースデータソースには、次の制限があります。

- データベースデータソースの許可リストのみを指定できます。拒否リストを指定することはできません。
- グループのみを指定できます。許可リストに個別のユーザーを指定することはできません。
- データベース列は、セミコロンで区切られたグループのリストを含む文字列である必要があります。

Quip データソースのユーザーコンテキストフィルタリング

Quip データソースを使用する場合、Quip Amazon Kendra インスタンスからユーザー情報を取得します。

Quip ユーザー ID は、次のようにマッピングされます。

- _user_id - ユーザー ID は、Quip のアクセス許可が設定されているファイルに存在します。これらは Quip の ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Salesforce データソースのユーザーコンテキストフィルタリング

Salesforce データソースは、Salesforce アクセスコントロールリスト (ACL) エンティティからユーザーおよびグループ情報を返します。Salesforce 標準オブジェクトおよび Chatter フィードにユーザーコンテキストフィルタリングを適用できます。ユーザーコンテキストフィルタリングは、Salesforce ナレッジ記事では使用できません。

標準オブジェクトについては、_user_id および _group_ids が次のように使用されます。

- _user_id - Salesforce ユーザーのユーザー名。
- _group_ids—
  - Salesforce Profile の名前
  - Salesforce Group の名前
  - Salesforce UserRole の名前
  - Salesforce PermissionSet の名前

Chatter フィードの場合、_user_id および _group_ids は次のように使用されます。

- _user_id - Salesforce ユーザーのユーザー名。項目がユーザーのフィードに投稿されている場合にのみ使用できます。
- _group_ids - グループ ID は、次のように使用されます。フィードフィードが Chatter またはコラボレーショングループに投稿されている場合にのみ使用できます。
  - Chatter グループまたはコラボレーショングループの名前。
  - グループが公開されている場合、PUBLIC:ALL。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

ServiceNow データソースのユーザーコンテキストフィルタリング

ServiceNow のユーザーコンテキストフィルタリングは TemplateConfiguration API と ServiceNow Connector v2.0 でのみサポートされています。 ServiceNowConfigurationAPI と ServiceNow Connector v1.0 はユーザーコンテキストフィルタリングをサポートしていません。

ServiceNow データソースを使用する場合、 Amazon Kendra インスタンスからユーザーとグループの情報を取得します。 ServiceNow

グループ ID とユーザー ID は、次のようにマッピングされます。

- _group_ids—グループ ID は、 ServiceNow アクセス権限が設定されているファイルに存在します。in sys_ids のロール名からマッピングされます。 ServiceNow
- _user_id—ユーザー ID は、 ServiceNow アクセス権限が設定されているファイルにあります。ユーザー E メールから ID としてマッピングされます。 ServiceNow

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Slack データソースのユーザーコンテキストフィルタリング

Slack データソースを使用する場合、Slack Amazon Kendra インスタンスからユーザー情報を取得します。

Slack ユーザー ID は、次のようにマッピングされます。

- _user_id - ユーザー ID は、アクセス許可が設定されている Slack のメッセージとチャネルに存在します。これらは Slack の ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

Zendesk データソースのユーザーコンテキストフィルタリング

Zendesk データソースを使用する場合、Zendesk Amazon Kendra インスタンスからユーザーとグループの情報を取得します。

グループ ID とユーザー ID は、次のようにマッピングされます。

- _group_ids - グループ ID は、アクセス許可が設定されている Zendesk のチケットおよび記事に存在します。これらは Zendesk のグループの名前からマッピングされます。

- _user_id - グループ ID は、アクセス許可が設定されている Zendesk のチケットおよび記事に存在します。これらは Zendesk の ID としてユーザーの E メールからマッピングされます。

AccessControlList フィールドでは最大 200 個のエントリを追加できます。

クエリレスポンスとレスポンスタイプ

Amazon Kendra さまざまなクエリレスポンスとレスポンスタイプをサポートします。

クエリレスポンス

クエリ API を呼び出すと、検索結果に関する情報を返します。QueryResultItem結果はオブジェクトの配列 (ResultItems) に格納されます。各 QueryResultItem には、結果の概要が含まれます。クエリの結果に関連付けられたドキュメント属性が含まれます。

概要情報

概要情報は、結果のタイプによって異なります。いずれの場合も、検索条件に一致するドキュメントテキストが含まれます。また、アプリケーションの出力で検索テキストを強調表示表示するために使用できる強調表示情報も含まれます。例えば、検索条件が what is the height of the Space Needle? の場合、概要情報には、height および space needle の単語のテキストロケーションが含まれます。レスポンスタイプに関する情報については、クエリレスポンスとレスポンスタイプ を参照してください。

ドキュメント属性

各結果には、クエリに一致するドキュメントのドキュメント属性が含まれます。一部の属性は、DocumentId、DocumentTitle、および DocumentUri のような事前定義がされています。その他は、独自に定義するカスタム属性です。ドキュメント属性を使用して、Query API からのレスポンスをフィルタリングできます。例えば、特定の作成者に書かれたドキュメントまたは特定のバージョンのドキュメントのみが必要な場合があります。詳細については、「フィルタリングとファセット検索」を参照してください。ドキュメント属性は、ドキュメントをインデックスに追加するときに指定します。詳細については、「Custom fields or attributes」を参照してください。

次に、クエリ結果の JSON コードのサンプルを示します。DocumentAttributes および AdditionalAttributes のドキュメント属性に注意してください。

```
{
    "QueryId": "query-id",
    "ResultItems": [
```

```
        {
            "Id": "result-id",
            "Type": "ANSWER",
            "AdditionalAttributes": [
                {
                    "Key": "AnswerText",
                    "ValueType": "TEXT_WITH_HIGHLIGHTS_VALUE",
                    "Value": {
                        "TextWithHighlightsValue": {
                            "Text": "text",
                            "Highlights": [
                                {
                                    "BeginOffset": 55,
                                    "EndOffset": 90,
                                    "TopAnswer": false
                                }
                            ]
                        }
                    }
                }
            ],
            "DocumentId": "document-id",
            "DocumentTitle": {
                "Text": "title"
            },
            "DocumentExcerpt": {
                "Text": "text",
                "Highlights": [
                    {
                        "BeginOffset": 0,
                        "EndOffset": 300,
                        "TopAnswer": false
                    }
                ]
            },
            "DocumentURI": "uri",
            "DocumentAttributes": [],
            "ScoreAttributes": "score",
            "FeedbackToken": "token"
        },
        {
            "Id": "result-id",
            "Type": "ANSWER",
            "Format": "TABLE",
```

```
            "DocumentId": "document-id",
            "DocumentTitle": {
                "Text": "title"
            },
            "TableExcerpt": {
                "Rows": [{
                    "Cells": [{
                        "Header": true,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": true,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": true,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": true,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": true,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }]
                }, {
                    "Cells": [{
                        "Header": false,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": false,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    }, {
                        "Header": false,
                        "Highlighted": true,
                        "TopAnswer": true,
                        "Value": "value"
                    }, {
```

```
                        "Header": false,
                        "Highlighted": false,
                        "TopAnswer": false,
                        "Value": "value"
                    ]}
                }],
                    "TotalNumberofRows": number
    },
            "DocumentURI": "uri",
            "ScoreAttributes": "score",
            "FeedbackToken": "token"
        },
        {
            "Id": "result-id",
            "Type": "DOCUMENT",
            "AdditionalAttributes": [],
            "DocumentId": "document-id",
            "DocumentTitle": {
                "Text": "title",
                "Highlights": []
            },
            "DocumentExcerpt": {
                "Text": "text",
                "Highlights": [
                    {
                        "BeginOffset": 74,
                        "EndOffset": 77,
                        "TopAnswer": false
                    }
                ]
            },
            "DocumentURI": "uri",
            "DocumentAttributes": [
                {
                    "Key": "_source_uri",
                    "Value": {
                        "StringValue": "uri"
                    }
                }
            ],
            "ScoreAttributes": "score",
            "FeedbackToken": "token",
        }
     ],
```

```
    "FacetResults": [],
    "TotalNumberOfResults": number
}
```

レスポンスのタイプ

Amazon Kendra 3 種類のクエリレスポンスを返します。

- 回答 (表の回答を含む)
- ドキュメント
- 質問と回答

TypeQueryResultItemレスポンスのタイプはオブジェクトのレスポンスフィールドに返されます。

回答

Amazon Kendra 応答に 1 つ以上の質問の回答が検出されました。Factoid とは、誰が、何を、いつ、どこで、の質問に対するレスポンスです。例えば「Where is the nearest service center to me?」では、 Amazon Kendra はクエリに最も一致するインデックス内のテキストを返します。テキストは AnswerText フィールドにあり、レスポンステキスト内の検索条件の強調表示情報を含みます。AnswerText は、強調表示されたテキストを含むドキュメント全体の抜粋を含み、DocumentExcerpt は、切り捨てられた (290 文字) のドキュメントの抜粋と強調表示されたテキストを含みます。

Amazon Kendra 1 つのドキュメントにつき 1 つの回答のみを返し、それが最も信頼度の高い回答です。ドキュメントから複数の回答を返すには、ドキュメントを複数のドキュメントに分割する必要があります。

```
{
    'AnswerText': {
        'TextWithHighlights': [
            {
                'BeginOffset': 271,
                'EndOffset': 279,
                'TopAnswer': False
            },
            {
                'BeginOffset': 481,
                'EndOffset': 489,
                'TopAnswer': False
```

```
            },
            {
                'BeginOffset': 547,
                'EndOffset': 555,
                'TopAnswer': False
            },
            {
                'BeginOffset': 764,
                'EndOffset': 772,
                'TopAnswer': False
            }
        ],
        'Text': 'Asynchronousoperationscan\n''alsoprocess
\n''documentsthatareinPDF''format.UsingPDFformatfilesallowsyoutoprocess''multi-
page\n''documents.\n''Forinformationabouthow''AmazonTextractrepresents
\n''documentsasBlockobjects,
        ''seeDocumentsandBlockObjects.
\n''\n''\n''\n''Forinformationaboutdocument''limits,
        seeLimitsinAmazonTextract.
\n''\n''\n''\n''TheAmazonTextractsynchronous''operationscanprocessdocumentsstoredinanAmazon
\n''S3Bucketoryoucanpass''base64encodedimagebytes.\n''Formoreinformation,

 see''CallingAmazonTextractSynchronousOperations.''Asynchronousoperationsrequireinputdocuments
\n''tobesuppliedinanAmazon''S3Bucket.'
    },
    'DocumentExcerpt': {
        'Highlights': [
            {
                'BeginOffset': 0,
                'EndOffset': 300,
                'TopAnswer': False
            }
        ],
        'Text': 'Asynchronousoperationscan\n''alsoprocess
\n''documentsthatareinPDF''format.UsingPDFformatfilesallowsyoutoprocess''multi-page
\n''documents.\n''ForinformationabouthowAmazon''Textractrepresents\n'''
    },
    'Type': 'ANSWER'
}
```

ドキュメント

Amazon Kendra 検索語に一致するドキュメントをランク付けして返します。ランキングは、検索結果の正確性に対する信頼度に基づいて決定されます。 Amazon Kendra 一致するドキュメントに関する情報がに返されますQueryResultItem。これにはドキュメントのタイトルが含まれます。この抜粋には、検索テキストの強調表示情報と、ドキュメント内の一致するテキストのセクションが含まれています。一致するドキュメントの URI は、SourceURI ドキュメント属性にあります。次のサンプル JSON は、一致するドキュメントのドキュメント概要を示しています。

```
{
    'DocumentTitle': {
        'Highlights': [
            {
                'BeginOffset': 7,
                'EndOffset': 15,
                'TopAnswer': False
            },
            {
                'BeginOffset': 97,
                'EndOffset': 105,
                'TopAnswer': False
            }
        ],
        'Text': 'AmazonTextractAPIPermissions: Actions,
        \n''Permissions,
        andResourcesReference-''AmazonTextract'
    },
    'DocumentExcerpt': {
        'Highlights': [
            {
                'BeginOffset': 68,
                'EndOffset': 76,
                'TopAnswer': False
            },
            {
                'BeginOffset': 121,
                'EndOffset': 129,
                'TopAnswer': False
            }
        ],
        'Text': '...LoggingandMonitoring\tMonitoring
\n''\tCloudWatchMetricsforAmazonTextract
```

```
\n''\tLoggingAmazonTextractAPICallswithAWSCloudTrail\n''\tAPIReference\tActions
\tAnalyzeDocument\n''\tDetectDocumentText\n''\tGetDocumentAnalysis...'
    },
    'Type': 'DOCUMENT'
}
```

質問と回答

Amazon Kendra 質問がインデックス内のよくある質問の1つと一致すると、質問と回答が返されます。回答には、QueryResultItemフィールド内の一致する質問と回答が含まれます。また、クエリ文字列で検出されたクエリ条件の強調表示情報も含まれます。次の JSON は、質問と回答のレスポンスを示しています。レスポンスには質問のテキストが含まれていることに注意してください。

```
{
    'AnswerText': {
        'TextWithHighlights': [

        ],
        'Text': '605feet'
    },
    'DocumentExcerpt': {
        'Highlights': [
            {
                'BeginOffset': 0,
                'EndOffset': 8,
                'TopAnswer': False
            }
        ],
        'Text': '605feet'
    },
    'Type': 'QUESTION_ANSWER',
    'QuestionText': {
        'Highlights': [
            {
                'BeginOffset': 12,
                'EndOffset': 18,
                'TopAnswer': False
            },
            {
                'BeginOffset': 26,
                'EndOffset': 31,
                'TopAnswer': False
            },
```

```
            {
                'BeginOffset': 32,
                'EndOffset': 38,
                'TopAnswer': False
            }
        ],
        'Text': 'whatistheheightoftheSpaceNeedle?'
    }
}
```

質問と回答のテキストをインデックスに追加する方法の詳細については、「Creating FAQ」を参照してください。

レスポンスのチューニングとソート

検索の関連性に対するフィールドまたは属性の影響は、[relevance tuning] (関連性チューニング) で変更できます。検索結果を特定の属性やフィールドでソートすることもできます。

トピック

- レスポンスのチューニング
- レスポンスのソート

レスポンスのチューニング

検索の関連性に対するフィールドまたは属性の影響は、[relevance tuning] (関連性チューニング) で変更できます。関連性のチューニングをすばやくテストするには、クエリ API を使用してクエリ内のチューニング設定を渡します。次に、さまざまな構成から取得したさまざまな検索結果が表示されます。クエリレベルでの関連性のチューニングは、コンソールではサポートされていません。また、インデックスレベルに限り、`StringList` タイプのフィールドや属性をチューニングできます。詳細については、「Tuning search relevance」を参照してください。

デフォルトでは、クエリレスポンスは、 Amazon Kendra レスポンスに含まれる結果ごとに決まる関連性スコアでソートされます。

次のタイプの組み込み属性またはカスタム属性/フィールドの結果をチューニングできます。

- 日付値
- 長い値

- 文字列値

次のタイプの属性はソートできません。

- 文字列リスト値

ドキュメント結果のランク付けと調整 (AWS SDK)

設定: `Searchable` パラメータを true に設定すると、ドキュメントのメタデータ構成が強化されます。

クエリ内の属性をチューニングするには、`Query` API の `DocumentRelevanceOverrideConfigurations` パラメータを設定し、チューニングする属性の名前を指定します。

次の JSON 例は、インデックス内の「department」という属性のチューニングを上書きする `DocumentRelevanceOverrideConfigurations` オブジェクトを示しています。

```
"DocumentRelevanceOverrideConfigurations" : [
    "Name": "department",
    "Relevance": {
        "Importance": 1,
        "ValueImportanceMap": {
            "IT": 3,
            "HR": 7
        }
    }
]
```

レスポンスのソート

Amazon Kendra クエリによって返されるドキュメントの条件の一部として、ソート属性またはフィールドを使用します。例えば、「_created_at」でソートされたクエリによって返される結果には、「_version」でソートされたクエリと同じ結果が含まれない場合があります。

デフォルトでは、クエリのレスポンスは、 Amazon Kendra レスポンスに含まれる結果ごとに決まる関連性スコアでソートされます。並べ替え順序を変更するには、文書属性を並べ替え可能にし、Amazon Kendra その属性を使用して応答を並べ替えるように設定します。

次のタイプの組み込み属性/フィールドまたはカスタム属性/フィールドの結果をソートできます。

- 日付値
- 長い値
- 文字列値

次のタイプの属性はソートできません。

- 文字列リスト値

各クエリは、1 つまたは複数のドキュメント属性でソートできます。クエリは 100 の結果を返します。ソート属性が設定されたドキュメントが 100 未満の場合、ソート属性の値を持たないドキュメントが結果の最後に返され、クエリとの関連性でソートされます。

ドキュメントの結果をソートするには (AWS SDK)

1. UpdateIndexAPI を使用して属性をソート可能にするには、`Sortable`パラメータをに設定します。`true`次の JSON の例は、`DocumentMetadataConfigurationUpdates` を使用して「Department」という属性をインデックスに追加して、ソートできるようにしています。

```
"DocumentMetadataConfigurationUpdates": [
   {
       "Name": "Department",
       "Type": "STRING_VALUE",
       "Search": {
           "Sortable": "true"
       }
   }
]
```

2. クエリで 1 つのソート可能な属性を使用するには、クエリ API の `SortingConfiguration` パラメータを設定します。ソートする属性の名前と、レスポンスを昇順または降順のどちらでソートするかを指定します。

   次の JSON の例は、「Department」属性で昇順でクエリの結果をソートするために使用する `SortingConfiguration` パラメータを示しています。

```
   "SortingConfiguration": {
      "DocumentAttributeKey": "Department",
      "SortOrder": "ASC"
   }
```

3. クエリで複数のソート可能な属性を使用するには、クエリ API の SortingConfigurations パラメータを設定します。 Amazon Kendra によって結果をソートするフィールドを 3 つまで設定できます。また、結果を昇順と降順のどちらでソートするかを指定できます。ソートフィールドのクォータは、増やすことができます。

   ソート設定を指定しない場合、 Amazon Kendra 結果は結果を決定する関連性に基づいてソートされます。結果のソートでタイの場合、結果は関連性でソートされます。

   次の JSON の例では、クエリの結果を「Name」と「Price」で昇順にソートするために使用する SortingConfigurations パラメータを示しています。

```
"CollapseConfiguration" : {
    "DocumentAttributeKey": "Name",
    "SortingConfigurations": [
        {
            "DocumentAttributeKey": "Price",
            "SortOrder": "ASC"
        }
    ],
    "MissingAttributeKeyStrategy": "IGNORE"
}
```

ドキュメントの結果をソートするには (コンソール)

> **Note**
>
> 現在、 AWS Management Consoleによる複数属性のソートはサポートしていません。

1. コンソールで属性をソート可能にするには、属性定義で [Sortable] (ソート可能) を選択します。属性の作成時に属性をソート可能にすることも、後で変更することもできます。
2. コンソールでクエリのレスポンスをソートするには、属性を選択して、[Sort] (ソート) メニューからレスポンスをソートします。データソース設定中にソート可能とマークされた属性のみがリストに表示されます。

クエリ結果の折りたたみ/展開

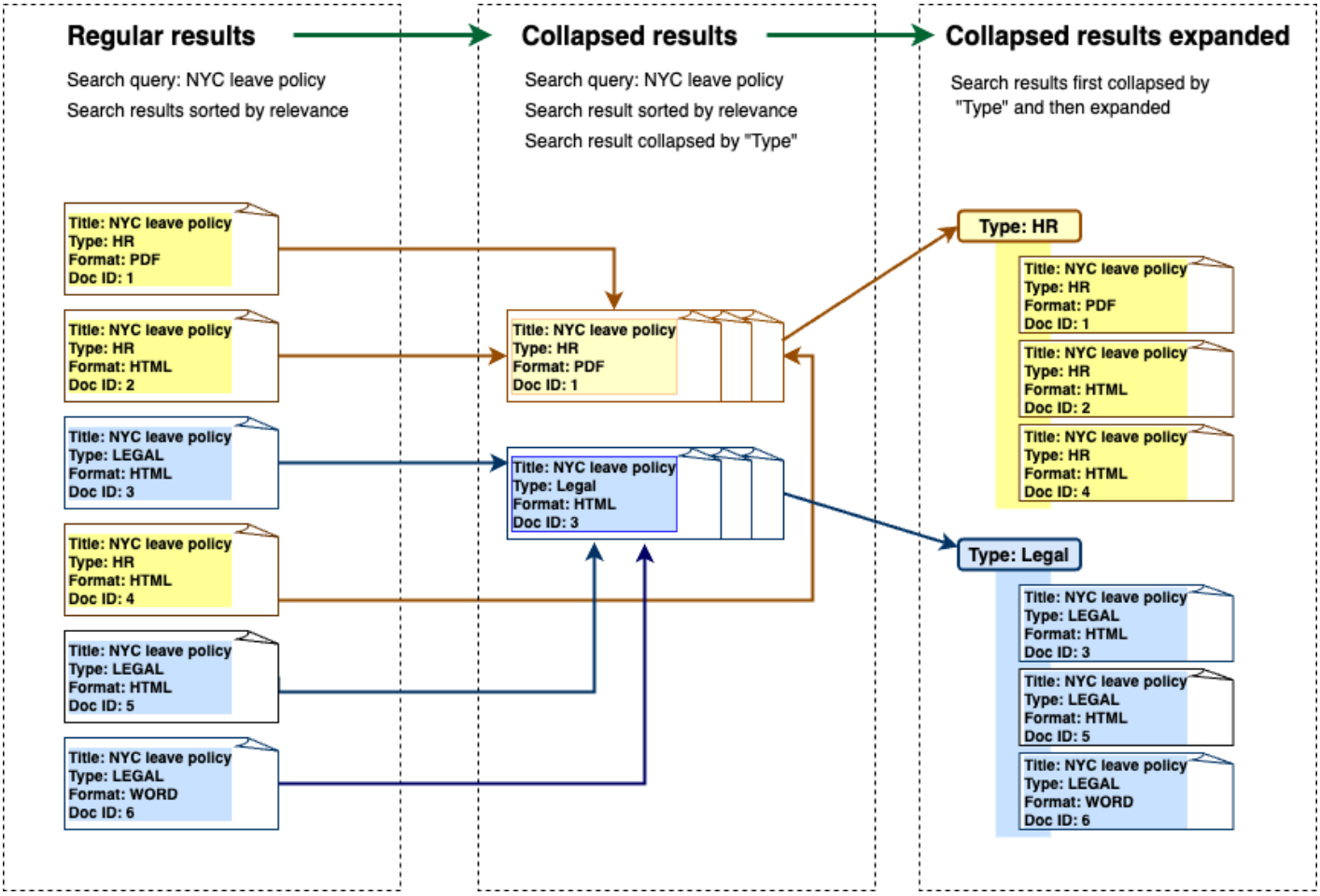
Amazon Kendra データに接続すると、、、[などのドキュメントメタデータ属性がクロールされ_document_title_created_at_document_id、これらの属性またはフィールドを使用してクエリ時に高度な検索機能が提供されます](#)。

Amazon Kendraのクエリ結果の折りたたみと展開機能を使用すると、共通のドキュメント属性を使用して検索結果をグループ化して、指定したプライマリドキュメントの下に (折りたたみまたは部分的に展開) 表示できます。

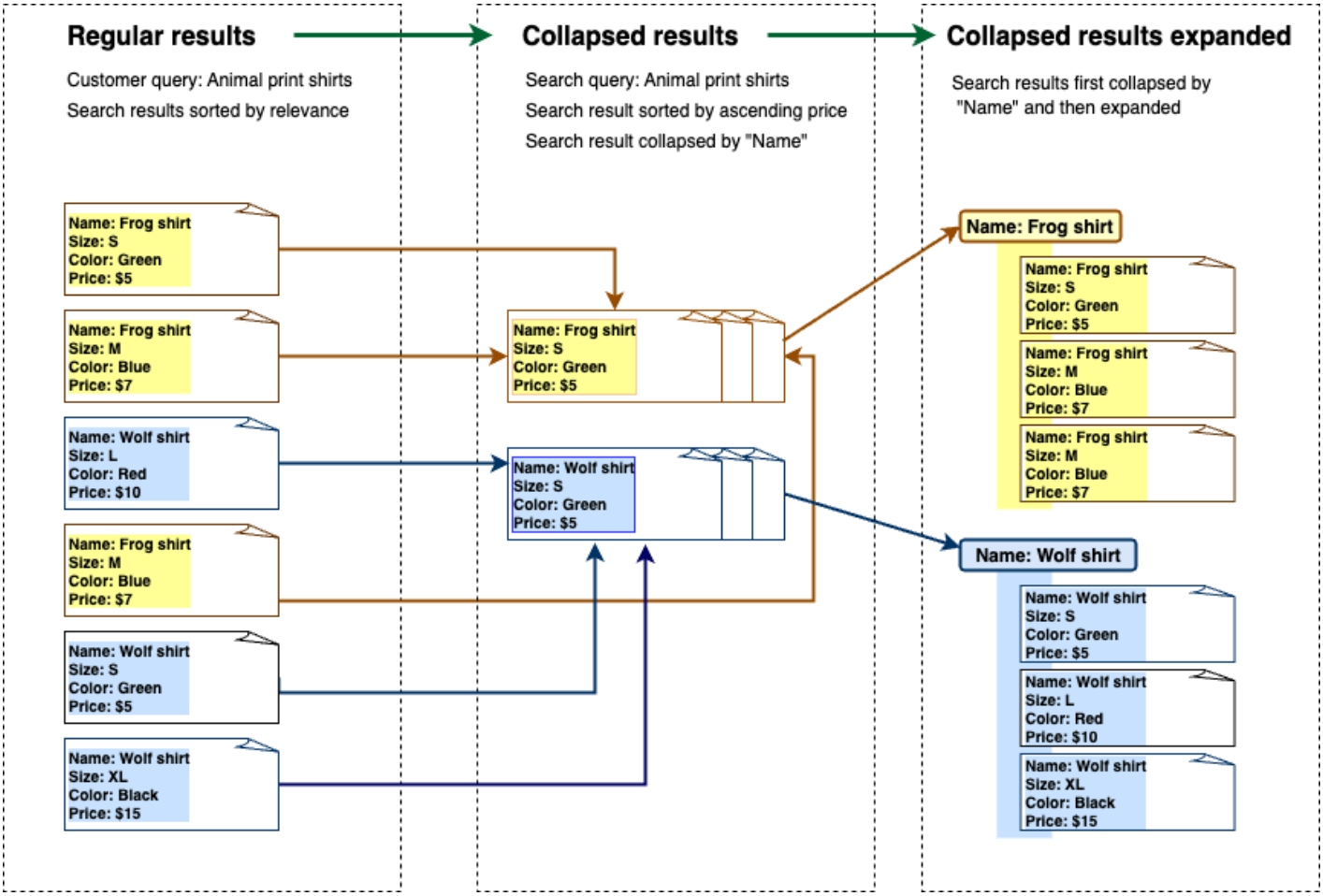
> **Note**
>
> クエリ結果の折りたたみと展開機能は、現在 Amazon Kendra API 経由でのみ利用できます。

これは、次のような検索状況で役に立ちます。

- インデックス内のドキュメントに複数のバージョンのコンテンツが存在する。エンドユーザーがインデックスをクエリする際に、重複を非表示または折りたたむことにより、最も関連性の高いバージョンのドキュメントを表示する場合。たとえば、インデックスに「NYC Leave policy」という名前のドキュメントの複数のバージョンが含まれている場合、「Type」属性/フィールドを使用して「HR」と「Legal」という特定のグループのドキュメントを折りたたむことができます。



- インデックスには、商品在庫など、1 種類の項目またはオブジェクトに関する固有の情報を含む複数のドキュメントが含まれています。アイテム情報の取得やソートしやすくするために、エンドユーザーがアイテムやオブジェクトにリンクされたすべてのドキュメントに 1 つの検索結果としてアクセス可能にする。以下の例では、お客様が「アニマルプリントシャツ」を検索すると、結果が名前でグループ化され、価格の昇順でソートされます。



結果の折りたたみ

類似または関連するドキュメントをグループ化するには、折りたたむ基準となる属性を指定する必要があります (たとえば、ドキュメントを折りたたんだりグループ化したりできます)。_categoryそのためには、Query API を呼び出し、CollapseConfigurationオブジェクトを使用して折りたたむ対象を指定しますDocumentAttributeKey。DocumentAttributeKey は、どのフィールドで検索結果を折りたたむかをコントロールします。サポートされている属性キーフィールドには、String と Number が含まれます。String list と Date タイプはサポートされていません。

ソート順を使用してプライマリドキュメントを選択する

折りたたまれたグループにプライマリードキュメントが表示されるように設定するには、SortingConfigurationsCollapseConfiguration以下のパラメータを使用します。たとえば、ドキュメントの最新バージョンを取得するには、折りたたまれた各グループを次の基準でソートします。_versionSortingConfigurations を使用して、ソートする属性/フィールドを 3 つまで指

定し、各属性/フィールドのソート順を指定できます。ソート属性数のクォータの引き上げをリクエストできます。

デフォルトでは、クエリの回答は、 Amazon Kendra 応答内の結果ごとに決定される関連性スコアでソートされます。デフォルトのソート順序を変更するには、ドキュメント属性をソート可能にし、Amazon Kendra その属性を使用してレスポンスをソートするように設定します。詳細については、「Sorting responses」を参照してください。

ドキュメントのキーストラテジーの欠損

ドキュメントに折りたたみ属性値がない場合、 Amazon Kendra には次の 3 つのカスタマイズオプションがあります。

- COLLAPSE を選択して、null 値または欠損値を含むすべてのドキュメントを 1 つのグループに折りたたみます。これはデフォルトの設定です。
- IGNORE を選択して、null 値または欠損値を含むドキュメントを無視します。無視したドキュメントは、クエリ結果に表示されません。
- EXPAND を選択して、null または欠落しているドキュメントをそれぞれ独自のグループに展開します。

結果の拡張

折りたたまれた検索結果グループを拡張するかどうかは、ExpandCollapseConfigurationオブジェクト内のパラメータを使用して選択できます。展開された結果では、グループのプライマリドキュメントを選択したときと同じソート順序が維持されます。

折りたたまれた検索結果グループの数を拡張するように設定するには、MaxResultItemstoExpandExpandConfigurationオブジェクト内のパラメータを使用します。例えば、この値を 10 に設定すると、100 個の結果グループのうち最初の 10 個だけが展開機能を持ちます。

折りたたまれたプライマリドキュメントごとに表示する展開結果の数を設定するには、MaxExpandResultsPerItem パラメータを使用します。例えば、この値を 3 に設定すると、折りたたまれたグループごとに最大 3 つの結果が表示されます。

Amazon Kendra 他の機能との相互作用

- 結果を折りたたんだり展開したりしても、ファセットの数は変化せず、表示される結果の総数にも影響しません。

- Amazon Kendra 注目の検索結果は、設定した折りたたみフィールドと同じフィールド値であっても折りたたまれません。
- 結果の折りたたみと展開は、DOCUMENTそのタイプの結果にのみ適用されます。

検索の関連性のチューニング

Amazon Kendra クエリは、関連性でランク付けされた検索結果を生成します。インデックス内の検索可能なフィールドまたは属性はすべて、このランキングに貢献します。

検索の関連性に対するフィールドまたは属性の影響は、[relevance tuning] (関連性チューニング) で変更できます。検索関連性のチューニングは、インデックスのチューニング構成を設定するインデックスレベルで手動で行うことも、インデックスレベルで設定された構成を上書きしてクエリレベルで行うこともできます。

関連性チューニングを使用すると、フィールドまたは属性に一致する用語がクエリ内にあると、レスポンスでの結果がブーストされます。また、一致する場合にドキュメントが受け取るブーストの量も指定します。関連性の調整によって がクエリレスポンスにドキュメントを含めることはなく Amazon Kendra 、 がドキュメントの関連性を判断するために Amazon Kendra 使用する要素の 1 つにすぎません。

インデックス内の特定のフィールドまたは属性をブーストして、特定のレスポンスをより重要度に割り当てることができます。例えば、誰かが「When is re:Invent?」と検索したとします。`_last_update_at` フィールドでドキュメントの鮮度の関連性を高めることができます。または、調査レポートのインデックスで、「ソース」フィールドで特定のデータソースをブーストすることもできます。

フォーラムやその他のサポートナレッジベースで一般的な投票や閲覧数に基づいてドキュメントを増やすこともできます。例えば、ブーストを組み合わせて、最近表示したドキュメントをブーストすることもできます。

ドキュメントが受け取るブーストの量は、`Importance` パラメータを使用して設定できます。`Importance` が高いほど、フィールドまたは属性によってドキュメントの関連性が高まります。インデックスをチューニングしたり、クエリレベルでチューニングしたりする場合は、必要なエフェクトが得られるまで、`Importance` パラメータを少しずつ指定します。検索結果を改善しているかどうかを判断するには、検索を実行し、結果を以前のクエリと比較します。

日付、数値、または文字列属性を指定して、インデックスをチューニングしたり、クエリレベルでチューニングしたりできます。インデックスレベルに限り、`StringList` タイプのフィールドや属性をチューニングできます。各フィールドまたは属性には、結果をブーストするタイミングに関する特定の基準があります。

- 日付フィールドまたは属性 - 日付フィールドには、`Duration`、`Freshness` および `RankOrder` の 3 つの特定の基準があります。

- Duration は、ブーストが適用される期間を設定します。例えば、期間を 86400 秒 (つまり 1 日) に設定すると、ブーストは 1 日後に減り始めます。重要度が高いほど、ブーストの低下が速くなります。
- Freshness ドキュメントがフィールドまたは属性に適用されるときのドキュメントがどの程度最近のものかを決定します。Freshness を [作成日] または [最終更新日] のいずれかのフィールドに適用する場合、より最近作成または最終更新されたドキュメントが、古いドキュメントよりも「新鮮」と見なされます。例えば、ドキュメント 1 の作成日が 11 月 14 日で、ドキュメント 2 の作成日が 11 月 5 日である場合、ドキュメント 1 はドキュメント 2 よりも「新鮮」です。また、ドキュメント 1 の最終更新日が 11 月 14 日で、ドキュメント 2 の最終更新日が 11 月 20 日である場合、ドキュメント 2 はドキュメント 1 よりも「新鮮」です。ドキュメントが新鮮であればあるほど、このブーストはより多く適用されます。インデックス内の Freshness フィールドは 1 つのみ設けることができます。
- RankOrder は昇順または降順でブーストを適用します。ASCENDING を指定した場合、新しい日付が優先されます。DESCENDING を指定した場合、古い日付が優先されます。

- 数値フィールドまたは属性 — 数値フィールドまたは属性の場合、フィールドまたは属性の関連性を判断するときに Amazon Kendra が使用するランク順を指定できます。ASCENDING を指定した場合、大きい数値が優先されます。DESCENDING を指定した場合、小さい数字が優先されます。
- 文字列フィールドまたは属性 - 文字列フィールドまたは属性の場合、フィールドのカテゴリを作成して、各カテゴリに異なるブーストを与えることができます。例えば、「Departmen」というフィールドまたは属性をブーストすると、「Legal」のドキュメントとは異なるブーストを「HR」からドキュメントに与えることができます。タイプ String のフィールドまたは属性をブーストできます。StringList フィールドは、インデックスレベルでのみブーストできます。

インデックスレベルでの関連性のチューニング

インデックスレベルでフィールドまたは属性の関連性をチューニングするには、コンソールを使用してインデックスの詳細または UpdateIndex API でチューニングを設定します。

次の例では、 _last_updated_atフィールドをドキュメントの Freshnessフィールドとして設定します。

```
"DocumentMetadataConfigurationUpdates" : [
    {
        "Name": "_last_updated_at",
        "Type": "DATE_VALUE",
        "Relevance": {
            "Freshness": TRUE,
```

```
            "Importance": 2
        }
    }
]
```

次の例では、「department」フィールドのカテゴリごとに異なる重要度を適用します。

```
"DocumentMetadataConfigurationUpdates" : [
    {
        "Name": "department",
        "Type": "STRING_VALUE",
        "Relevance": {
            "Importance": 2,
            "ValueImportanceMap": {
                "HR": 3,
                "Legal": 1
            }
        }
    }
]
```

クエリレベルでの関連性のチューニング

クエリレベルでフィールドまたは属性の関連性をチューニングするには、クエリ API を使用します。

クエリレベルでの関連性のチューニングは、コンソールではサポートされていません。

クエリレベルでチューニングすると、各テストのインデックス内のチューニング設定を手動で更新する必要がないため、関連性チューニングのテストプロセスが高速化されます。クエリでチューニング設定を渡すことで、ドキュメントの関連性をチューニングできます。次に、さまざまな設定から取得したさまざまな結果が表示されます。クエリで渡される設定は、インデックスレベルで設定された構成を上書きします。

次の例では、上記の例に示すように、「department」フィールドおよびインデックスレベルで設定された各部門カテゴリに適用される重要度を上書きします。ユーザーが検索クエリを入力すると、「department」フィールドの重要度は公正で、リーガル部門は HR 部門よりも重要度が高くなります。

```
"DocumentRelevanceOverrideConfigurations" : [
```

```
    {
        "Name": "department",
        "Type": "STRING_VALUE",
        "Relevance": {
            "Importance": 2,
            "ValueImportanceMap": {
                "HR": 2,
                "Legal": 8
            }
        }
    }
]
```

検索分析で同作を得る

Amazon Kendra 検索分析を使用すると、検索アプリケーションがユーザーによる情報の検索にどのように成功したか、または失敗しているかに関するインサイトを得ることができます。

Amazon Kendra Analytics は、ユーザーが検索アプリケーションとやり取りする方法と、検索アプリケーションの設定がどの程度効果的であるかを示すスナップショットを提供します。GetSnapshots API を使用するか、コンソールのナビゲーションパネルで分析を選択して、メトリクスデータを表示できます。

独自のカスタムビルドダッシュボードの GetSnapshots で、生成したデータをレンダリングできます。または、コンソールに提供される、ビジュアルグラフを含むメトリクスダッシュボードを使用することもできます。ビジュアルダッシュボードを使用すると、時間の経過に伴うユーザー行動の傾向やパターンを探したり、検索アプリケーション構成で問題を明らかにしたりできます。例えば、1 日あたりのクエリ数が一貫して増加し、一定の増加を示す折れ線グラフは、採用と使用量の増加を示している可能性があります。一方、急激な低下は、調査が必要な問題があることを示している可能性があります。

このメトリクスを使用して、さまざまなデータポイントの繋がりを確立し、ユーザーが情報を検索したり、ビジネスチャンスを発見したりする方法に関する問題を解決できます。例えば、「How does AI work?」というドキュメントは、検索結果内で最もクリックされたドキュメントで、検索された上位のクエリは「How does machine learning work?」です。これにより、ユーザーが使用する優先用語と言語が通知されます。これらの用語をドキュメントに統合したり、これらの用語にカスタムシノニムを使用して、ドキュメントをユーザーに対して検索しやすくすることができます。

検索のメトリクス

検索アプリケーションのパフォーマンスや、ユーザーが検索している情報を分析するための 10 のメトリクスがあります。メトリクスデータを取得するには、GetSnapshots を呼び出すときに取得するメトリクスデータの文字列名を指定します。

また、メトリクスデータを表示するには、時間間隔またはタイムウィンドウを指定する必要があります。時間間隔は、インデックスのタイムゾーンを使用します。データは次のタイムウィンドウで表示できます。

- THIS_WEEK: 今週。日曜日から始まり、今日の前日に終わる。
- ONE_WEEK_AGO: 先週。日曜日から始まり、次の土曜日に終わる。

- TWO_WEEKS_AGO: 先々週。日曜日から始まり、次の土曜日に終わる。
- THIS_MONTH: 今月。月の第一日から始まり、今日の前日に終わる。
- ONE_MONTH_AGO: 先月。月の第一日から始まり、月の末日に終了する。
- TWO_MONTHS_AGO: 先々月。月の第一日から始まり、月の末日に終了する。

コンソールでは、サポートされているタイムウィンドウは、[This week] (今週)、[Previous week] (先週)、[This month] (今月)、[Previous month] (先月) です。

クリックスルー率

検索結果内のドキュメントへのクリックスルーにつながるクエリの割合。これにより、検索アプリケーション構成がユーザーのクエリに関連する情報を見つけるのに役立つかどうかを把握できます。インスタント回答を返すクエリの場合、ユーザーはドキュメントをクリックして詳細を確認する必要がない場合があります。詳細については、「the section called “即時回答率”」を参照してください。クリックスルーフィードバックが確実に収集されるようにSubmitFeedbackするには、 を呼び出す必要があります。

GetSnapshots API を使用してクリックスルー率のデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

ゼロクリック率

検索結果内のドキュメントへのゼロクリックにつながるクエリの割合。これは、無関係な検索結果を提供するコンテンツのギャップを把握するのに役立ちます。インスタント回答を返すクエリの場合、ユーザーはドキュメントをクリックして詳細を確認する必要がない場合があります。詳細については、「the section called “即時回答率”」を参照してください。また、チューニング構成などの検索設定は、検索結果でドキュメントが返される方法に影響を与える可能性があります。

GetSnapshots API を使用してゼロクリックのデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

ゼロ検索結果率

ゼロ検索結果につながるクエリの割合。これは、無関係な検索結果を提供するコンテンツのギャップを把握するのに役立ちます。

GetSnapshots API を使用してゼロ検索結果率のデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

即時回答率

即時回答またはよくある質問が返されたクエリの割合。これは、情報提供における即時回答の役割を把握するのに役立ちます。

GetSnapshots API を使用して即時回答率のデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

上位のクエリ

ユーザーが検索した上位 100 件のクエリ。これは、どのクエリが人気があり、ユーザーが最も関心を持っている情報の種類はどれかを把握するのに役立ちます。

メトリクスには、クエリが検索された回数、ドキュメントに対するクリックスルーの割合、ドキュメントに対するクリックスルーなしの割合、クエリの検索結果の平均クリック深度、クエリの即時回答の割合、および最初の 10 件のクエリの検索結果の平均信頼度が含まれます。

GetSnapshots API を使用して上位クエリのデータを取得するには、QUERIES_BY_COUNT に metricType を指定します。コンソールのナビゲーションパネルで、[Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。[Query lists] (クエリリスト) の [Top queries] (上位クエリ) を選択します。

ゼロクリックの上位クエリ

検索結果内のゼロクリックにつながる上位 100 クエリ。これにより、一部のクエリに関連するドキュメントが不足している場合や、検索アプリケーションの構成で無関係な検索結果が返されている場合に、コンテンツのギャップを把握できます。インスタント回答を返すクエリの場合、ユーザーはドキュメントをクリックして詳細を確認する必要がない場合があります。詳細については、「the section called “即時回答率”」を参照してください。

メトリクスには、クエリがゼロクリックにつながった回数、クエリに対するゼロクリックの割合、クエリの即時回答の割合、およびクエリの最初の 10 件の検索結果の平均信頼度が含まれます。

GetSnapshots API を使用してゼロクリックを伴う上位クエリのデータを取得するには、QUERIES_BY_ZERO_CLICK_RATE に metricType を指定します。コンソールのナビゲーショ

ンパネルで、[Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。[Query lists] (クエリリスト) の [Top zero click queries] (上位ゼロクリッククエリ) を選択します。

ゼロ検索結果の上位クエリ

ゼロ検索結果につながる上位 100 クエリ。これにより、クエリに関連するドキュメントがないコンテンツのギャップを把握するのに役立ちます。または、検索結果 0 件につながる可能性がある特殊な用語でユーザーがクエリを実行し、カスタムシノニム を作成してこれに対処するよう促す場合があります。

メトリクスには、クエリのゼロ検索結果につながる回数、クエリの検索結果ゼロの割合、およびすべてのクエリに対するクエリが検索された回数の割合が含まれます。

GetSnapshots API を使用してゼロ検索結果の上位クエリのデータを取得するには、QUERIES_BY_ZERO_RESULT_RATE に metricType を指定します。コンソールのナビゲーションパネルで、[Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。[Query lists] (クエリリスト) の [Top zero result queries] (上位ゼロ結果クエリ) を選択します。

クリックされた上位ドキュメント

検索結果内で最もクリックされたドキュメントの上位 100 件。これにより、ユーザーが情報をクエリするときに、どのドキュメントまたは検索結果が最も関連しているかを把握できます。

メトリクスには、ドキュメントがクリックされた回数、ドキュメントがユーザーから受け取るいいね! の数 (高評価)、ドキュメントがユーザーから受け取る嫌いの数 (低評価) が含まれます。

GetSnapshots API を使用してクリックされた上位ドキュメントのデータを取得するには、DOCS_BY_CLICK_COUNT に metricType を指定します。コンソールのナビゲーションパネルで、[Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。[Query lists] (クエリリスト) の [Top clicked documents] (クリックされた上位ドキュメント) を選択します。

合計クエリ数

ユーザーが検索したクエリの合計数。これにより、ユーザーが検索アプリケーションにどの程度関わっているかを理解できます。

GetSnapshots API を使用してクエリの合計数のデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

合計ドキュメント

インデックス内のドキュメントの合計数。これにより、インデックスのサイズとクエリの総数を比較して、クエリの量に対して適切な数のドキュメントがあるかどうかをチェックできます。

GetSnapshots API を使用してドキュメントの合計数のデータを取得するには、AGG_QUERY_DOC_METRICS に metricType を指定します。ナビゲーションパネルの [Analytics] (分析) を選択して、このメトリクスをコンソールで表示することもできます。

メトリクスデータの取得例

次のコードは、先月の上位のクエリのデータを取得する例です。

Console

先月の上位クエリを取得するには

1. 左側のナビゲーションペインで、[Indexes] (インデックス) で、インデックスを選択し、[Analytics] (分析) を選択します。
2. [分析] ページで、[今週] ボタンをクリックし、データを取得するための期間を [先月] に変更します。
3. [Analytics] (分析) ページの [Query lists] (クエリリスト) で、[Top queries] (上位クエリ) を選択します。

CLI

先月の上位クエリを取得するには

```
aws kendra get-snapshots \
--index-id index-id \
--interval "ONE_MONTH_AGO" \
--metric-type "QUERIES_BY_COUNT"
```

Python

先月の上位クエリを取得するには

```
import boto3

kendra = boto3.client("kendra")
```

```
index_id = "index-id"
interval = "ONE_MONTH_AGO"
metric_type = "QUERIES_BY_COUNT"

snapshots_response = kendra.get_snapshots(
    IndexId = index_id,
    Interval = interval,
    MetricType = metric_type
)

print("Top queries data: " + snapshots_response["snapshotsData"])
```

Java

先月の上位クエリを取得するには

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.GetSnapshotsRequest;
import software.amazon.awssdk.services.kendra.model.GetSnapshotsResponse;

public class TopQueriesExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        String indexId = "indexID";
        String interval = "ONE_MONTH_AGO";
        String metricType = "QUERIES_BY_COUNT";

        GetSnapshotsRequest getSnapshotsRequest = GetSnapshotsRequest
            .builder()
            .indexId(indexId)
            .interval(interval)
            .metricType(metricType)
            .build();

        GetSnapshotsResponse getSnapshotsResponse =
 kendra.getSnapshots(GetSnapshotsRequest);

        System.out.println(String.format("Top queries data: ",
 getSnapshotsResponse.snapshotsData()))
```

メトリクスから実用的なインサイトまで

実用的なインサイトは、生データから抽出された有意義な情報であり、行動や意思決定の指針として使用されます。メトリクスから意味を抽出し、それらを使用して実用的なインサイトを導き出すには、メトリクスを個別に調べるだけでなく、メトリクス間のつながりも確立することが重要です。

例えば、クリック数がゼロの上位クエリは、「Which regions are currently available?」 のようになります。ただし、インスタント回答率も 100% です。これは、利用可能なリージョンに関する情報を提供する検索結果やドキュメントをクリックしなくても、ユーザーがこの質問に対する回答を受け取ることを示唆しています。ゼロクリックだけを見た場合、完全なストーリーは得られず、このクエリを処理する際の検索アプリケーション構成の成功について間違った結論を出す可能性があります。

実用的なインサイトの別の例は、ビジネスチャンスの発見です。企業は多くの場合、検索メトリクスを分析してお客様を成長させる機会を求めています。最もクリックされたドキュメントは「利用可能なリージョン」です。これに加えて、検索された上位クエリのほとんどは、海洋リージョンでの製品の可用性に関する質問に関連しています。回答の一部として、利用可能なリージョンに関するより多くの情報への即時回答率が 100%、クリックスルー率は高くなります。これは、このリージョンに製品やサービスに対する関心と需要があることを示唆しています。

検索分析の視覚化とレポート

トレンドデータを含む 5 つのメトリクスがあり、時間の経過とともにトレンドやパターンを視覚化して検索できます。コンソールを使用する場合、トレンドデータのグラフが表示されます。API を使用する場合、トレンドデータを取得して、独自のグラフまたは視覚化された情報を作成できます。コンソールのほとんどのグラフは、選択したタイムウィンドウにおける日次データポイントをプロットします。

コンソールには、表示したいグラフとトップリストを選択できるメトリクスのダッシュボードが表示されます。ダッシュボードに表示されるメトリクスを CSV 形式でエクスポートするには、[Analytics] (分析) ホームページの [Export] (エクスポート) を選択します。これらのレポートは、ビジネスドキュメントまたはプレゼンテーションに含めることができます。

以下のメトリクスを視覚化することができます。

合計クエリグラフ

1 日に発行されたクエリ数の折れ線グラフ。このグラフは、毎日のユーザーエンゲージメントのパターンを視覚化するのに役立ちます。例としては、ユーザーエンゲージメントの着実な増減や、検索

アプリケーションのクラッシュやウェブサイトの問題により、クエリが 0 件まで大幅に低下することがあります。

API を使用する場合は、TREND_QUERY_DOC_METRICS を指定して、これらのデータを取得できます。データを使用して独自のグラフを作成することも、コンソールで提供されているグラフを使用することもできます。

クリックスルー率グラフ

1 日あたりのクリックスルーの割合の折れ線グラフ。このグラフは、毎日のクリックスルー率のパターンを視覚化するのに役立ちます。例としては、クリックスルー率の着実な増減、即時回答の減少がクリックスルーの増加に影響を与える可能性があります。

API を使用する場合は、TREND_QUERY_DOC_METRICS を指定して、これらのデータを取得できます。データを使用して独自のグラフを作成することも、コンソールで提供されているグラフを使用することもできます。

ゼロクリック率グラフ

1 日あたりのゼロクリックの割合の折れ線グラフ。このグラフは、毎日のゼロクリック率のパターンを視覚化するのに役立ちます。例としては、ゼロクリック率の着実な増減、即時回答の増加がゼロクリックの増加に影響を与える可能性があります。

API を使用する場合は、TREND_QUERY_DOC_METRICS を指定して、これらのデータを取得できます。データを使用して独自のグラフを作成することも、コンソールで提供されているグラフを使用することもできます。

ゼロ検索結果率グラフ

1 日あたりのゼロ検索結果の割合の折れ線グラフ。このグラフは、毎日のゼロ検索結果率のパターンを視覚化するのに役立ちます。例としては、ゼロ検索結果率の着実な増減、インデックス内のドキュメント数の急激な減少がゼロ検索結果の増加に影響を与える可能性があります。

API を使用する場合は、TREND_QUERY_DOC_METRICS を指定して、これらのデータを取得できます。データを使用して独自のグラフを作成することも、コンソールで提供されているグラフを使用することもできます。

即時回答率グラフ

即時回答またはよくある質問が返されたクエリの割合の折れ線グラフ。このグラフは、毎日の即時回答率のパターンを視覚化するのに役立ちます。例としては、質問応答タイプのクエリの着実な増減、またはクリックスルーの減少が即時回答の増加に影響を与える可能性があります。

API を使用する場合は、TREND_QUERY_DOC_METRICS を指定して、これらのデータを取得できます。データを使用して独自のグラフを作成することも、コンソールで提供されているグラフを使用することもできます。

増分学習のためのフィードバックの送信

Amazon Kendra インクリメンタルラーニングを使用して検索結果を改善します。クエリからのフィードバックを使用して、増分学習によってランク付けアルゴリズムが改善され、検索結果が最適化され、精度が向上します。

例えば、ユーザーが「health care benefits」という語句を検索するとします。複数のユーザーがリストから 2 番目の結果を一貫して選択すると、 Amazon Kendra はその結果を第 1 位の結果に格上げします。ブーストは時間が経つにつれて減少するため、ユーザーが結果を選択しなくなると、 Amazon Kendra 最終的にはその結果が削除され、代わりに別の人気の高い結果が表示されます。これにより、関連性、年齢、 Amazon Kendra コンテンツに基づいて結果に優先順位を付けることができます。

増分学習は、すべてのインデックスとすべてのサポートされているドキュメントタイプで有効になっています。

Amazon Kendra フィードバックを提供するとすぐに学習を開始しますが、フィードバックの結果が表示されるまでに 24 時間以上かかることがあります。 Amazon Kendra には、 AWS コンソール、JavaScript 検索結果ページに含めることができるライブラリ、使用できる API という 3 つの方法でフィードバックを送信できます。

Amazon Kendra 次の 2 種類のユーザーフィードバックを受け付けます。

- クリック - ユーザーが選択したクエリの結果に関する情報。フィードバックには、結果 ID と、検索結果が選択された日時の Unix タイムスタンプが含まれます。

  クリックフィードバックを送信するには、アプリケーションがユーザーのアクティビティからクリック情報を収集し、その情報を Amazon Kendraに送信する必要があります。クリック情報は、コンソール、 JavaScript ライブラリ、 Amazon Kendra API で収集できます。
- 関連性 - ユーザーが通常提供する検索結果の関連性に関する情報。フィードバックには、結果 ID と関連性インジケータ (`RELEVANT` または `NOT_RELEVANT`) が含まれます。ユーザーが関連性情報を決定します。

  関連性のフィードバックを送信するには、ユーザーがクエリ結果の適切な関連性を選択し、その情報を Amazon Kendraに送信できるフィードバックメカニズムをアプリケーションに提供する必要があります。関連情報はコンソールと Amazon Kendra API でのみ収集できます。

フィードバックは、インデックスがアクティブである間に使用されます。フィードバックは、送信先のインデックスにのみ影響し、インデックス間や異なるアカウントで使用することはできません。

Amazon Kendra インデックスをクエリするときは、追加のユーザーコンテキストを提供する必要があります。ユーザーコンテキストを提供すると、 Amazon Kendra フィードバックが 1 人のユーザーによるものか、複数のユーザーによるものかを判断し、それに応じて検索結果を調整できます。

ユーザーコンテキストを指定すると、クエリのフィードバックは、コンテキストで指定した特定のユーザーに関連付けられます。ユーザーコンテキストを指定しない場合は、クエリをグループ化および集計するために使用する訪問者 ID を指定できます。

ユーザーコンテキストまたは訪問者 ID を提供しない場合、フィードバックは匿名で、他の匿名のフィードバックと共に集計されます。

次のコードは、ユーザーコンテキストをトークンまたは訪問者 ID として含める方法を示しています。

```
response = kendra.query(
    QueryText = query,
    IndexId = index,
    UserToken = {
        Token = "token"
    })

    OR

    response = kendra.query(
    QueryText = query,
    IndexId = index,
    VisitorId = "visitor-id")
```

ウェブアプリケーションでは、cookie、ロケーション、またはブラウザユーザーを使用して、各ユーザーの訪問者 ID を生成できます。

ヘッドクエリはクエリ量が最大で、クリックスルーフィードバックを提供することにより全体的な精度を向上させるのに十分な情報が提供されます。テールクエリはまれですが、主題のエキスパートは、これらのクエリの精度を向上させるために、関連性がないフィードバックを送信する必要があります。

コンソール以外にも、 JavaScript ライブラリと SubmitFeedbackAPI の 2 つの方法のいずれかを使用できます。フィードバックの収集は 1 つの方法のみをご使用ください。最良の結果を得るには、クエリを実行してから 24 時間以内にフィードバックを送信する必要があります。

トピック

Amazon Kendra JavaScript ライブラリを使用してフィードバックを送信する

Amazon Kendra には、 JavaScript 検索結果ページにクリックフィードバックを追加できるライブラリが用意されています。ライブラリを使用するには、検索結果を表示するスクリプトタグをクライアントコードに挿入し、結果リストの各ドキュメントリンクに情報を追加します。ユーザーがドキュメントを表示するリンクを選択すると、クリック情報が Amazon Kendraに送信されます。

このライブラリは、ES6/ES2015 JavaScript バージョンをサポートするブラウザーで動作します。

ステップ 1: 検索アプリケーションに script タグを挿入します。 Amazon Kendra

Amazon Kendra 検索結果をレンダリングするクライアントコードに、<script>タグを挿入し、JavaScript ライブラリへの参照を追加します。

```
<script>
 (function(w, d, s, c, g, n) {
   if(!w[n]) {
     w[n] = w[n] || function () {
           (w[n].q = w[n].q || []).push(arguments);
     }
     w[n].st = new Date().getTime();
     w[n].ep = g;
     var e = document.createElement(s),
         j = document.getElementsByTagName(s)[0];
     e.async = 1;
     e.src = c;
     e.type = 'module';
     j.parentNode.insertBefore(e, j);
   }
 })(window, document, 'script',
 'library download URL',
 'feedback endpoint',
 'kendraFeedback');
```

```
</script>
```

このスクリプトは、 Amazon Kendra ホストされている CDN JavaScript からライブラリを非同期的にダウンロードし、kendraFeedbackオプションパラメータを設定できるというグローバル変数を初期化します。

########### URL ################、インデックスをホストする地域に基づいて次の表の識別子に置き換えます。 Amazon Kendra

リージョン	URL のダウンロード	フィードバックエンドポイント
us-east-1	https://d2zm0lpns956f8.cloudfront.net/ksf-v1.js	https://ujxwp5s92h.execute-api.us-east-1.amazonaws.com/prod/submit
us-east-2	https://d2crv7fufeg244.cloudfront.net/ksf-v1.js	https://i6h76zwzf3.execute-api.us-east-2.amazonaws.com/prod/submit
us-west-2	https://d2iezfpnpcoujy.cloudfront.net/ksf-v1.js	https://wg6nim909c.execute-api.us-west-2.amazonaws.com/prod/submit
ca-central-1	https://d1zbkfomowykaq.cloudfront.net/ksf-v1.js	https://budi8txevj.execute-api.ca-central-1.amazonaws.com/prod/submit
eu-west-1	https://d3gptlxtulu4us.cloudfront.net/ksf-v1.js	https://po2b11740b.execute-api.eu-west-1.amazonaws.com/prod/submit
ap-southeast-1	https://d1vvuam7g4taoe.cloudfront.net/ksf-v1	https://9je5uw7t5l.execute-api.ap-southeast-1.amazonaws.com/prod/submit
ap-southeast-2	https://dopqntoe6z0ce.cloudfront.net/ksf-v1.js	https://oovf4nvjj7.execute-api.ap-southeast-2.amazonaws.com/prod/submit

リージョン	URL のダウンロード	フィードバックエンドポイント
ap-south-1	https://d1ts9ouelsmk3g.cloudfront.net/ksf-v1.js	https://k1abnmd43b.execute-api.ap-south-1.amazonaws.com/prod/submit
ap-northeast-1	https://d3w0ybsa293kb4.cloudfront.net/ksf-v1.js	https://wg7rz0uzjh.execute-api.ap-northeast-1.amazonaws.com/prod/submit
eu-west-2	https://d1tsrujswld1d1.cloudfront.net/ksf-v1.js	https://qi7mct3x7f.execute-api.eu-west-2.amazonaws.com/prod/submit

例えば、インデックスが米国東部 (バージニア北部) の場合は、*[library download URL]* (ライブラリダウンロード URL) は `https://d2zm0lpns956f8.cloudfront.net/ksf-v1.js`、*[feedback endpoint]* (フィードバックエンドポイント) は `https://ujxwp5s92h.execute-api.us-east-1.amazonaws.com/prod/submit` になります。

Amazon Kendra JavaScript ライブラリにはオプションで 2 つの設定ができます。

- `disableCookies`— デフォルトでは、ユーザを一意に識別する Cookie Amazon Kendra を設定します。これを `true` に設定して、cookie を無効にします。

  ```
  kendraFeedback('disableCookie', 'true | false');
  ```

  `searchDivClassName` - デフォルトでは、 Amazon Kendra は検索結果ページのすべてのリンクでクリックをモニタリングします。これを `<div>` のクラス名に設定し、指定されたクラスのリンクのみをモニタリングします。

  ```
  kendraFeedback('searchDivClassName', 'class name');
  ```

ステップ 2: フィードバックトークンを検索結果に追加する

結果ページで、`data-kendra-token` という HTML 属性を追加し、クエリレスポンスからのドキュメントへのリンクを含むアンカータグまたは直接の親 div タグに移動します。例:

```
<a href="document location" data-kendra-token="feedback token value"></a>
OR
<div data-url="document location" data-kendra-token="feedback token value"></div>
```

クエリレスポンスには、feedbackToken フィールドのトークンが含まれます。トークンは、ユーザーが選択した場合にレスポンスを一意に識別します。トークンの値を data-kendra-token 属性に割り当てます。 Amazon Kendra JavaScript ライブラリは、ユーザーが結果を選択したときにこのトークンを探し、 Amazon Kendra フィードバックとしてエンドポイントに送信します。

Amazon Kendra JavaScript ライブラリは、フィードバックトークンと、結果が選択された時刻や固有の訪問者 ID などのメタデータのみを送信します。

ステップ 3: フィードバックスクリプトをテストする

JavaScript ライブラリが正しく設定され、適切なエンドポイントにフィードバックが送信されていることを確認するには、次の操作を行います。この例では Chrome ブラウザを使用します。

1. ブラウザで ウェブデベロッパーツールを開きます。Chrome で、ブラウザの右上隅の [Chrome menu] (Chrome メニュー) を開き、[More tools] (その他のツール) を選択して [Developer tools] (デベロッパーツール) を選択します。
2. Amazon Kendra JavaScript コンソールタブにライブラリに関するエラーがないことを確認します。
3. 検索を行い、任意の結果を選択します。デベロッパーツールの [Network] (ネットワーク) タブを選択します。フィードバックエンドポイントに送信されたリクエスト、結果のトークン、200 OK ステータスが表示されます。

Amazon Kendra API を使用してフィードバックを送信する

Amazon Kendra API を使用してクエリのフィードバックを送信するには、SubmitFeedbackAPI を使用してください。クエリを識別するには、クエリが適用されるインデックスのインデックス ID と、クエリ API からの応答で返されるクエリ ID を指定します。

以下の例は、 Amazon Kendra API を使用してクリックおよび関連性のフィードバックを送信する方法を示しています。ClickFeedbackItems および RelevanceFeedbackItems 配列を介して、複数のフィードバックセットを送信できます。この例では、1 回のクリックと 1 つの関連性フィードバック項目を送信します。フィードバックの送信では現在時刻が使用されます。

検索 (AWS SDK) に関するフィードバックを送信するには

1. 必要な値を指定した次のコード例を使用できます。

   a. index id— クエリが適用されるインデックスの ID。

   b. query id— フィードバックを提供したいクエリ。

   c. result id— フィードバックを提供したいクエリ結果の ID。クエリレスポンスには、結果 ID が含まれます。

   d. relevance value— RELEVANT (クエリ結果が関連している) または NOT_RELEVANT (クエリ結果は関連性がない) のいずれか。

   Python

```
import boto3
import time

kendra = boto3.client("kendra")

# Provide the index ID
index_id = "index-id"
# Provide the query ID
query_id = "query-id"
# Provide the search result ID
result_id = "result-id"

# Configure the feedback item
feedback_item = {"ClickTime": int(time.time()),
    "ResultId":result_id}

# Configure the relevance value
relevance_value = "RELEVANT"
relevance_item = {"RelevanceValue": relevance_value,
    "ResultId": result_id
    }

response = kendra.submit_feedback(
    QueryId = query_id,
    IndexId = index_id,
    ClickFeedbackItems = [feedback_item],
    RelevanceFeedbackItems = [relevance_item]
)
```

```
print("Submitted feedback for query: " + query_id)
```

Java

```
package com.amazonaws.kendra;

import java.time.Instant;
import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.ClickFeedback;
import software.amazon.awssdk.services.kendra.model.RelevanceFeedback;
import software.amazon.awssdk.services.kendra.model.RelevanceType;
import software.amazon.awssdk.services.kendra.model.SubmitFeedbackRequest;
import software.amazon.awssdk.services.kendra.model.SubmitFeedbackResponse;

public class SubmitFeedbackExample {
    public static void main(String[] args) {
        KendraClient kendra = KendraClient.builder().build();

        SubmitFeedbackRequest submitFeedbackRequest = SubmitFeedbackRequest
            .builder()
            .indexId("IndexId")
            .queryId("QueryId")
            .clickFeedbackItems(
                ClickFeedback
                .builder()
                .clickTime(Instant.now())
                .resultId("ResultId")
                .build())
            .relevanceFeedbackItems(
                RelevanceFeedback
                .builder()
                .relevanceValue(RelevanceType.RELEVANT)
                .resultId("ResultId")
                .build())
            .build();
```

```
        SubmitFeedbackResponse response =
 kendra.submitFeedback(submitFeedbackRequest);

        System.out.println("Feedback is submitted");
    }
}
```

2. コードを実行します。フィードバックが送信されると、コードがメッセージを表示します。

インデックスへのカスタムシノニムの追加

カスタムシノニムをインデックスに追加するには、シソーラスファイルでシノニムを指定します。Amazon Kendra シノニムの使用には、ビジネス固有の用語や特殊な用語を含めることができます。などの一般的な英語の同義語はシソーラスファイルに組み込まれているためleader, head、Amazon Kendra シソーラスファイルには含めないでください。 Amazon Kendra DOCUMENTレスポンスタイプやレスポンスタイプを含むすべてのレスポンスタイプのシノニムをサポートします。QUESTION_ANSWER ANSWER Amazon Kendra 現在、ストップワードのフラグが付いたシノニムの追加はサポートされていません。これは、将来のリリースに組み込まれます。

Amazon Kendra シノニム同士を相関させます。たとえば、Dynamo, Amazon DynamoDBシノニムペアを使用すると Dynamo がと相関します。 Amazon Kendra Amazon DynamoDB「What is dynamo?」というクエリは、次に、「What is?」などのドキュメントを返します。 Amazon DynamoDB Amazon Kendra シノニムを使用すると、相関関係をより簡単に把握できます。

シソーラスファイルはバケットに保存されるテキストファイルです。 Amazon S3 シソーラスをインデックスに追加する を参照してください。

シソーラスファイルは Solr シノニム形式を使用します。 Amazon Kendra 索引あたりのシソーラス数には制限があります。クォータを参照してください。

シノニムは、次のシナリオで役立ちます。

- 例えば、NLP, Natural Language Processing など、従来の英語のシノニムではない専門用語。
- 複雑な意味的関連を持つ固有名詞。例えば、機械学習では、cost, loss, model performance など、これらは一般の人が理解しにくい名詞です。
- 例えば、Elastic Compute Cloud, EC2 などの異なる形式の製品名。
- 製品名など、ドメイン固有またはビジネス固有の用語。例えば、Route53, DNS。

次のシナリオではシノニムを使用しないでください。

- leader, head など、一般的な英語のシノニム。これらのシノニムはドメイン固有ではなく、これらのシナリオでシノニムを使用すると、意図しない効果が生じる可能性があります。
- teh => the などの誤字。
- 名詞の複数形や所有格、形容詞の比較形および最上級形、動詞の過去形、過去分詞形、進行形のような形態学的変種。比較形容詞と最上級形容詞の一例は、good, better, best です。

- WHO などのユニグラム (1 単語) ストップワード。ユニグラムストップワードはシソーラスでは許可されず、検索から除外されます。例えば、WHO => World Health Organization は拒否されます。W.H.O. をシノニム用語として使用できますが、ストップワードをマルチワードシノニムの一部として使うことができます。例えば、of は許可されますが、United States of America は許可されません。

カスタムシノニムを使用すると、クエリをビジネス固有のシノニムを対象とするように拡張できるため、ビジネス固有の用語の理解が容易になります。 Amazon Kendraシノニムは検索の精度を向上させることができますが、シノニムがレイテンシーにどのように影響するかを理解して最適化することが重要です。

シノニムの一般的なルールは、クエリ内のシノニムと一致して拡張される用語が多いほど、レイテンシーへの影響が大きくなります。待ち時間に影響するその他の要因には、インデックスに登録されるドキュメントの平均サイズ、インデックスのサイズ、検索結果のフィルタリング、インデックスにかかる全体的な負荷などがあります。 Amazon Kendra シノニムと一致しないクエリは影響を受けません。

シノニムがレイテンシーにどのように影響するかに関する一般的なガイドライン:

ユースケース	レイテンシーの増加*
一般的な自然言語またはキーワードクエリ (それぞれ 3～5 語)	15% 未満
1 つのクエリ用語が 3 つのシノニムに展開されます	
約 50 万件のドキュメント (ドキュメントごとに抽出されたテキストの平均は 10.48 KB) または 30,000 のよくある質問/質問ペアのインデックス	

*パフォーマンスは、インデックスでのシノニムと構成の特定の使用方法によって異なります。検索のパフォーマンスをテストして、特定のユースケースに対してより正確なベンチマークを取得することをお勧めします。

シソーラスが大きく、用語の拡張率が高く、レイテンシーの増加が許容範囲内にない場合は、次のいずれかまたは両方を試してください。

- シソーラスをトリミングして、拡張率 (用語ごとのシノニム数) を減らします。
- 用語の全体的な範囲 (シソーラスの行数) をトリミングします。

または、プロビジョニングキャパシティ (仮想ストレージユニット) を増やして、レイテンシーの増加を相殺することもできます。

トピック

シソーラスファイルの作成

Amazon Kendra シソーラスファイルは、Solr シノニムリスト形式のシノニムのリストを含む UTF-8 でエンコードされたファイルです。シソーラスファイルは 5 MB 未満である必要があります。

シノニムのマッピングを指定するには、2 つの方法があります。

- 双方向シノニムは、用語をカンマで区切ったリストとして指定します。ユーザーがいずれかの用語を検索する場合、リスト内のすべての用語がドキュメント検索に使用されます。これには、クエリされた元の用語も含まれます。
- 単方向シノニムは、用語をシノニムにマッピングするために、「=>」で区切られた用語として指定されます。記号「=>」の左側にある用語をユーザーが検索すると、その用語は右側の用語にマップされ、シノニムを使用しているドキュメントを検索します。その逆はマッピングされないため、単方向になります。

シノニム自体では大文字と小文字が区別されますが、マップ先の用語では大文字と小文字は区別されません。例えば `ML => Machine Learning` の場合、ユーザーが「ML」や「ml」を検索したり、大文字小文字のその他の組み合わせを使用したりすると、「Machine Learning」にマッピングされます。逆に、`Machine Learning => ML` をマッピングすると、「Machine Learning」や「machine learning」、およびその他に組み合わせが、「ML」にマッピングされます。

シノニムでは、特殊文字が完全に一致するものは検索されません。たとえば、"" を検索すると、dead-letter-queue「デッドレターキュー」 Amazon Kendra に一致する文書が返されます (ハイ

フンなし)。文書に "dead-letter-queue" のようにハイフンが含まれている場合、 Amazon Kendra 一致する用語を検索する際に文書が処理されてハイフンが削除されます。

ストップワードやよく使われる単語を含むシノニムの場合、 Amazon Kendra ストップワードを含む用語と一致するドキュメントを返します。たとえば、「搭乗中」と「搭乗中」をマッピングするシノニムルールを作成できます。ストップワードだけをシノニムに使用することはできません。たとえば、「on」を検索しても、「on」 Amazon Kendra を含むすべてのドキュメントは返されません。

一部のシノニムルールは無視されます。たとえば、a => ba => aはルールであるが無視され、ルールとしてカウントされない。

用語数は、シソーラスファイル内の一意の用語の数です。以下のサンプルファイルにはAWS CodeStar、、ML、Machine Learningautoscaling groupASG、などの用語が含まれています。

シソーラスごとのシノニムルールには最大数があり、用語ごとのシノニム数にも上限があります。詳細については、「のクォータ Amazon Kendra」を参照してください。

次の例は、シノニムルールを含むシソーラスファイルを示しています。各行には 1 つのシノニムルールが含まれています。空白行とコメントは無視されます。

```
# Lines starting with pound are comments and blank lines are ignored.

# Synonym relationships can be defined as unidirectional or bidirectional
 relationships.

# Unidirection relationships are represented by any term sequence
# on the left hand side (LHS) of "=>" followed by synonyms on the right hand side (RHS)
CodeStar => AWS CodeStar
# This will map CodeStar to AWS CodeStar, but not vice-versa

# To map terms vice versa
ML => Machine Learning
Machine Learning => ML

# Multiple synonym relationships may be defined in one line as well by comma
 seperation.
autoscaling group, ASG => Auto Scaling group, autoscaling
# The above is equivalent to:
# autoscaling group => Auto Scaling group, autoscaling
# ASG => Auto Scaling group, autoscaling
```

```
# Bi-directional synonyms are comma separated terms with no "=>"
DNS, Route53, Route 53
# DNS, Route53, and Route 53 map to one another and are interchangeable at match time
# The above is equivalent to:
# DNS => Route53, Route 53
# Route53 => DNS, Route 53
# Route 53 => DNS, Route53

# Overlapping LHS terms will be merged
Beta => Alpha
Beta => Gamma
Beta, Delta
# is equivalent to:
# Beta => Alpha, Gamma, Delta
# Delta => Beta

# Each line contains a single synonym rule.
# Synonym rule count is the total number of lines defining synonym relationships
# Term count is the total number of unique terms for all rules.
# Comments and blanks lines do not count.
```

シソーラスをインデックスに追加する

以下の手順は、シノニムを含むシソーラスファイルをインデックスに追加する方法を示しています。更新されたシソーラスファイルの効果を確認するのに最大 30 分かかる場合があります。シソーラスファイルの詳細については、シソーラスファイルの作成 を参照してください。

Console

シソーラスを追加するには

1. 左側のナビゲーションペインで、シノニムのリスト、シソーラスを追加するインデックスで、[Synonyms] (シノニム) を選択します。
2. [Synonym] (シノニム) ページで、[Add Thesaurus] (シソーラスを追加) を選択します。
3. [Define thesaurus] (シソーラスを定義) で、シソーラスに名前とオプションの説明を付けます。
4. 「シソーラス設定」で、シソーラスファイルへのパスを指定します。 Amazon S3 ファイルは 5 MB より小さくなければなりません。
5. [IAM Role] では、ロールを選択するか、[Create a new role] を選択し、ロール名を指定して新しいロールを作成します。 Amazon Kendra このロールを使用して、 Amazon S3 ユー

ザーに代わってリソースにアクセスします。IAM ロールには「AmazonKendra-」というプレフィックスが付いています。

6. [Save] (保存) をクリックして設定を保存し、シソーラスを追加します。シソーラスが取り込まれると、そのシソーラスがアクティブになり、結果でシノニムが強調表示されます。シソーラスファイルの効果を確認するのに最大 30 分かかる場合があります。

CLI

を使用して索引に類義語辞典を追加するには、以下を呼び出します。 AWS CLIcreate-thesaurus

```
aws kendra create-thesaurus \
--index-id index-id \
--name "thesaurus-name" \
--description "thesaurus-description" \
--source-s3-path "Bucket=bucket-name,Key=thesaurus/synonyms.txt" \
--role-arn role-arn
```

list-thesauri を呼び出してシソーラスのリストを表示します。

```
aws kendra list-thesauri \
--index-id index-id
```

シソーラスの詳細を表示するには、describe-thesaurus を呼び出します。

```
aws kendra describe-thesaurus \
--index-id index-id \
--index-id thesaurus-id
```

シソーラスファイルの効果を確認するのに最大 30 分かかる場合があります。

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Create a thesaurus")
```

```
thesaurus_name = "thesaurus-name"
thesaurus_description = "thesaurus-description"
thesaurus_role_arn = "role-arn"

index_id = "index-id"

s3_bucket_name = "bucket-name"
s3_key = "thesaurus-file"
source_s3_path= {
    'Bucket': s3_bucket_name,
    'Key': s3_key
}

try:
    thesaurus_response = kendra.create_thesaurus(
        Description = thesaurus_description,
        Name = thesaurus_name,
        RoleArn = thesaurus_role_arn,
        IndexId = index_id,
        SourceS3Path = source_s3_path
    )

    pprint.pprint(thesaurus_response)

    thesaurus_id = thesaurus_response["Id"]

    print("Wait for Kendra to create the thesaurus.")

    while True:
        # Get thesaurus description
        thesaurus_description = kendra.describe_thesaurus(
            Id = thesaurus_id,
            IndexId = index_id
        )
        # If status is not CREATING quit
        status = thesaurus_description["Status"]
        print("Creating thesaurus. Status: " + status)
        if status != "CREATING":
            break
        time.sleep(60)

except ClientError as e:
        print("%s" % e)
```

```
print("Program ends.")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.CreateThesaurusRequest;
import software.amazon.awssdk.services.kendra.model.CreateThesaurusResponse;
import software.amazon.awssdk.services.kendra.model.DescribeThesaurusRequest;
import software.amazon.awssdk.services.kendra.model.DescribeThesaurusResponse;
import software.amazon.awssdk.services.kendra.model.S3Path;
import software.amazon.awssdk.services.kendra.model.ThesaurusStatus;

public class CreateThesaurusExample {

  public static void main(String[] args) throws InterruptedException {

    KendraClient kendra = KendraClient.builder().build();

    String thesaurusName = "thesaurus-name";
    String thesaurusDescription = "thesaurus-description";
    String thesaurusRoleArn = "role-arn";

    String s3BucketName = "bucket-name";
    String s3Key = "thesaurus-file";
    String indexId = "index-id";

    System.out.println(String.format("Creating a thesaurus named %s",
 thesaurusName));
    CreateThesaurusRequest createThesaurusRequest = CreateThesaurusRequest
        .builder()
        .name(thesaurusName)
        .indexId(indexId)
        .description(thesaurusDescription)
        .roleArn(thesaurusRoleArn)
        .sourceS3Path(S3Path.builder()
            .bucket(s3BucketName)
            .key(s3Key)
            .build())
        .build();
```

```
    CreateThesaurusResponse createThesaurusResponse =
 kendra.createThesaurus(createThesaurusRequest);
    System.out.println(String.format("Thesaurus response %s",
 createThesaurusResponse));

    String thesaurusId = createThesaurusResponse.id();

    System.out.println(String.format("Waiting until the thesaurus with ID %s is
 created.", thesaurusId));

    while (true) {
      DescribeThesaurusRequest describeThesaurusRequest =
 DescribeThesaurusRequest.builder()
          .id(thesaurusId)
          .indexId(indexId)
          .build();
      DescribeThesaurusResponse describeThesaurusResponse =
 kendra.describeThesaurus(describeThesaurusRequest);
      ThesaurusStatus status = describeThesaurusResponse.status();
      if (status != ThesaurusStatus.CREATING) {
        break;
      }

      TimeUnit.SECONDS.sleep(60);
    }

    System.out.println("Thesaurus creation is complete.");
  }
}
```

シソーラスを更新する

シソーラスの作成後に設定を変更することができます。シソーラス名や IAM 情報などの詳細を変更できます。シソーラスファイル Amazon S3 パスの場所を変更することもできます。シソーラスファイルへのパスを変更すると、 Amazon Kendra は既存のシソーラスを、更新されたパスで指定されたシソーラスに置き換えます。

更新されたシソーラスファイルの効果を確認するのに最大 30 分かかる場合があります。

> **Note**
>
> シソーラスファイルに検証エラーまたは構文エラーがある場合、以前にアップロードされたシソーラスファイルは保持されます。

以下の手順は、シソーラスの詳細を変更する方法を示しています。

Console

シソーラスの詳細を変更するには

1. 左側のナビゲーションペインの変更するインデックスで、[Synonyms] (シノニム) を選択します。
2. [Synonym] (シノニム) ページで、変更するシソーラスを選択し、[Edit] (編集) を選択します。
3. [Update thesaurus] (シソーラスを更新) ページで、シソーラスの詳細を更新します。
4. (オプション)「シソーラスファイルパスの変更」を選択し、 Amazon S3 新しいシソーラスファイルへのパスを指定します。既存のシソーラスファイルは、指定したファイルに置き換えられます。パスを変更しない場合、 Amazon Kendra 既存のパスからシソーラスを再ロードします。

   [現在のシソーラスファイルを保存] Amazon Kendra を選択しても、シソーラスファイルは再ロードされません。
5. [Save] (保存) を選択して設定を保存します。

既存のシソーラスパスからシソーラスをリロードすることもできます。

既存のパスからシソーラスをリロードするには

1. 左側のナビゲーションペインの変更するインデックスで、[Synonyms] (シノニム) を選択します。
2. [シノニム] ページで、リロードするシソーラスを選択し、[更新] を選択します。
3. [シソーラスファイルのリロード] ページで、シソーラスファイルを更新することを確認します。

CLI

シソーラスを更新するには、update-thesaurus を呼び出します。

```
aws kendra update-thesaurus \
--index-id index-id \
--name "thesaurus-name" \
--description "thesaurus-description" \
--source-s3-path "Bucket=bucket-name,Key=thesaurus/synonyms.txt" \
--role-arn role-arn
```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra = boto3.client("kendra")

print("Update a thesaurus")

thesaurus_name = "thesaurus-name"
thesaurus_description = "thesaurus-description"
thesaurus_role_arn = "role-arn"

thesaurus_id = "thesaurus-id"
index_id = "index-id"

s3_bucket_name = "bucket-name"
s3_key = "thesaurus-file"
source_s3_path= {
    'Bucket': s3_bucket_name,
    'Key': s3_key
}

try:
    kendra.update_thesaurus(
        Id = thesaurus_id,
        IndexId = index_id,
        Description = thesaurus_description,
        Name = thesaurus_name,
        RoleArn = thesaurus_role_arn,
```

```
        SourceS3Path = source_s3_path
    )

    print("Wait for Kendra to update the thesaurus.")

    while True:
        # Get thesaurus description
        thesaurus_description = kendra.describe_thesaurus(
            Id = thesaurus_id,
            IndexId = index_id
        )
        # If status is not UPDATING quit
        status = thesaurus_description["Status"]
        print("Updating thesaurus. Status: " + status)
        if status != "UPDATING":
            break
        time.sleep(60)

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.UpdateThesaurusRequest;
import software.amazon.awssdk.services.kendra.model.DescribeThesaurusRequest;
import software.amazon.awssdk.services.kendra.model.DescribeThesaurusResponse;
import software.amazon.awssdk.services.kendra.model.S3Path;
import software.amazon.awssdk.services.kendra.model.ThesaurusStatus;

public class UpdateThesaurusExample {

  public static void main(String[] args) throws InterruptedException {

    KendraClient kendra = KendraClient.builder().build();

    String thesaurusName = "thesaurus-name";
    String thesaurusDescription = "thesaurus-description";
    String thesaurusRoleArn = "role-arn";
```

```
    String s3BucketName = "bucket-name";
    String s3Key = "thesaurus-file";

    String thesaurusId = "thesaurus-id";
    String indexId = "index-id";

    UpdateThesaurusRequest updateThesaurusRequest = UpdateThesaurusRequest
        .builder()
        .id(thesaurusId)
        .indexId(indexId)
        .name(thesaurusName)
        .description(thesaurusDescription)
        .roleArn(thesaurusRoleArn)
        .sourceS3Path(S3Path.builder()
            .bucket(s3BucketName)
            .key(s3Key)
            .build())
        .build();
    kendra.updateThesaurus(updateThesaurusRequest);

    System.out.println(String.format("Waiting until the thesaurus with ID %s is
updated.", thesaurusId));

    // a new source s3 path requires re-consumption by Kendra
    // and so can take as long as a Create Thesaurus operation
    while (true) {
      DescribeThesaurusRequest describeThesaurusRequest =
DescribeThesaurusRequest.builder()
          .id(thesaurusId)
          .indexId(indexId)
          .build();
      DescribeThesaurusResponse describeThesaurusResponse =
kendra.describeThesaurus(describeThesaurusRequest);
      ThesaurusStatus status = describeThesaurusResponse.status();
      if (status != ThesaurusStatus.UPDATING) {
        break;
      }

      TimeUnit.SECONDS.sleep(60);
    }

    System.out.println("Thesaurus update is complete.");
  }
```

```
}
```

シソーラスを削除する

以下の手順は、シソーラスを削除する方法を示しています。

Console

1. 左側のナビゲーションペインの変更するインデックスで、[Synonyms] (シノニム) を選択します。
2. [Synonym] (シノニム) ページで、削除するシソーラスを選択します。
3. [Thesaurus detail] (シソーラスの詳細) ページで、[Delete] (削除) を選択し、削除を確定します。

CLI

を使用して索引の類義語辞典を削除するには、以下を呼び出します。 AWS CLI`delete-thesaurus`

```
aws kendra delete-thesaurus \
--index-id index-id \
--id thesaurus-id
```

Python

```
import boto3
from botocore.exceptions import ClientError

kendra = boto3.client("kendra")

print("Delete a thesaurus")

thesaurus_id = "thesaurus-id"
index_id = "index-id"

try:
    kendra.delete_thesaurus(
        Id = thesaurus_id,
        IndexId = index_id
```

```
    )

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

```
package com.amazonaws.kendra;

import software.amazon.awssdk.services.kendra.KendraClient;
import software.amazon.awssdk.services.kendra.model.DeleteThesaurusRequest;

public class DeleteThesaurusExample {

  public static void main(String[] args) throws InterruptedException {

    KendraClient kendra = KendraClient.builder().build();

    String thesaurusId = "thesaurus-id";
    String indexId = "index-id";

    DeleteThesaurusRequest updateThesaurusRequest = DeleteThesaurusRequest
        .builder()
        .id(thesaurusId)
        .indexId(indexId)
        .build();
    kendra.deleteThesaurus(updateThesaurusRequest);
  }
}
```

検索結果の強調表示

シノニムの強調表示はデフォルトでオンになっています。ハイライト情報は Amazon Kendra SDK と CLI のクエリ結果に含まれます。SDK または CLI Amazon Kendra を使用して操作する場合は、結果の表示方法を決定します。

シノニム強調表示には強調表示タイプ THESAURUS_SYNONYM があります。ハイライトの詳細については、「Highlight」オブジェクトを参照してください。

チュートリアル: Amazon Kendra を使用したメタデータに富んだインテリジェントな検索ソリューションの構築

このチュートリアルでは、Amazon Kendra、Amazon Comprehend、Amazon Simple Storage Service(S3)、AWS CloudShell を使用して、エンタープライズデータ向けのメタデータに富んだ自然言語ベースのインテリジェント検索ソリューションを構築する方法を説明します。

Amazon Kendra は、非構造化自然言語データリポジトリの検索インデックスを構築できるインテリジェントな検索サービスです。お客様が関連する回答を簡単に検索してフィルタリングできるようにするには、Amazon Comprehend を使用してデータからメタデータを抽出し、Amazon Kendra 検索インデックスに取り込みます。

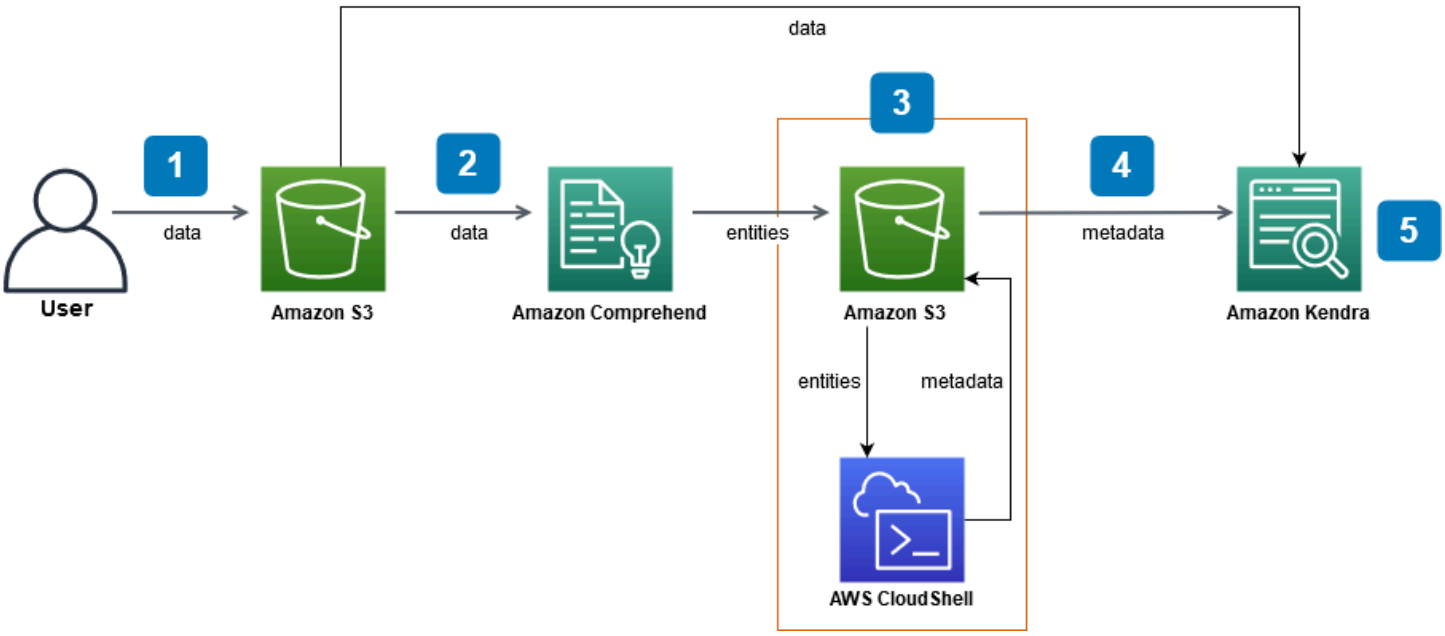
Amazon Comprehend は、エンティティを識別できる自然言語処理 (NLP) サービスです。エンティティは、データ内の人、場所、位置、組織、およびオブジェクトのリファレンスです。

このチュートリアルでは、ニュース記事のサンプルデータセットを使用して、エンティティを抽出し、メタデータに変換し、Amazon Kendra インデックスに取り込んで検索を実行します。追加されたメタデータを使用すると、これらのエンティティのサブセットを使用して検索結果をフィルタリングでき、検索の精度が向上します。このチュートリアルに従うことで、専門的な機械学習知識がなくても、エンタープライズデータの検索ソリューションを作成する方法を学習します。

このチュートリアルでは、以下のステップで検索ソリューションを構築する方法を示します。

1. Amazon S3 にニュース記事のサンプルデータセットを保存する。
2. Amazon Comprehend を使用してデータからエンティティを抽出します。
3. Python 3 スクリプトを実行してエンティティを Amazon Kendra インデックスメタデータ形式に変換し、このメタデータを S3 に保存します。
4. Amazon Kendra 検索インデックスを作成し、データとメタデータを取り込みます。
5. 検索インデックスのクエリ。

以下の図に、このワークフローを示しています。



このチュートリアルを完了する予定時間: 1 時間

推定コスト: このチュートリアルのアクションには、AWS アカウントの変更を引き起こすものがあります。各サービスのコストの詳細については、Amazon S3、Amazon Comprehend、AWS CloudShell、および Amazon Kendra の料金ページを参照してください。

トピック

前提条件

このチュートリアルを完了するには、以下のリソースが必要です。

- AWS アカウント。AWS アカウントをお持ちの場合は、Amazon Kendra のセットアップの手順を行い、AWS アカウントをセットアップします。

- AWS コマネジメントコンソールにアクセスするための、Windows、macOS、および Linux を実行している開発用コンピュータ。詳細については、AWS マネジメントコンソールの設定を参照してください。
- AWS Identity and Access Management (IAM) ユーザー。アカウントの IAM ユーザーとグループをセットアップする方法については、IAM ユーザーガイドの開始方法セクションを参照してください。

  AWS Command Line Interface を使用している場合、このチュートリアルを完了するために必要な基本的なアクセス権限を付与するために、IAM ユーザーに以下のポリシーをアタッチする必要があります。

  詳細については、IAM ポリシーの作成およびIAM アイデンティティアクセス許可の追加と削除を参照してください。
- AWS リージョンサービスリスト。レイテンシーを減らすには、Amazon Comprehend と Amazon Kendra の両方でサポートされている地理的な場所に最も近い AWS リージョンを選択する必要があります。
- (オプション) AWS Key Management Service。このチュートリアルでは暗号化を使用しませんが、特定のユースケースで暗号化のベストプラクティスを使用することをお勧めします。
- (オプション) Amazon Virtual Private Cloud。このチュートリアルでは VPC を使用しませんが、VPC のベストプラクティスを使用して特定のユースケースでデータセキュリティを確保することをお勧めします。

ステップ 1: Amazon S3 にドキュメントを追加する

データセットで Amazon Comprehend エンティティ分析ジョブを実行する前に、データ、メタデータ、および Amazon Comprehend エンティティ分析出力をホストする Amazon S3 バケットを作成します。

トピック

- サンプルデータセットをダウンロードする
- Amazon S3 バケットの作成
- S3 バケットにデータフォルダとメタデータフォルダを作成する
- 入力データをアップロードする

サンプルデータセットをダウンロードする

Amazon Comprehend がデータに対してエンティティ分析ジョブを実行できるようにするには、データセットをダウンロードして抽出し、S3 バケットにアップロードする必要があります。

データセットをダウンロードして抽出するには (コンソール)

1. デバイス上の tutorial-dataset.zip フォルダをダウンロードします。
2. `tutorial-dataset` フォルダを抽出して `data` フォルダにアクセスします。

データセットをダウンロードして抽出するには (チュートリアル)

1. `tutorial-dataset` をダウンロードするには、ターミナルウィンドウを開き、以下のコマンドを実行します。

   Linux

   ```
   curl -o path/tutorial-dataset.zip https://docs.aws.amazon.com/kendra/latest/dg/samples/tutorial-dataset.zip
   ```

   実行する条件は以下のとおりです。

   - *path/* は、zip フォルダを保存する場所のローカルファイルパスです。

   macOS

   ```
   curl -o path/tutorial-dataset.zip https://docs.aws.amazon.com/kendra/latest/dg/samples/tutorial-dataset.zip
   ```

   実行する条件は以下のとおりです。

   - *path/* は、zip フォルダを保存する場所のローカルファイルパスです。

   Windows

   ```
   curl -o path/tutorial-dataset.zip https://docs.aws.amazon.com/kendra/latest/dg/samples/tutorial-dataset.zip
   ```

実行する条件は以下のとおりです。

- *path/* は、zip フォルダを保存する場所のローカルファイルパスです。

2. zip フォルダからデータを抽出するには、ターミナルウィンドウで次のコマンドを実行します。

Linux

```
unzip path/tutorial-dataset.zip -d path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した zip フォルダへのローカルファイルパスです。

macOS

```
unzip path/tutorial-dataset.zip -d path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した zip フォルダへのローカルファイルパスです。

Windows

```
tar -xf path/tutorial-dataset.zip -C path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した zip フォルダへのローカルファイルパスです。

このステップを完了すると、抽出されたファイルが `tutorial-dataset` という解凍したフォルダにあるはずです。このフォルダには、Apache 2.0 オープンソースのアトリビューションのある `README` ファイルと、このチュートリアルのデータセットが含まれている `data` というフォルダがあります。データセットは `.story` 拡張子のある 100 個のファイルで構成されます。

Amazon S3 バケットの作成

サンプルデータフォルダをダウンロードして抽出したら、Amazon S3 バケットに保存します。

> **Important**
>
> Amazon S3 バケットの名前はすべての AWS 全体で一意である必要があります。

S3 バケットを作成するには (コンソール)

1. AWS Management Console にサインインし、Amazon S3 コンソール https://console.aws.amazon.com/s3/ を開きます。
2. [Buckets] (バケット) で、[Create bucket] (バケットの作成) を選択します。
3. [Bucket name] (バケット名) に、一意の名前を入力します。
4. [Region] (リージョン) では、バケットを格納する AWS リージョンを選択します。

   > **Note**
   >
   > Amazon Comprehend と Amazon Kendra の両方をサポートするリージョンを選択する必要があります。作成後にバケットのリージョンを変更することはできません。

5. [Block Public Access settings for this bucket] (このバケットのパブリックアクセス設定をブロックする)、[Bucket Versioning] (バケットバージョニング)、および [Tags] (タグ) はデフォルト設定のままにしておきます。
6. [Default encryption] (デフォルトの暗号化) には、[Disable] (無効) を選択します。
7. [Advanced settings] (詳細設定) はデフォルト設定のままにしておきます。
8. バケットの設定を確認して、[Create bucket] (バケットの作成) を選択します。

S3 バケットを作成するには (AWS CLI)

1. S3 バケットを作成するには、AWS CLI で [create-bucket] コマンドを使用します。

   Linux

   ```
   aws s3api create-bucket \
           --bucket DOC-EXAMPLE-BUCKET \
           --region aws-region \
           --create-bucket-configuration LocationConstraint=aws-region
   ```

   実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名、
- *aws-region* は、バケットを作成するリージョンです。

macOS

```
aws s3api create-bucket \
        --bucket DOC-EXAMPLE-BUCKET \
        --region aws-region \
        --create-bucket-configuration LocationConstraint=aws-region
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名、
- *aws-region* は、バケットを作成するリージョンです。

Windows

```
aws s3api create-bucket ^
        --bucket DOC-EXAMPLE-BUCKET ^
        --region aws-region ^
        --create-bucket-configuration LocationConstraint=aws-region
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名、
- *aws-region* は、バケットを作成するリージョンです。

> **Note**
>
> Amazon Comprehend と Amazon Kendra の両方をサポートするリージョンを選択する必要があります。作成後にバケットのリージョンを変更することはできません。

2. バケットが正常に作成されたことを確認するには、[list] コマンドを使用します。

Linux

```
aws s3 ls
```

macOS

```
aws s3 ls
```

Windows

```
aws s3 ls
```

S3 バケットにデータフォルダとメタデータフォルダを作成する

S3 バケットを作成した後、その中のフォルダにデータフォルダとメタデータフォルダを作成します。

S3 バケットにフォルダーを作成するには (コンソール)

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. [Buckets] (バケット) で、バケットのリストからバケットの名前をクリックします。
3. [Objects] (オブジェクト) タブから、[Create folder] (フォルダの作成) を選択します。
4. 新しいフォルダ名に、**`data`** を入力します。
5. 暗号化設定については、[Disable] (無効) を選択します。
6. [Create folder] (フォルダの作成) を選択します。
7. ステップ 3 から 6 を繰り返して Amazon Kendra メタデータを保存する別のフォルダーを作成し、ステップ 4 **`metadata`** で作成したフォルダに名前を付けます。

S3 バケットにフォルダを作成するには (AWS CLI)

1. S3 バケットで `data` フォルダを作成するには、AWS CLI で [put-object] コマンドを使用します。

   Linux

   ```
   aws s3api put-object \
   ```

```
        --bucket DOC-EXAMPLE-BUCKET \
        --key data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

macOS

```
aws s3api put-object \
        --bucket DOC-EXAMPLE-BUCKET \
        --key data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

Windows

```
aws s3api put-object ^
        --bucket DOC-EXAMPLE-BUCKET ^
        --key data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

2. S3 バケットで metadata フォルダを作成するには、AWS CLI で [put-object] コマンドを使用します。

Linux

```
aws s3api put-object \
        --bucket DOC-EXAMPLE-BUCKET \
        --key metadata/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

macOS

```
aws s3api put-object \
        --bucket DOC-EXAMPLE-BUCKET \
        --key metadata/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

Windows

```
aws s3api put-object ^
        --bucket DOC-EXAMPLE-BUCKET ^
        --key metadata/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

3. フォルダが正常に作成されたことを確認するには、[list] コマンドを使用してバケットの内容をチェックします。

Linux

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

macOS

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

Windows

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* はバケット名です。

入力データをアップロードする

データフォルダとメタデータフォルダを作成したら、サンプルデータセットを data フォルダにアップロードします。

サンプルデータセットをデータフォルダにアップロードするには (コンソール)

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. [Buckets] (バケット) で、バケットのリストからバケットの名前、data の順にクリックします。
3. [Upload] (アップロード)、[Add files] (ファイルの追加) の順に選択します。
4. ダイアログボックスで、ローカルデバイスの tutorial-dataset フォルダ内の data フォルダで、すべてのファイルを選択し、[Open] (開く) をクリックします。
5. [Destination] (送信先)、[Permissions] (アクセス許可)、および [Properties] (プロパティ) はデフォルト設定のままにしておきます。
6. [Upload] (アップロード) を選択します。

サンプルデータセットをデータフォルダにアップロードするには (AWS CLI)

1. サンプルデータを data フォルダにアップロードするには、AWS CLI で [copy] コマンドを使用します。

   Linux

   ```
   aws s3 cp path/tutorial-dataset/data s3://DOC-EXAMPLE-BUCKET/data/ --recursive
   ```

   実行する条件は以下のとおりです。

- *path/* は、デバイス上の tutorial-dataset フォルダへのファイルパス、
- *DOC-EXAMPLE-BUCKET* はバケット名です。

macOS

```
aws s3 cp path/tutorial-dataset/data s3://DOC-EXAMPLE-BUCKET/data/ --recursive
```

実行する条件は以下のとおりです。

- *path/* は、デバイス上の tutorial-dataset フォルダへのファイルパス、
- *DOC-EXAMPLE-BUCKET* はバケット名です。

Windows

```
aws s3 cp path/tutorial-dataset/data s3://DOC-EXAMPLE-BUCKET/data/ --recursive
```

実行する条件は以下のとおりです。

- *path/* は、デバイス上の tutorial-dataset フォルダへのファイルパス、
- *DOC-EXAMPLE-BUCKET* はバケット名です。

2. データセットファイルが data フォルダに正常にアップロードされたことを確認する場合は、AWS CLI で [list] コマンドを使用します。

Linux

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

macOS

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

Windows

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/data/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

このステップを完了すると、data フォルダに S3 バケットのデータセットが保存されされた、metadata フォルダが空になります。このフォルダは Amazon Kendra メタデータを保存します。

ステップ 2: Amazon Comprehend でエンティティ分析ジョブを実行する

S3 バケットにサンプルデータセットを保存した後、Amazon Comprehend エンティティ分析ジョブを実行して、ドキュメントからエンティティを抽出します。これらのエンティティは Amazon Kendra カスタム属性を形成し、インデックスの検索結果をフィルタリングするのに役立ちます。詳細については、エンティティの検出を参照してください。

トピック

- Amazon Comprehend でエンティティ分析ジョブを実行する

Amazon Comprehend でエンティティ分析ジョブを実行する

データセットからエンティティを抽出するには、Amazon Comprehend エンティティ分析ジョブを実行します。

このステップで AWS CLI を使用している場合、最初に Amazon Comprehend の AWS IAM ロールとポリシー作成してアタッチし、その後エンティティ分析ジョブを実行します。サンプルデータでエンティティ分析ジョブを実行するには、Amazon Comprehend は次のものが必要です。

- 信頼されたエンティティとして認識する AWS Identity and Access Management (IAM) ロール
- S3 バケットへのアクセス許可を付与する IAM ロールに添付された AWS IAM ポリシー

詳細については、「How Amazon Comprehend works with IAM」および「Identity-based policy examples for Amazon Comprehend」を参照してください。

Amazon Comprehend エンティティ分析ジョブを実行するには (コンソール)

1. Amazon Comprehend コンソール (https://console.aws.amazon.com/comprehend/) を開きます。

   > **Important**
   >
   > Amazon S3 バケットを作成したリージョンと同じリージョンに存在することを確認します。別のリージョンにいる場合は、トップナビゲーションバーの [Region selector] (リージョンセレクタ) から S3 バケットを作成した AWS リージョンを選択します。

2. [Launch Amazon Comprehend] (Amazon Comprehend の起動) を選択します。
3. 左側のナビゲーションペインで、[Analysis jobs] (分析ジョブ) を選択します。
4. [Create job] (ジョブの作成) を選択します。
5. [Job settings] (ジョブの設定) セクションで、以下の操作を行います。
   a. [Name] (名前)に **`data-entities-analysis`** と入力します。
   b. [Analysis type] (分析タイプ) で、[Entities] (エンティティ) を選択します。
   c. [Language] (言語) で、[English] (英語) を選択します。
   d. [Job encryption] (ジョブの暗号化) は無効のままにしておきます。
6. [Input data] (入力データ) セクションで、以下の操作を行います。
   a. [Data source] (データソース) で、[My documents] (マイドキュメント) を選択します。
   b. [S3 location] (S3 の場所) で、[Browse S3] (S3 を閲覧する) を選択します。
   c. [Choose resources] (リソースの選択) については、バケットのリストからバケットの名前をクリックします。
   d. [Objects] (オブジェクト) で、`data` のオプションボタンを選択し、[Choose] (選択) をクリックします。
   e. [Input format] (入力形式) で、[One document per file] (ファイルあたり 1 つのドキュメント) を選択します。

7. [Output data] (出力データ) セクションで、以下の操作を行います。

   a. [S3 locatio] (S3 の場所) で、[Browse S3] (S3 を閲覧する)、バケットのリストからバケットのオプションボックスの順に選択し、[Choose] (選択) をクリックします。

   b. [Encryption] (暗号化) は無効のままにしておきます。

8. [Access permissions] (アクセス許可) セクションで、以下の操作を行います。

   a. [IAM role] (IAM ロール) で、[Create an IAM role] (IAM ロールの選択) を選択します。

   b. [Permissions to access] (アクセスの許可) で、[Input and Output S3 buckets] (S3 バケットの入力と出力) を選択します。

   c. [Name suffix] (サフィックスに名前を付ける) で、**`comprehend-role`** と入力します。このロールは、Amazon S3 バケットへのアクセスを提供します。

9. [VPC settings] (VPC 設定) は、デフォルト設定のままにしておきます。

10. [Create job] (ジョブの作成) を選択します。

Amazon Comprehend エンティティ分析ジョブを実行するには (AWS CLI)

1. 信頼されたエンティティとして認識する Amazon Comprehend の IAM ロールを作成してアタッチするには、以下の操作を行います。

   a. 次の信頼ポリシーを、ローカルデバイスのテキストエディタで `comprehend-trust-policy.json` という JSON ファイルとして保存します。

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

   b. `comprehend-role` という IAM ロールを作成するには、保存した `comprehend-trust-policy.json` ファイルをアタッチしてくださいそれをファイルして、[create-role] コマンドを使用します。

Linux

```
aws iam create-role \
          --role-name comprehend-role \
          --assume-role-policy-document file://path/comprehend-trust-
policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-trust-policy.json フォルダへのファイルパスです。

macOS

```
aws iam create-role \
          --role-name comprehend-role \
          --assume-role-policy-document file://path/comprehend-trust-
policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-trust-policy.json フォルダへのファイルパスです。

Windows

```
aws iam create-role ^
          --role-name comprehend-role ^
          --assume-role-policy-document file://path/comprehend-trust-
policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-trust-policy.json フォルダへのファイルパスです。

c. Amazon リソースネーム (ARN) をテキストエディタにコピーし、comprehend-role-arn としローカルに保存します。

> **Note**
>
> ARN は、*arn:aws:iam::123456789012:role/comprehend-role* というような形式になります。Amazon Comprehend 分析ジョブを実行するには、comprehend-role-arn として保存した ARN が必要になります。

2. S3 バケットへのアクセス許可を付与する IAM ポリシーを IAM ロールに作成し、アタッチするには、次の操作を行います。

   a. 次の信頼ポリシーを、ローカルデバイスのテキストエディタで comprehend-S3-access-policy.json という JSON ファイルとして保存します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ],
            "Effect": "Allow"
```

```
        }
    ]
}
```

b. S3 バケットにアクセスする comprehend-S3-access-policy という IAM ポリシーを作成するには、[create-policy] コマンドを使用します。

Linux

```
aws iam create-policy \
          --policy-name comprehend-S3-access-policy \
          --policy-document file://path/comprehend-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-S3-access-policy.json フォルダへのファイルパスです。

macOS

```
aws iam create-policy \
          --policy-name comprehend-S3-access-policy \
          --policy-document file://path/comprehend-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-S3-access-policy.json フォルダへのファイルパスです。

Windows

```
aws iam create-policy ^
          --policy-name comprehend-S3-access-policy ^
          --policy-document file://path/comprehend-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の comprehend-S3-access-policy.json フォルダへのファイルパスです。

c. Amazon リソースネーム (ARN) をテキストエディタにコピーし、comprehend-S3-access-arn としローカルに保存します。

> **Note**
>
> ARN は、*arn:aws:iam::123456789012:role/comprehend-S3-access-policy* というような形式になります。comprehend-S3-access-policy を IAM ロールにアタッチするには、comprehend-S3-access-arn として保存した ARN が必要になります。

d. comprehend-S3-access-policy を IAM ロールにアタッチするには、[attach-role-policy] コマンドを使用します。

Linux

```
aws iam attach-role-policy \
          --policy-arn policy-arn \
          --role-name comprehend-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、comprehend-S3-access-arn として保存した ARN です。

macOS

```
aws iam attach-role-policy \
          --policy-arn policy-arn \
          --role-name comprehend-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、comprehend-S3-access-arn として保存した ARN です。

Windows

```
aws iam attach-role-policy ^
          --policy-arn policy-arn ^
          --role-name comprehend-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、comprehend-S3-access-arn として保存した ARN です。

3. Amazon Comprehend エンティティ分析ジョブを実行するには、[start-entities-detection-job] コマンドを使用します。

Linux

```
aws comprehend start-entities-detection-job \
        --input-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/
data/,InputFormat=ONE_DOC_PER_FILE \
        --output-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/ \
        --data-access-role-arn role-arn \
        --job-name data-entities-analysis \
        --language-code en \
        --region aws-region
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* S3 バケットの名前、
- *role-arn* は、comprehend-role-arn として保存した ARN です。
- *aws-region* は、AWS リージョンです。

macOS

```
aws comprehend start-entities-detection-job \
        --input-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/
data/,InputFormat=ONE_DOC_PER_FILE \
        --output-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/ \
        --data-access-role-arn role-arn \
        --job-name data-entities-analysis \
        --language-code en \
        --region aws-region
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* S3 バケットの名前、
- *role-arn* は、comprehend-role-arn として保存した ARN です。

- *aws-region* は、AWS リージョンです。

Windows

```
aws comprehend start-entities-detection-job ^
        --input-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/
data/,InputFormat=ONE_DOC_PER_FILE ^
        --output-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/ ^
        --data-access-role-arn role-arn ^
        --job-name data-entities-analysis ^
        --language-code en ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* S3 バケットの名前、
- *role-arn* は、comprehend-role-arn として保存した ARN です。
- *aws-region* は、AWS リージョンです。

4. エンティティ分析 JobId をコピーし、テキストエディタで comprehend-job-id という名前を付けて保存します。JobId は、エンティティ分析ジョブのステータスを追跡するのに役立ちます。
5. エンティティ分析ジョブの進行状況を追跡するには、[describe-entities-detection-job] コマンドを使用します。

Linux

```
aws comprehend describe-entities-detection-job \
        --job-id entities-job-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws comprehend describe-entities-detection-job \
        --job-id entities-job-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws comprehend describe-entities-detection-job ^
        --job-id entities-job-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

JobStatus が COMPLETED に変わるまで数分かかることがあります。

このステップを完了すると、Amazon Comprehend はエンティティ分析結果を、S3 バケット内の自動生成されたフォルダ内の output フォルダに、output.tar.gz 圧縮ファイルとして保存します。分析ジョブのステータスが完了していることを確認し、次のステップに進みます。

ステップ 3: エンティティ分析出力を Amazon Kendra メタデータとして書式設定する

Amazon Comprehend によって抽出されたエンティティを Amazon Kendra インデックスに必要なメタデータ形式に変換するには、Python 3 スクリプトを実行します。変換の結果は、Amazon S3 バケット内の metadata フォルダに保存されます。

Amazon Kendra メタデータの形式と構造の詳細については、S3 ドキュメントメタデータを参照してください。

トピック

- Amazon Comprehend の出力をダウンロードして抽出する
- S3 バケットに出力をアップロードする
- Amazon Kendra メタデータ形式への出力変換
- Amazon S3 バケットをクリーンアップする

Amazon Comprehend の出力をダウンロードして抽出する

Amazon Comprehend エンティティ分析出力を書式設定するには、まず Amazon Comprehend エンティティ分析 `output.tar.gz` アーカイブをダウンロードして、エンティティ分析ファイルを抽出する必要があります。

出力ファイルをダウンロードして抽出するには (コンソール)

1. Amazon Comprehend コンソールのナビゲーションペインで、[Analysis jobs] (分析ジョブ) に移動します。
2. エンティティ分析ジョブ `data-entities-analysis` を選択します。
3. [Output] (出力) で、[Output data location] (出力データの場所) の隣に表示されるリンクをクリックします。これにより、S3 バケットの `output.tar.gz` アーカイブにリダイレクトします。
4. [Overview] (概要) タブで、[Download] (ダウンロード) を選択します。

   > **Tip**
   >
   > すべての Amazon Comprehend 分析ジョブの出力は同じ名前になります。アーカイブの名前を変更すると、アーカイブの追跡が容易になります。

5. ダウンロードした Amazon Comprehend ファイルを解凍してデバイスに抽出します。

出力ファイルをダウンロードして抽出するには (AWS CLI)

1. エンティティ分析ジョブの結果を含む S3 バケット内の Amazon Comprehend 自動生成フォルダの名前にアクセスするには、[describe-entities-detection-job] コマンドを使用します。

Linux

```
aws comprehend describe-entities-detection-job \
          --job-id entities-job-id \
          --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、the section called “ステップ 2: エンティティを検出する” から保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws comprehend describe-entities-detection-job \
          --job-id entities-job-id \
          --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、the section called “ステップ 2: エンティティを検出する” から保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws comprehend describe-entities-detection-job ^
          --job-id entities-job-id ^
          --region aws-region
```

実行する条件は以下のとおりです。

- *entities-job-id* は、the section called “ステップ 2: エンティティを検出する” から保存した comprehend-job-id、
- *aws-region* は、AWS リージョンです。

2. エンティティのジョブの説明の OutputDataConfig オブジェクトから、テキストエディタで comprehend-S3uri としての S3Uri 値をコピーおよび保存します。

> **Note**
>
> S3Uri 値は、*s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz* のような形式になります。

3. エンティティ出力アーカイブをダウンロードするには、[copy] コマンドを使用します。

Linux

```
aws s3 cp s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz path/output.tar.gz
```

実行する条件は以下のとおりです。

- *s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz* は、comprehend-S3uri と名前を付けて保存した S3Uri 値、
- *path/* は、出力を保存するローカルディレクトリです。

macOS

```
aws s3 cp s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz path/output.tar.gz
```

実行する条件は以下のとおりです。

- *s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz* は、comprehend-S3uri と名前を付けて保存した S3Uri 値、
- *path/* は、出力を保存するローカルディレクトリです。

Windows

```
aws s3 cp s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz path/output.tar.gz
```

実行する条件は以下のとおりです。

- *s3://DOC-EXAMPLE-BUCKET/.../output/output.tar.gz* は、comprehend-S3uri と名前を付けて保存した S3Uri 値、
- *path/* は、出力を保存するローカルディレクトリです。

4. エンティティ出力を抽出するには、ターミナルウィンドウを開き、以下のコマンドを実行します。

Linux

```
tar -xf path/output.tar.gz -C path/
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上にダウンロードした output.tar.gz アーカイブへのファイルパスです。

macOS

```
tar -xf path/output.tar.gz -C path/
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上にダウンロードした output.tar.gz アーカイブへのファイルパスです。

Windows

```
tar -xf path/output.tar.gz -C path/
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上にダウンロードした output.tar.gz アーカイブへのファイルパスです。

このステップを完了すると、output というファイルと Amazon Comprehend 識別エンティティのリストがデバイス上に作成されます。

S3 バケットに出力をアップロードする

Amazon Comprehend エンティティ分析ファイルをダウンロードして抽出した後、抽出した output ファイルを Amazon S3 バケットにファイルへアップロードします。

抽出された Amazon Comprehend 出力ファイルをアップロードするには (コンソール)

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. [Buckets] (バケット) で、バケットの名前をクリックし、その後 [Upload] (アップロード) をクリックします。
3. [Files and folders] (ファイルとフォルダ) で、[Add files] (ファイルを追加) を選択します。
4. ダイアログボックスで、デバイスの抽出した output ファイルに移動して選択し、[Open] (開く) をクリックします。
5. [Destination] (送信先)、[Permissions] (アクセス許可)、および [Properties] (プロパティ) はデフォルト設定のままにしておきます。
6. [Upload] (アップロード) を選択します。

抽出された Amazon Comprehend 出力ファイルをアップロードするには (AWS CLI)

1. 抽出したoutput ファイルをバケットにアップロードするには、[copy] コマンドを使用します。

   Linux

   ```
   aws s3 cp path/output s3://DOC-EXAMPLE-BUCKET/output
   ```

   実行する条件は以下のとおりです。

   - *path/* は、抽出した output ファイルへのローカルファイルパス、
   - *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

   macOS

   ```
   aws s3 cp path/output s3://DOC-EXAMPLE-BUCKET/output
   ```

   実行する条件は以下のとおりです。

   - *path/* は、抽出した output ファイルへのローカルファイルパス、

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

Windows

```
aws s3 cp path/output s3://DOC-EXAMPLE-BUCKET/output
```

実行する条件は以下のとおりです。

- *path/* は、抽出した output ファイルへのローカルファイルパス、
- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

2. output ファイルが S3 バケットに正常にアップロードされたことを確認するには、[list] コマンドを使用してその内容をチェックします。

Linux

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

macOS

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

Windows

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

Amazon Kendra メタデータ形式への出力変換

Amazon Comprehend 出力を Amazon Kendra メタデータに変換するには、Python 3 スクリプトを実行します。コンソールを使用している場合は、このステップで AWS CloudShell を使用します。

Python 3 スクリプトを実行するには (コンソール)

1. デバイス上の converter.py.zip 圧縮ファイルをダウンロードします。
2. Python 3 ファイル `converter.py` を抽出します。
3. [AWS Management Console] (マネジメントコンソール) にサインインして AWS リージョンが S3 バケットと Amazon Comprehend 分析ジョブと同じリージョンに設定されていることを確認します。
4. [AWS CloudShell icon] (アイコン)を選択するか、上部のナビゲーションバーの [Search] (検索) ボックスに AWSCloudShell と入力して環境を起動します。

   > **Note**
   >
   > AWS CloudShell が新しいブラウザウィンドウで初めて起動すると、ウェルカムパネルが表示され、主要な機能が一覧表示されます。このパネルを閉じて、コマンドプロンプトが表示されると、シェルが対話できる状態になります。

5. ターミナルの準備が完了したら、ナビゲーションペインで [Actions] (アクション) を選択し、メニューから [Upload file] (ファイルをアップロードする) を選択します。
6. 開いたダイアログボックスで、[Select file] (ファイルを選択) をクリックし、お使いのデバイスからダウンロードした Python 3 ファイル `converter.py` を選択します。[Upload] (アップロード) を選択します。
7. AWS CloudShell 環境で、次のコマンドを入力します。

   ```
   python3 converter.py
   ```

8. シェルインターフェイスが [S3 バケットの名前を入力する] プロンプトを表示したら、S3 バケットの名前を入力し、[Enter] キーを押します。
9. シェルインターフェイスが [Enter the full filepath to your Comprehend output file] (Comprehend 出力ファイルへの完全なファイルパスを入力する) プロンプトを表示したら、**`output`** と入力し、[Enter] キーを押します。

10. シェルインターフェイスが [Enter the full filepath to your metadata folder] (メタデータフォルダへの完全なファイルパスを入力する) プロンプトを表示したら、**metadata/** と入力し、[Enter] キーを押します。

> **Important**
>
> メタデータを正しく書式設定するには、ステップ 8～10 の入力値が正確である必要があります。

Python 3 スクリプトを実行するには (AWS CLI)

1. Python 3 ファイル `converter.py` をダウンロードするには、ターミナルウィンドウを開き、以下のコマンドを実行します。

   Linux

   ```
   curl -o path/converter.py.zip https://docs.aws.amazon.com/kendra/latest/dg/
   samples/converter.py.zip
   ```

   実行する条件は以下のとおりです。

   - *path/* は、圧縮フォルダを保存する場所へのファイルパスです。

   macOS

   ```
   curl -o path/converter.py.zip https://docs.aws.amazon.com/kendra/latest/dg/
   samples/converter.py.zip
   ```

   実行する条件は以下のとおりです。

   - *path/* は、圧縮フォルダを保存する場所へのファイルパスです。

   Windows

   ```
   curl -o path/converter.py.zip https://docs.aws.amazon.com/kendra/latest/dg/
   samples/converter.py.zip
   ```

実行する条件は以下のとおりです。

- *path/* は、圧縮フォルダを保存する場所へのファイルパスです。

2. Python 3 ファイル を抽出するには、ターミナルウィンドウを開き、以下のコマンドを実行します。

Linux

```
unzip path/converter.py.zip -d path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した converter.py.zip へのファイルパスです。

macOS

```
unzip path/converter.py.zip -d path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した converter.py.zip へのファイルパスです。

Windows

```
tar -xf path/converter.py.zip -C path/
```

実行する条件は以下のとおりです。

- *path/* は、保存した converter.py.zip へのファイルパスです。

3. 次のコマンドを実行して、Boto3 がお使いのデバイスにインストールされていることを確認します。

Linux

```
pip3 show boto3
```

macOS

```
pip3 show boto3
```

Windows

```
pip3 show boto3
```

> **Note**
>
> Boto3 がインストールされていない場合は、`pip3 install boto3` を実行してインストールしてください。

4. Python 3 スクリプトを実行して `output` ファイルを変換し、次のコマンドを実行します。

Linux

```
python path/converter.py
```

実行する条件は以下のとおりです。

- *path/* は、保存した `converter.py.zip` へのファイルパスです。

macOS

```
python path/converter.py
```

実行する条件は以下のとおりです。

- *path/* は、保存した `converter.py.zip` へのファイルパスです。

Windows

```
python path/converter.py
```

実行する条件は以下のとおりです。

- *path/* は、保存した converter.py.zip へのファイルパスです。

5. AWS CLI が Enter the name of your S3 bucket プロンプトを表示した場合、S3 バケットの名前を入力し、[Enter] キーを押します。
6. AWS CLI が Enter the full filepath to your Comprehend output file プロンプトを表示した場合、**output** と入力し、[Enter] キーを押します。
7. AWS CLI が Enter the full filepath to your metadata folder プロンプトを表示した場合、**metadata/** と入力し、[Enter] キーを押します。

> **Important**
>
> メタデータを正しく書式設定するには、ステップ 5~7 の入力値が正確である必要があります。

このステップが完了すると、書式設定されたメタデータは S3 バケット内の metadata フォルダ内に置かれます。

Amazon S3 バケットをクリーンアップする

Amazon Kendra インデックスはバケットに保存されているすべてのファイルを同期するため、検索結果の重複を防ぐため、Amazon S3 バケットをクリーンアップすることをお勧めします。

Amazon S3 バケットをクリックアップするには (コンソール)

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. [Buckets] (バケット) で、バケットを選択し、Amazon Comprehend エンティティ分析出力フォルダ、Amazon Comprehend エンティティ分析 .temp ファイル、および抽出された Amazon Comprehend output ファイルを選択します。
3. [Overview] (概要) タブから [Delete] (削除) を選択します。
4. [Delete objects] (オブジェクトの削除) で、[Permanently delete objects?] (オブジェクトを完全に削除しますか) を選択し、テキスト入力フィールドに **permanently delete** を入力します。
5. [Delete objects] (オブジェクトの削除) を選択します。

Amazon S3 バケットをクリーンアップするには (AWS CLI)

1. S3 バケット内の data ファイルと metadata フォルダをすべて削除するには、AWS CLI で [remove] コマンドを使用します。

   Linux

   ```
   aws s3 rm s3://DOC-EXAMPLE-BUCKET/ --recursive --exclude "data/*" --exclude
    "metadata/*"
   ```

   実行する条件は以下のとおりです。

   - *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

   macOS

   ```
   aws s3 rm s3://DOC-EXAMPLE-BUCKET/ --recursive --exclude "data/*" --exclude
    "metadata/*"
   ```

   実行する条件は以下のとおりです。

   - *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

   Windows

   ```
   aws s3 rm s3://DOC-EXAMPLE-BUCKET/ --recursive --exclude "data/*" --exclude
    "metadata/*"
   ```

   実行する条件は以下のとおりです。

   - *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

2. オブジェクトが S3 バケットから正常に削除されたことを確認するには、[list] コマンドを使用してその内容をチェックします。

   Linux

   ```
   aws s3 ls s3://DOC-EXAMPLE-BUCKET/
   ```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

macOS

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

Windows

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET/
```

実行する条件は以下のとおりです。

- *DOC-EXAMPLE-BUCKET* は、S3 バケットの名前です。

このステップが完了すると、Amazon Comprehend エンティティ分析出力が Amazon Kendra メタデータに変換されます。これで、Amazon Kendra インデックスを作成する準備ができました。

ステップ 4: Amazon Kendra インデックスを作成し、メタデータを取り込む

インテリジェント検索ソリューションを実装するには、Amazon Kendra インデックスを作成し、S3 データとメタデータをそこに取り込みます。

Amazon Kendra インデックスにメタデータを追加する前に、カスタムドキュメント属性に対応するカスタムインデックスフィールドを作成します。これは、Amazon Comprehend エンティティタイプに対応します。Amazon Kendra では、作成したインデックスフィールドとカスタムドキュメント属性を使用して、ドキュメントを検索およびフィルタリングします。

詳細については、インデックスおよびカスタムドキュメント属性の作成を参照してください。

トピック

- Amazon Kendra インデックスの作成
- Amazon S3 アクセスのための IAM ロールの更新
- Amazon Kendra カスタム検索インデックスフィールドを作成する
- Amazon S3 バケットをインデックスのデータソースとして追加する
- Amazon Kendra インデックスの同期

Amazon Kendra インデックスの作成

ソースドキュメントをクエリするには、Amazon Kendra インデックスを作成します。

このステップで AWS CLI を使用している場合、インデックスを作成する前に Amazon Kendra の CloudWatch ログへのアクセスを許可する AWS IAM ロールとポリシーを作成しアタッチします。詳細については、前提条件を参照してください。

Amazon Kendra インデックスを作成するには (コンソール)

1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。

   > **Important**
   >
   > Amazon Comprehend エンティティジョブと Amazon S3 バケットを作成したリージョンと同じリージョンに存在することを確認します。別のリージョンにいる場合は、トップナビゲーションバーの [Region selector] (リージョンセレクタ) から Amazon S3 バケットを作成した AWS リージョンを選択します。

2. [Create an index] (インデックスの作成) を選択します。
3. [Index details] (インデックスの詳細の指定) ページの [Specify index details] (インデックスの詳細) で、次の操作を行います。

   a. [Index name] (インデックス名) に **kendra-index** と入力します。
   b. [Description] (説明) フィールドは空白のままにしておきます。
   c. [IAM role] (IAM ロール) は、[Create a new role] (新しいロールの作成) を選択します。このロールは、Amazon S3 バケットへのアクセスを提供します。
   d. [Role name] (ロール名) に **kendra-role** と入力します。IAM ロールにはプレフィックス `AmazonKendra-` が付いています。
   e. [Encryption] (暗号化) と [Tags] (タグ) のデフォルト設定はそのままにして、[Next] (次へ) をクリックします。

4. [Access control setting] (アクセスコントロールの設定) ページの [Configure user access control] (アクセスコントロールの設定) で、[No] (いいえ) を選択してから、[Next] (次へ) をクリックします。
5. [Provisioning editions] (プロビジョニングの詳細) ページの [Provisioning details] (プロビジョニングエディション) で、[Developer edition] (デベロッパーエディション) を選択し、[Create] (作成) をクリックします。

Amazon Kendra インデックスを作成するには (AWS CLI)

1. 信頼されたエンティティとして認識する Amazon Kendra の IAM ロールを作成してアタッチするには、以下の操作を行います。

   a. 次の信頼ポリシーを、ローカルデバイスのテキストエディタで `kendra-trust-policy.json` という JSON ファイルとして保存します。

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

   b. `kendra-role` という IAM ロールを作成するには、保存した `kendra-trust-policy.json` ファイルをアタッチしてくださいそれをファイルして、[create-role] コマンドを使用します。

   Linux

   ```
   aws iam create-role \
            --role-name kendra-role \
            --assume-role-policy-document file://path/kendra-trust-policy.json
   ```

   実行する条件は以下のとおりです。

   - *path/* は、ローカルデバイス上の `kendra-trust-policy.json` フォルダへのファイルパスです。

macOS

```
aws iam create-role \
          --role-name kendra-role \
          --assume-role-policy-document file://path/kendra-trust-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-trust-policy.json フォルダへのファイルパスです。

Windows

```
aws iam create-role ^
          --role-name kendra-role ^
          --assume-role-policy-document file://path/kendra-trust-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-trust-policy.json フォルダへのファイルパスです。

c. Amazon リソースネーム (ARN) をテキストエディタにコピーし、kendra-role-arn としローカルに保存します。

> **Note**
>
> ARN は、*arn:aws:iam::123456789012:role/kendra-role* というような形式になります。Amazon Kendra ジョブを実行するには、kendra-role-arn として保存した ARN が必要になります。

2. インデックスを作成する前に、kendra-role に CloudWatch Logs への書き込み許可を提供する必要があります。そのためには、以下のステップを完了します。

a. 次の信頼ポリシーを、ローカルデバイスのテキストエディタで kendra-cloudwatch-policy.json という JSON ファイルとして保存します。

```
{
```

```
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"cloudwatch:PutMetricData",
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "cloudwatch:namespace":"Kendra"
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":"logs:DescribeLogGroups",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"logs:CreateLogGroup",
         "Resource":"arn:aws:logs:aws-region:aws-account-id:log-group:/aws/
kendra/*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "logs:DescribeLogStreams",
            "logs:CreateLogStream",
            "logs:PutLogEvents"
         ],
         "Resource":"arn:aws:logs:aws-region:aws-account-id:log-group:/aws/
kendra/*:log-stream:*"
      }
   ]
}
```

aws-region をお客様の AWS リージョンと、*aws-account-id* を 12 桁の AWS アカウント ID と置き換えます。

b. CloudWatch Logs にアクセスするための IAM ポリシーを作成するには、[create-policy] コマンドを使用します。

Linux

```
aws iam create-policy \
          --policy-name kendra-cloudwatch-policy \
          --policy-document file://path/kendra-cloudwatch-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-cloudwatch-policy.json フォルダへのファイルパスです。

macOS

```
aws iam create-policy \
          --policy-name kendra-cloudwatch-policy \
          --policy-document file://path/kendra-cloudwatch-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-cloudwatch-policy.json フォルダへのファイルパスです。

Windows

```
aws iam create-policy ^
          --policy-name kendra-cloudwatch-policy ^
          --policy-document file://path/kendra-cloudwatch-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-cloudwatch-policy.json フォルダへのファイルパスです。

c. Amazon リソースネーム (ARN) をテキストエディタにコピーし、kendra-cloudwatch-arn としローカルに保存します。

> **Note**
>
> ARN は、*arn:aws:iam::123456789012:role/kendra-cloudwatch-policy* というような形式になります。kendra-cloudwatch-arn を IAM ロールにアタッチするには、kendra-cloudwatch-policy として保存した ARN が必要になります。

d. kendra-cloudwatch-policy を IAM ロールにアタッチするには、[attach-role-policy] コマンドを使用します。

Linux

```
aws iam attach-role-policy \
          --policy-arn policy-arn \
          --role-name kendra-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、保存した kendra-cloudwatch-arn です。

macOS

```
aws iam attach-role-policy \
          --policy-arn policy-arn \
          --role-name kendra-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、保存した kendra-cloudwatch-arn です。

Windows

```
aws iam attach-role-policy ^
          --policy-arn policy-arn ^
          --role-name kendra-role
```

実行する条件は以下のとおりです。

- *policy-arn* は、保存した kendra-cloudwatch-arn です。

3. インデックスを作成するには、[create-index] コマンドを使用します。

Linux

```
aws kendra create-index \
        --name kendra-index \
        --edition DEVELOPER_EDITION \
        --role-arn role-arn \
        --region aws-region
```

実行する条件は以下のとおりです。

- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra create-index \
        --name kendra-index \
        --edition DEVELOPER_EDITION \
        --role-arn role-arn \
        --region aws-region
```

実行する条件は以下のとおりです。

- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra create-index ^
        --name kendra-index ^
        --edition DEVELOPER_EDITION ^
        --role-arn role-arn ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

4. テキストエディタでインデックス Id をコピーし、kendra-index-id という名前を付けて保存します。Id は、インデックス作成のステータスを追跡するのに役立ちます。
5. インデックス作成ジョブの進行状況を追跡するには、[describe-index] コマンドを使用します。

Linux

```
aws kendra describe-index \
        --id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra describe-index \
        --id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra describe-index ^
        --id kendra-index-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

インデックスの作成プロセスには平均で 15 分かかりますが、さらに時間がかかる場合があります。インデックスのステータスがアクティブになると、インデックスが使用可能になります。インデックスの作成中に、次のステップを開始できます。

このステップで AWS CLI を使用している場合は、S3 バケットへのアクセス許可をインデックスに付与する IAM ポリシーを作成し、Amazon Kendra IAM ロールにアタッチします。

Amazon S3 アクセスのための IAM ロールの更新

インデックスの作成中に、Amazon Kendra IAM ロールを更新して、作成したインデックスが Amazon S3 バケットからデータを読み取ることを許可します。詳細については、Amazon Kendra の IAM アクセスロールを参照してください。

IAM ロールを更新するには (コンソール)

1. IAM コンソール (https://console.aws.amazon.com/iam/) を開きます。
2. 左側のナビゲーションペインで、[Roles] (ロール) を選択し、[Role name] (ロール名) の上の [Search] (検索) ボックスに **kendra-role** と入力します。
3. 提案されるオプションで、kendra-role をクリックします。
4. [Summary] (概要) で、[Attach policies] (ポリシーの添付) を選択します。
5. [Attach permissions] (許可の添付) の [Search] (検索) ボックスで、**S3** と入力し、提案されたオプションで、[AmazonS3ReadOnlyAccess] ポリシーの隣にあるチェックボックスをオンにします。
6. [Attach policy] (ポリシーの添付) を選択します。[Summary] (概要) ページで、IAM ロールに添付された 2 つのポリシーが表示されます。
7. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) に戻り、インデックスのステータスが [Creating] (作成中) から [Active] (アクティブ) に変わるのを待って次のステップに進みます。

IAM ロールを更新するには (AWS CLI)

1. 次のテキストを、ローカルデバイスのテキストエディタで kendra-S3-access-policy.json という JSON ファイルに保存します。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Action":[
            "s3:GetObject"
         ],
         "Resource":[
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
         ],
         "Effect":"Allow"
      },
      {
         "Action":[
            "s3:ListBucket"
         ],
         "Resource":[
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
         ],
         "Effect":"Allow"
      },
      {
         "Effect":"Allow",
         "Action":[
            "kendra:BatchPutDocument",
            "kendra:BatchDeleteDocument",
            "kendra:ListDataSourceSyncJobs"
         ],
         "Resource":[
            "arn:aws:kendra:aws-region:aws-account-id:index/kendra-index-id"
         ]
      }
   ]
}
```

DOC-EXAMPLE-BUCKET を S3 バケット名に、*aws-region* を AWS リージョンに、*aws-account-id* を 12 桁の AWS アカウント ID に、*kendra-index-id* を保存した kendra-index-id に置き換えます。

2. S3 バケットにアクセスする IAM ポリシーを作成するには、[create-policy] コマンドを使用します。

Linux

```
aws iam create-policy \
          --policy-name kendra-S3-access-policy \
          --policy-document file://path/kendra-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-S3-access-policy.json フォルダへのファイルパスです。

macOS

```
aws iam create-policy \
          --policy-name kendra-S3-access-policy \
          --policy-document file://path/kendra-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-S3-access-policy.json フォルダへのファイルパスです。

Windows

```
aws iam create-policy ^
          --policy-name kendra-S3-access-policy ^
          --policy-document file://path/kendra-S3-access-policy.json
```

実行する条件は以下のとおりです。

- *path/* は、ローカルデバイス上の kendra-S3-access-policy.json フォルダへのファイルパスです。

3. Amazon リソースネーム (ARN) をテキストエディタにコピーし、kendra-S3-access-arn としローカルに保存します。

> **Note**
>
> ARN は、*arn:aws:iam::123456789012:role/kendra-S3-access-policy* というような形式になります。kendra-S3-access-policy を IAM ロールにアタッチするには、kendra-S3-access-arn として保存した ARN が必要になります。

4. kendra-S3-access-policy を Amazon Kendra IAM ロールにアタッチするには、[attach-role-policy] コマンドを使用します。

   Linux

   ```
   aws iam attach-role-policy \
            --policy-arn policy-arn \
            --role-name kendra-role
   ```

   実行する条件は以下のとおりです。

   - *policy-arn* は、保存した kendra-S3-access-arn です。

   macOS

   ```
   aws iam attach-role-policy \
            --policy-arn policy-arn \
            --role-name kendra-role
   ```

   実行する条件は以下のとおりです。

   - *policy-arn* は、保存した kendra-S3-access-arn です。

   Windows

   ```
   aws iam attach-role-policy ^
            --policy-arn policy-arn ^
            --role-name kendra-role
   ```

   実行する条件は以下のとおりです。

   - *policy-arn* は、保存した kendra-S3-access-arn です。

Amazon Kendra カスタム検索インデックスフィールドを作成する

メタデータをカスタムドキュメント属性として認識するように Amazon Kendra を準備するには、Amazon Comprehend エンティティタイプに対応するカスタムフィールドを作成します。次の 9 つの Amazon Comprehend エンティティタイプをカスタムフィールドとして入力します。

- COMMERCIAL_ITEM
- DATE
- EVENT
- LOCATION
- ORGANIZATION
- OTHER
- PERSON
- QUANTITY
- TITLE

> **Important**
>
> スペルミスのあるエンティティタイプは、インデックスによって認識されません。

Amazon Kendra インデックスのカスタムフィールドを作成するには (コンソール)

1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. [Indexes] (インデックス) リストから、`kendra-index` をクリックします。
3. 左側のナビゲーションパネルの [Data management] (データ管理) で、[Facet definition] (ファセット定義) を選択します。
4. [Index fields] (インデックスフィールド) メニューで、[Add field] (フィールドを追加) を選択します。
5. [Add index field] (インデックスフィールドの追加) ダイアログボックスで、以下の操作を行います。

    a. [Field name] (フィールド名) に **`COMMERCIAL_ITEM`** と入力します。

    b. [Data type] (データタイプ) で、[String list] (文字列リスト) を選択します。

c. [Usage type] (使用タイプ) で、[Facetable] (ファセット可能)、[Searchable] (検索可能)、および [Displayable] (表示可能)、を選択して [Add] (追加) を選択します。

d. Amazon Comprehend の各エンティティタイプ (COMMERCIAL_ITEM、DATE、EVENT、LOCATION、ORGANIZATION、OTHER、PERSON、Q について、ステップ a から c を繰り返します。

コンソールに成功したフィールド追加メッセージが表示されます。次のステップに進む前に、それらを閉じることができます。

Amazon Kendra インデックスのカスタムフィールドを作成するには (AWS CLI)

1. 次のテキストを、ローカルデバイスのテキストエディタで `custom-attributes.json` という JSON ファイルとして保存します。

```
[
   {
       "Name": "COMMERCIAL_ITEM",
       "Type": "STRING_LIST_VALUE",
       "Search": {
           "Facetable": true,
           "Searchable": true,
           "Displayable": true
       }
   },
   {
       "Name": "DATE",
       "Type": "STRING_LIST_VALUE",
       "Search": {
           "Facetable": true,
           "Searchable": true,
           "Displayable": true
       }
   },
   {
       "Name": "EVENT",
       "Type": "STRING_LIST_VALUE",
       "Search": {
           "Facetable": true,
           "Searchable": true,
           "Displayable": true
       }
```

```
    },
    {
        "Name": "LOCATION",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
        }
    },
    {
        "Name": "ORGANIZATION",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
        }
    },
    {
        "Name": "OTHER",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
        }
    },
    {
        "Name": "PERSON",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
        }
    },
    {
        "Name": "QUANTITY",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
```

```
        }
    },
    {
        "Name": "TITLE",
        "Type": "STRING_LIST_VALUE",
        "Search": {
            "Facetable": true,
            "Searchable": true,
            "Displayable": true
        }
    }
]
```

2. インデックスにカスタムフィールドを作成するには、[update-index] コマンドを使用します。

Linux

```
aws kendra update-index \
        --id kendra-index-id \
        --document-metadata-configuration-updates file://path/custom-
attributes.json \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *path/* は、ローカルデバイス上の custom-attributes.json フォルダへのファイルパス、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra update-index \
        --id kendra-index-id \
        --document-metadata-configuration-updates file://path/custom-
attributes.json \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *path/* は、ローカルデバイス上の custom-attributes.json フォルダへのファイルパス、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra update-index ^
        --id kendra-index-id ^
        --document-metadata-configuration-updates file://path/custom-
attributes.json ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *path/* は、ローカルデバイス上の custom-attributes.json フォルダへのファイルパス、
- *aws-region* は、AWS リージョンです。

3. カスタム属性がインデックスに追加されていることを確認するには、[describe-index] コマンドを使用します。

Linux

```
aws kendra describe-index \
        --id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra describe-index \
```

```
        --id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra describe-index ^
        --id kendra-index-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Amazon S3 バケットをインデックスのデータソースとして追加する

インデックスを同期する前に、S3 データソースをそのインデックスに接続する必要があります。

S3 バケットを Amazon Kendra インデックスに接続するには (コンソール)

1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. [Indexes] (インデックス) リストから、kendra-index をクリックします。
3. 左側のナビゲーションメニューから、[Data management] (データ管理) で、[Data sources] (データソース) を選択します。
4. [Select data source connector type] (データソースコネクタのタイプを選択する) セクションで [Amazon S3] に移動し、[Add connector] (コネクタの追加) を選択します。
5. [Specify data source details] (データソースの詳細の指定) ページで、以下の操作を行います。
    a. [Name and description] (名前と説明) の [Data source name] (データソース名) に、**S3-data-source** と入力します。
    b. [Description] (説明) セクションは空白のままにしておきます。
    c. [Tags] (タグ) は、デフォルト設定のままにしておきます。

d. [Next] (次へ) を選択します。

6. [Configure sync settings] (設定の定義) ページの [Sync scope] (同期の適用範囲) セクションで、以下の操作を行います。

   a. [Enter the data source location] (データソースの場所を入力する) で、[Browse S3] (S3 を閲覧する)を選択します。

   b. [Choose resources] (リソースの選択) で、S3 バケット、[Choose] (選択) の順に選択します。

   c. [Metadata files prefix folder location] (メタデータファイルのプレフィックスフォルダの場所) で、[Browse S3] (S3 を閲覧する) を選択します。

   d. [Choose resources] (リソースの選択) で、バケットのリストからバケットの名前をクリックします。

   e. [Objects] (オブジェクト) で、`metadata` のオプションボックスを選択し、[Choose] (選択) をクリックします。ロケーションフィールドには、`metadata/` と表示されます。

   f. [Access control list configuration file location] (アクセスコントロールリスト設定ファイルの場所)、[Select decryption key] (復号キーを選択)、および [Additional configuration] (追加設定) は、デフォルト設定のままにしておきます。

7. [Configure sync settings] (同期設定を構成する) ページの [IAM role] (IAM ロール) で、[`kendra-role`] を選択します。

8. [Configure sync settings] (同期設定を構成する) ページの [Sync run schedule] (同期実行スケジュール) で、[Frequency] (頻度) に [Run on demand] (オンデマンドで実行する) を選択して [Next] (次へ) をクリックします。

9. [Review and create] (確認と作成) ページで、データソースの詳細について選択内容を確認し、[Add data source] (データソースの追加) を選択します。

S3 バケットを Amazon Kendra インデックスに接続するには (AWS CLI)

1. 次のテキストを、ローカルデバイスのテキストエディタで `S3-data-connector.json` という JSON ファイルとして保存します。

```
{
   "S3Configuration":{
      "BucketName":"DOC-EXAMPLE-BUCKET",
      "DocumentsMetadataConfiguration":{
         "S3Prefix":"metadata"
      }
```

```
    }
}
```

DOC-EXAMPLE-BUCKET を S3 バケットの名前に置き換えます。

2. S3 バケットをインデックスに接続するには、[create-data-source] コマンドを使用します。

Linux

```
aws kendra create-data-source \
        --index-id kendra-index-id \
        --name S3-data-source \
        --type S3 \
        --configuration file://path/S3-data-connector.json \
        --role-arn role-arn \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *path/* は、ローカルデバイス上の S3-data-connector.json フォルダへのファイルパス、
- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra create-data-source \
        --index-id kendra-index-id \
        --name S3-data-source \
        --type S3 \
        --configuration file://path/S3-data-connector.json \
        --role-arn role-arn \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、

- *path/* は、ローカルデバイス上の S3-data-connector.json フォルダへのファイルパス、
- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra create-data-source ^
        --index-id kendra-index-id ^
        --name S3-data-source ^
        --type S3 ^
        --configuration file://path/S3-data-connector.json ^
        --role-arn role-arn ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *path/* は、ローカルデバイス上の S3-data-connector.json フォルダへのファイルパス、
- *role-arn* は、保存した kendra-role-arn、
- *aws-region* は、AWS リージョンです。

3. テキストエディタでコネクタ Id をコピーし、S3-connector-id という名前を付けて保存します。Id は、データ接続プロセスのステータスを追跡するのに役立ちます。
4. S3 データソースが正常に接続されていることを確認するには、[describe-data-source] コマンドを使用します。

   Linux

   ```
   aws kendra describe-data-source \
           --id S3-connector-id \
           --index-id kendra-index-id \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *S3-connector-id* は、保存した S3-connector-id、

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra describe-data-source \
        --id S3-connector-id \
        --index-id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra describe-data-source ^
        --id S3-connector-id ^
        --index-id kendra-index-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

このステップを終了すると、Amazon S3 データソースがインデックスに接続されます。

Amazon Kendra インデックスの同期

Amazon S3 データソースを追加すると、Amazon Kendra インデックスが同期されます。

Amazon Kendra インデックスを同期するには (コンソール)

1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. [Indexes] (インデックス) リストから、kendra-index をクリックします。
3. 左側のナビゲーションメニューから [Data sources] (データソース) を選択します。
4. [Data sources] (データソース) から、S3-data-source を選択します。
5. 上部のナビゲーションバーから、[Sync now] (今すぐ同期) を選択します。

Amazon Kendra インデックスを同期するには (AWS CLI)

1. インデックスを同期するには、[start-data-source-sync-job] コマンドを使用します。

   Linux

   ```
   aws kendra start-data-source-sync-job \
           --id S3-connector-id \
           --index-id kendra-index-id \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *S3-connector-id* は、保存した S3-connector-id、
   - *kendra-index-id* は、保存した kendra-index-id、
   - *aws-region* は、AWS リージョンです。

   macOS

   ```
   aws kendra start-data-source-sync-job \
           --id S3-connector-id \
           --index-id kendra-index-id \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *S3-connector-id* は、保存した S3-connector-id、
   - *kendra-index-id* は、保存した kendra-index-id、
   - *aws-region* は、AWS リージョンです。

Windows

```
aws kendra start-data-source-sync-job ^
        --id S3-connector-id ^
        --index-id kendra-index-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

2. インデックス同期のステータスを確認するには、[list-data-source-sync-jobs] コマンドを使用します。

Linux

```
aws kendra list-data-source-sync-jobs \
        --id S3-connector-id \
        --index-id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra list-data-source-sync-jobs \
        --id S3-connector-id \
        --index-id kendra-index-id \
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra list-data-source-sync-jobs ^
        --id S3-connector-id ^
        --index-id kendra-index-id ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *S3-connector-id* は、保存した S3-connector-id、
- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

このステップを完了すると、データセットに検索可能でフィルター可能な Amazon Kendra インデックスが作成されます。

ステップ 5: Amazon Kendra インデックスをクエリする

Amazon Kendra インデックスは自然言語クエリの準備が整いました。インデックスを検索すると、Amazon Kendra は指定したすべてのデータとメタデータを使用して、検索クエリに対する最も正確な回答を返します。

Amazon Kendra が回答できるクエリには、次の 3 種類があります。

- Factoid 型クエリ (「誰が」、「何を」、「いつ」、または「どこで」の質問)
- 説明的なクエリ (「どのように」の質問)
- キーワード検索 (意図と対象範囲が明確でない質問)

トピック

- Amazon Kendra インデックスをクエリする
- 検索結果のフィルタリング

Amazon Kendra インデックスをクエリする

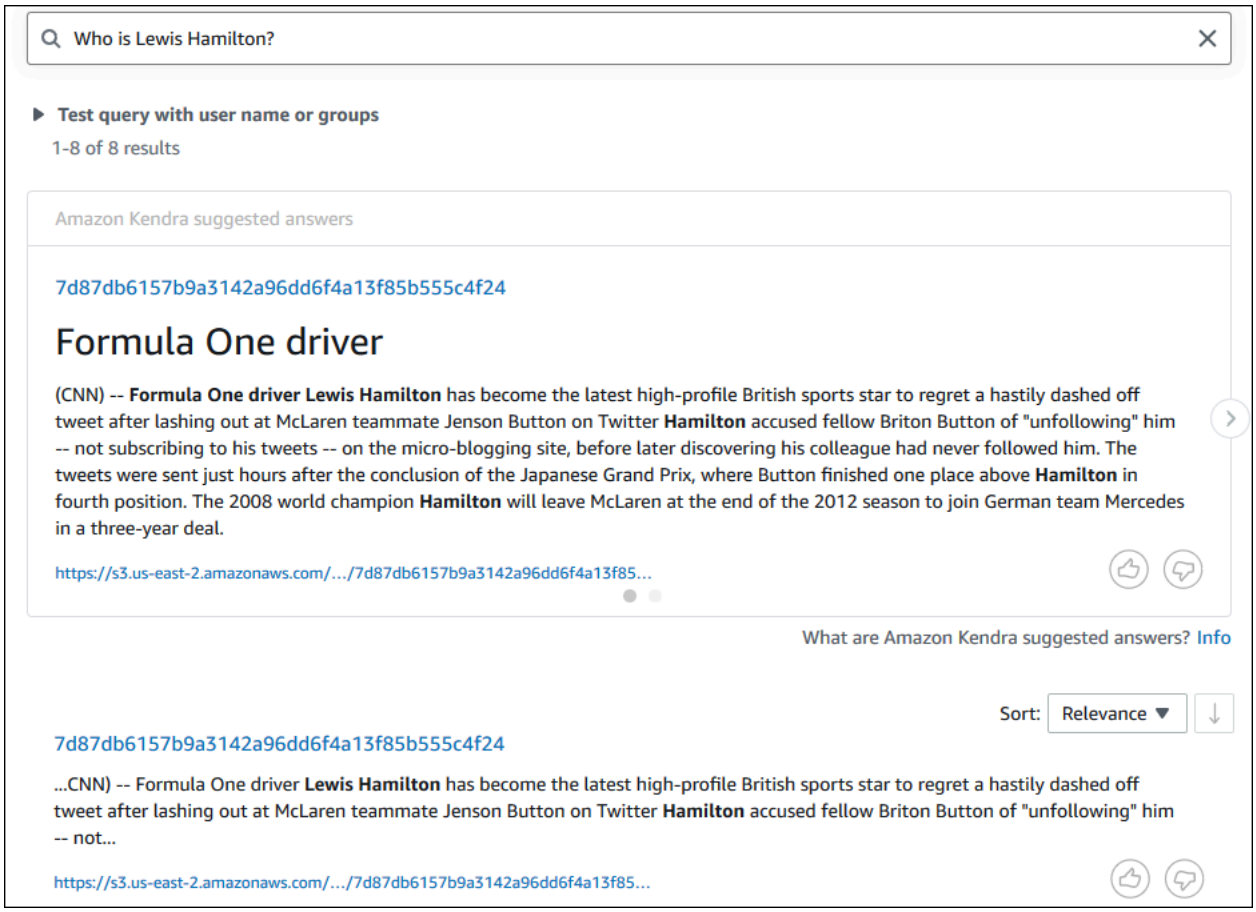
Amazon Kendra がサポートする 3 種類のクエリに対応する質問を使用して、Amazon Kendra インデックスをクエリできます。詳細については、クエリを参照してください。

このセクションの質問例は、サンプルデータセットに基づいて選択されています。

Amazon Kendra インデックスをクエリするには (コンソール)

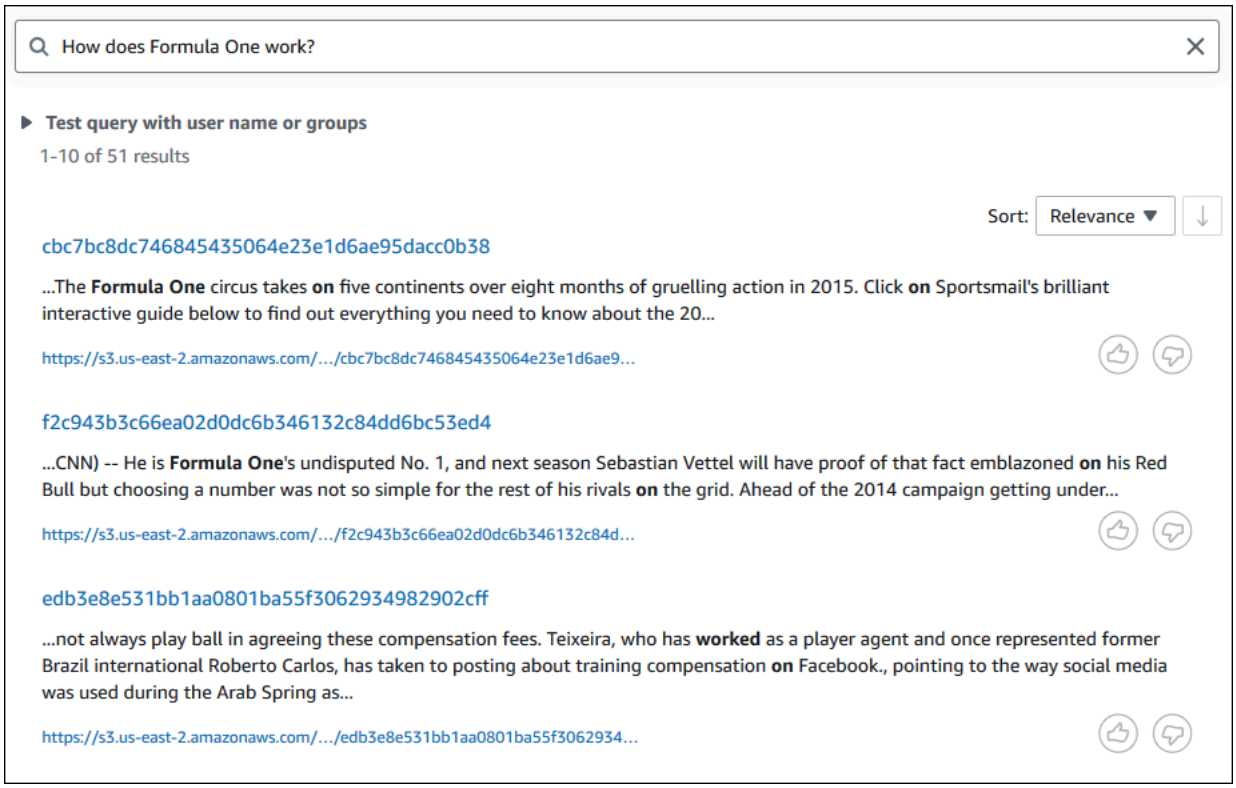
1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. [Indexes] (インデックス) リストから、`kendra-index` をクリックします。
3. 左側のナビゲーションメニューから、インデックスを検索するオプションを選択します。
4. Factoid 型クエリを実行するには、検索ボックスに **`Who is Lewis Hamilton?`** と入力し、[Enter] キーを押します。

   最初に返された結果は、Amazon Kendra が提案した回答と、その回答を含むデータファイルです。残りの結果は、一連の推奨ドキュメントを形成します。



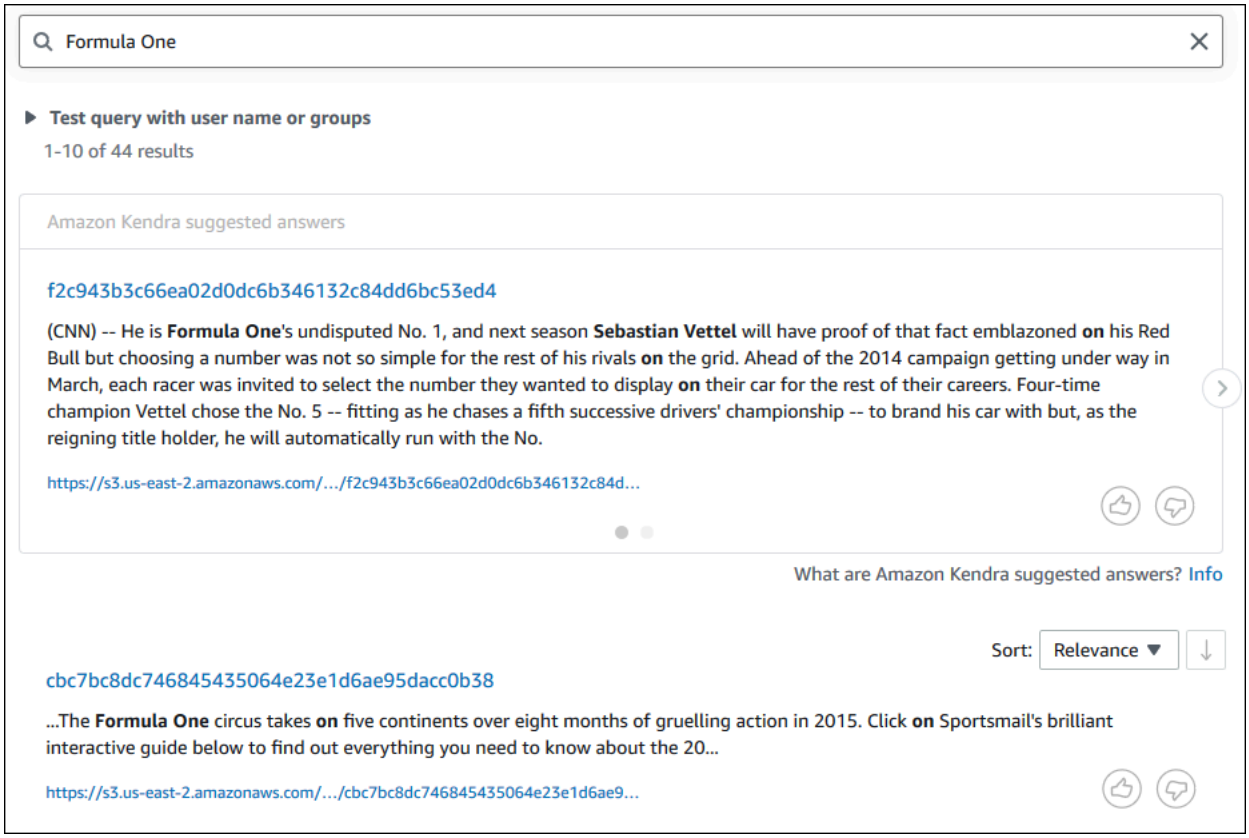
5. 説明的なクエリを実行するには、検索ボックスに **`How does Formula One work?`** と入力し、[Enter] キーを押します。

   Amazon Kendra コンソールから返された別の結果が表示されます。今回は、関連する語句が強調表示されます。



6. キーワード検索を実行するには、検索ボックスに **`Formula One`** と入力し、[Enter] キーを押します。

   Amazon Kendra コンソールから返される別の結果と、データセット内の語句に関する他のすべてのメンションの結果が表示されます。



Amazon Kendra インデックスをクエリするには (AWS CLI)

1. サンプルの Factoid 型クエリを実行するには、[query] コマンドを使用します。

   Linux

   ```
   aws kendra query \
           --index-id kendra-index-id \
           --query-text "Who is Lewis Hamilton?" \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *kendra-index-id* は、保存した kendra-index-id、
   - *aws-region* は、AWS リージョンです。

macOS

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "Who is Lewis Hamilton?" \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra query ^
        --index-id kendra-index-id ^
        --query-text "Who is Lewis Hamilton?" ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

クエリの結果が AWS CLI に表示されます。

2. サンプルの説明的なクエリを実行するには、[query] コマンドを使用します。

Linux

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "How does Formula One work?" \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、

- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "How does Formula One work?" \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra query ^
        --index-id kendra-index-id ^
        --query-text "How does Formula One work?" ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

クエリに対する結果が AWS CLI に表示されます。

3. サンプルキーワード検索を実行するには、[query] コマンドを使用します。

Linux

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "Formula One" \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "Formula One" \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra query ^
        --index-id kendra-index-id ^
        --query-text "Formula One" ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

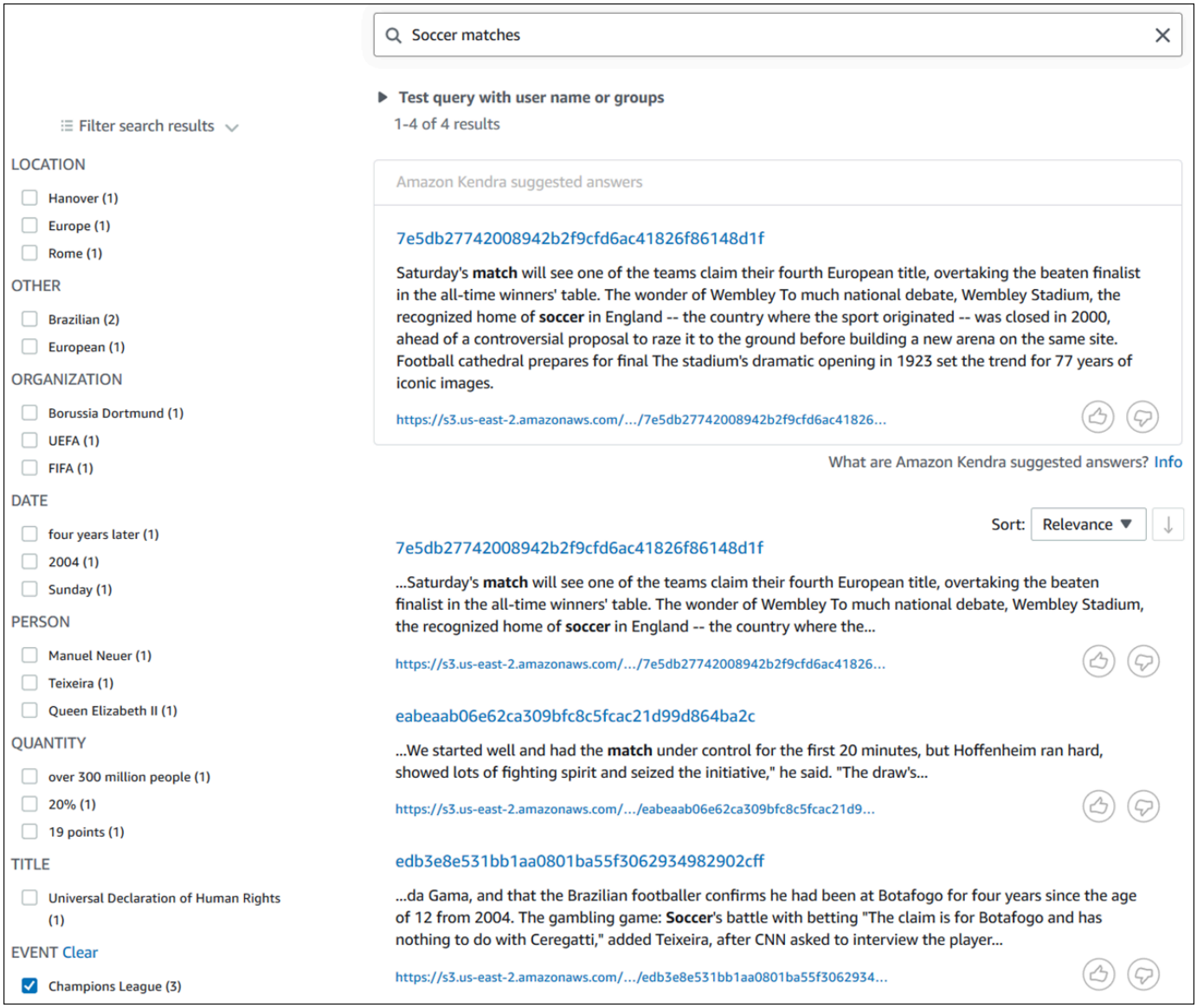
クエリに対して返された回答が AWS CLI に表示されます。

検索結果のフィルタリング

Amazon Kendra コンソールのカスタムドキュメント属性を使用して、検索結果をフィルタリングしてソートすることができます。Amazon Kendra がクエリを処理する方法の詳細については、クエリのフィルタリングを参照してください。

検索結果をフィルタリングするには (コンソール)

1. Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/) を開きます。
2. [Indexes] (インデックス) リストから、`kendra-index` をクリックします。
3. 左側のナビゲーションメニューから、インデックスを検索するオプションを選択します。
4. 検索ボックスに、クエリとして **`Soccer matches`** と入力し、[Enter] キーを押します。
5. 左側のナビゲーションメニューから、[Filter search results] (検索結果をフィルタリング) を選択し、検索結果をフィルタリングするために使用できるファセットのリストを表示します。
6. [EVENT] の小見出しの「Champions League」のチェックボックスをオンにし、「Champions League」を含む結果のみにフィルタリングされた検索結果を表示します。



検索結果をフィルタリングするには (AWS CLI)

1. 検索可能な特定のタイプのエンティティ (EVENT など) を表示するには、[query] コマンドを使用します。

   Linux

   ```
   aws kendra query \
           --index-id kendra-index-id \
           --query-text "Soccer matches" \
           --facets '[{"DocumentAttributeKey":"EVENT"}]' \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *kendra-index-id* は、保存した kendra-index-id、
   - *aws-region* は、AWS リージョンです。

   macOS

   ```
   aws kendra query \
           --index-id kendra-index-id \
           --query-text "Soccer matches" \
           --facets '[{"DocumentAttributeKey":"EVENT"}]' \
           --region aws-region
   ```

   実行する条件は以下のとおりです。

   - *kendra-index-id* は、保存した kendra-index-id、
   - *aws-region* は、AWS リージョンです。

   Windows

   ```
   aws kendra query ^
           --index-id kendra-index-id ^
           --query-text "Soccer matches" ^
           --facets '[{"DocumentAttributeKey":"EVENT"}]' ^
           --region aws-region
   ```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

検索結果が AWS CLI に表示されます。タイプ EVENT のファセットのリストを取得するには、AWS CLI アウトプットの「FacetResults」セクションに移動し、フィルタリング可能なファセットとその数の一覧を表示します。例えば、ファセットの 1 つは「Champions League」です。

> **Note**
>
> EVENT の代わりに、DocumentAttributeKey 値に対して the section called “Amazon Kendra インデックスの作成” で作成したインデックスフィールドを選択できます。

2. 同じ検索を実行し、「Champions League」を含む結果のみでフィルタリングするには、[query] コマンドを使用します。

Linux

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "Soccer matches" \
        --attribute-filter '{"ContainsAny":{"Key":"EVENT","Value":
{"StringListValue":["Champions League"]}}}' \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

macOS

```
aws kendra query \
        --index-id kendra-index-id \
        --query-text "Soccer matches" \
```

```
        --attribute-filter '{"ContainsAny":{"Key":"EVENT","Value":
{"StringListValue":["Champions League"]}}}' \
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

Windows

```
aws kendra query ^
        --index-id kendra-index-id ^
        --query-text "Soccer matches" ^
        --attribute-filter '{"ContainsAny":{"Key":"EVENT","Value":
{"StringListValue":["Champions League"]}}}' ^
        --region aws-region
```

実行する条件は以下のとおりです。

- *kendra-index-id* は、保存した kendra-index-id、
- *aws-region* は、AWS リージョンです。

フィルタリングされた検索結果が AWS CLI に表示されます。

ステップ 6: クリーンアップする

ファイルをクリーンアップする

AWS アカウントで課金を中止するには、このチュートリアルを完了した後、以下の手順を実行できます。

1. Amazon S3 バケットを削除する

   バケットを削除する方法の詳細については、バケットの削除を参照してください。

2. Amazon Kendra インデックスを削除する

Amazon Kendra インデックスを削除する方法の詳細については、インデックスの削除を参照してください。

3. **converter.py** を削除する
   - コンソールの場合: AWS CloudShell に移動し、リージョンが AWS リージョンに設定されていることを確認します。bash シェルがロードされたら、以下のコマンドを環境に入力し、[Enter] キーを押します。

     ```
     rm converter.py
     ```

   - For AWS CLI: ターミナルウィンドウで、以下のコマンドを実行します。

     Linux

     ```
     rm file/converter.py
     ```

     実行する条件は以下のとおりです。

     - *path/* は、ローカルデバイス上の converter.py へのファイルパスです。

     macOS

     ```
     rm file/converter.py
     ```

     実行する条件は以下のとおりです。

     - *path/* は、ローカルデバイス上の converter.py へのファイルパスです。

     Windows

     ```
     rm file/converter.py
     ```

     実行する条件は以下のとおりです。

     - *path/* は、ローカルデバイス上の converter.py へのファイルパスです。

詳細

Amazon Kendra をワークフローに統合する方法の詳細については、以下のブログ記事をご覧ください。

- 詳細検索のためのコンテンツメタデータのタグ付け

- 自動化コンテンツに富んだインテリジェントな検索ソリューションを構築する

Amazon Comprehend の詳細については、Amazon Comprehend デベロッパーガイドを参照してください。

Amazon Kendra のモニタリングとログ記録

トピック

インデックスのモニタリング (コンソール)

Amazon Kendra コンソールを使用して、インデックスとデータソースの状態をモニタリングします。この情報を使用して、インデックスのサイズとストレージ要件を追跡し、インデックスとデータソース間の同期の進行状況と成功をモニタリングできます。

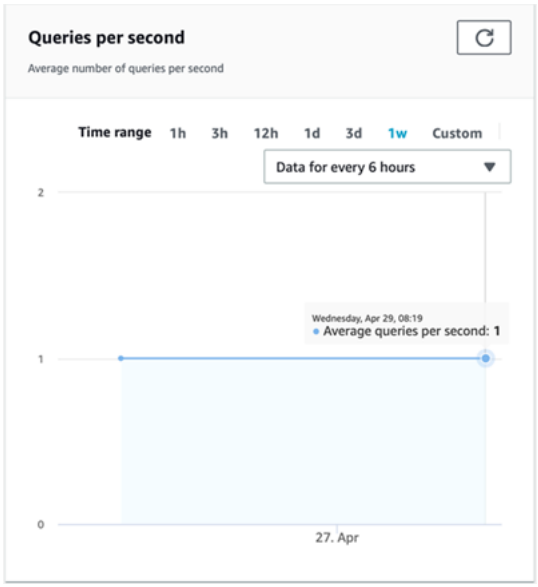
インデックスメトリクスを表示するには (コンソール)

1. AWS Management Console にサインインして、Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/home) を開きます。
2. インデックスのリストから、表示するインデックスを選択します。
3. 画面をスクロールして、インデックスメトリクスを表示します。

インデックスに関する以下のメトリクスを確認できます。

- ドキュメント数 - インデックスが作成されたドキュメントの合計数。これには、すべてのデータソースのすべてのドキュメントが含まれます。このメトリクスを使用して、より多くのまたは少ないインデックスのストレージユニットを購入する必要があるかどうかを決定します。

- 1 秒あたりのクエリ数 - 1 秒間に要求されるインデックスクエリの数。このメトリクスを使用して、より多くのまたは少ないインデックスのクエリユニットを購入する必要があるかどうかを決定します。



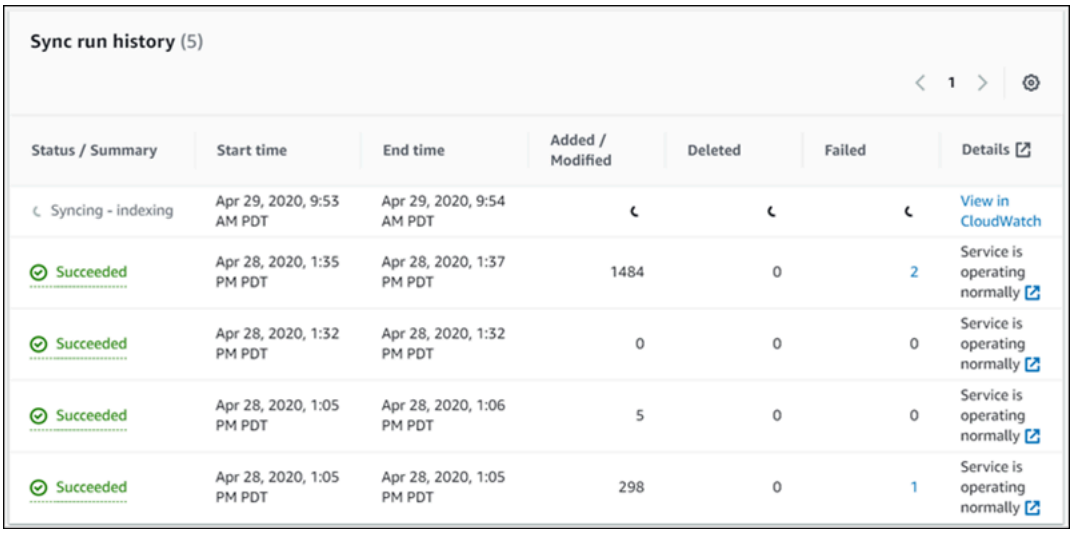
インデックスとデータソース間の同期の進行状況と成功をモニタリングするには、Amazon Kendra コンソールを使用します。この情報を使用して、データソースの正常性を判断します。

同期メトリクスを表示するには (コンソール)

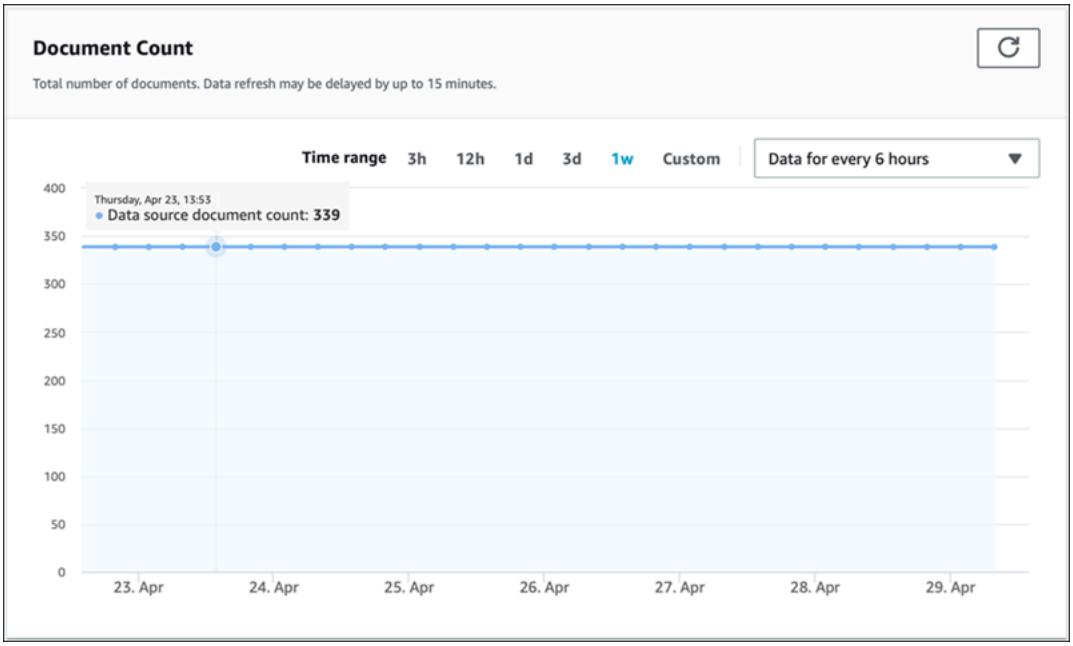
1. AWS Management Console にサインインして、Amazon Kendra コンソール (https://console.aws.amazon.com/kendra/home) を開きます。
2. インデックスのリストから、同期メトリクスを表示するインデックスを選択します。
3. 左側のメニューから [Data sources] (データソース) を選択します。
4. データソースのリストから、表示するデータソースを選択します。
5. 画面をスクロールして、同期実行メトリクスを表示します。

表示できる情報は次のとおりです。

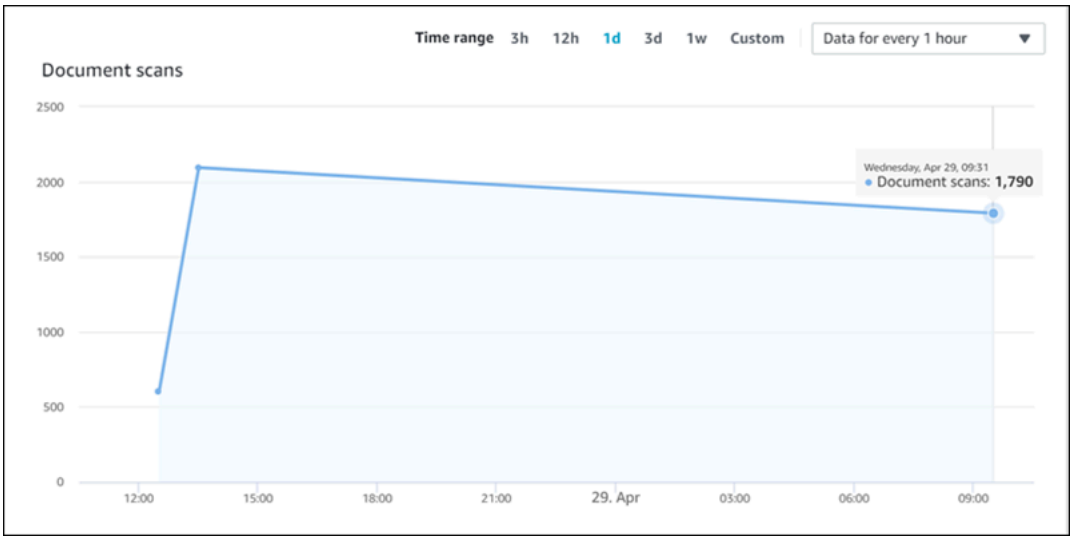
- 同期実行履歴 - 開始時刻と終了時刻、追加、削除、失敗したドキュメントの数など、同期実行に関する統計。同期の実行に失敗すると、CloudWatch Logs へのリンクが詳細に表示されます。左上の設定アイコンを選択して、履歴に表示される列を変更します。この情報を使用して、データソースの全般的な正常性を判断します。



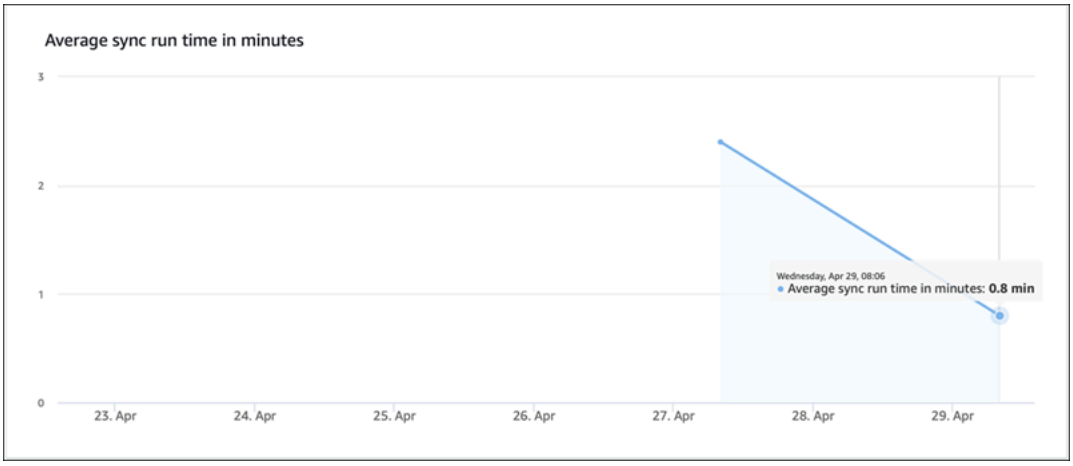
- ドキュメント数 - このデータソースからインデックスが作成されたドキュメントの合計数。これは、データソースに追加されたすべてのドキュメントの合計からデータソースから削除されたすべてのドキュメントの合計を引いた値です。この情報を使用して、このデータソースからインデックスに含まれるドキュメントの数を特定します。



- ドキュメントスキャン - 同期実行中にスキャンされたドキュメントの合計数。これには、追加、更新、削除されたドキュメント、または変更されていないドキュメントなど、データソース内のすべてのドキュメントが含まれます。この情報を使用して、Amazon Kendra がデータソース内のすべてのドキュメントをスキャンしているかどうかを判断します。スキャンされたドキュメントの数は、サービスの請求額に影響します。



- 平均同期実行時間 (分) - 同期実行が完了するまでにかかった平均時間。データソースの同期にかかる時間は、サービスの請求額に影響します。



AWS CloudTrail ログでの Amazon Kendra API コールのログ記録

Amazon Kendraは、Amazon Kendra 内のユーザー、ロール、または AWS サービスによって実行されたアクションのレコードを提供するサービスである AWS CloudTrail と統合されています。CloudTrail は、Amazon Kendra コンソールからの呼び出しと Amazon Kendra API へのコード呼び出しからの呼び出しを含む、Amazon Kendra からのすべての API コールをイベントとしてキャプチャします。追跡を作成する場合は、Amazon Kendra 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できます。追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (イベント履歴) で最新のイベントを表示できます。CloudTrail で収集された情報を使用して、Amazon Kendra 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。

設定や有効化の方法など、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 内の Amazon Kendra 情報

CloudTrail は、AWS アカウント作成時にアクティブ化されます。Amazon Kendra でアクティビティが発生すると、そのアクティビティは CloudTrail [Event history] (イベント履歴) の他の AWS サービスイベントと共に CloudTrail イベントに記録されます。最近のイベントは、AWS アカウントで表示、検索、ダウンロードできます。詳細については、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 Kendra 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] は、指定された S3 バケットに CloudTrail がイベントをログファイルとして配信できるようにする設定です。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。証跡では、AWS パーティションのすべてのリージョ

ンからのイベントがログに記録され、指定した S3 バケットにログファイルが配信されます。さらに、CloudTrail ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS のサービスを設定できます。詳細については、次を参照してください。

- 証跡を作成するための概要
- CloudTrail のサポート対象サービスと統合
- Amazon SNS の CloudTrail の通知の設定
- 「複数のリージョンから CloudTrail ログファイルを受け取る」および「複数のアカウントから CloudTrail ログファイルを受け取る」

CloudTrail では、API リファレンスで文書化されているすべての Amazon Kendra アクションがログに記録されます。たとえば `CreateIndex`、`CreateDataSource`、および `Query` の各オペレーションへのコールは、CloudTrail ログファイル内にエントリを生成します。

各イベントまたはログエントリには、リクエストの生成者に関する情報が含まれます。詳細については、CloudTrail userIdentity 要素を参照してください。

例: Amazon Kendra ログファイルのエントリ

[証跡] は、指定された S3 バケットにイベントをログファイルとして配信できるようにする設定です。CloudTrail のログファイルには、単一か複数のログエントリがあります。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、パブリック 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。

`Query` 演算を呼び出すと、以下のエントリが作成されます。

```
       {
           "eventVersion": "1.05",
           "userIdentity": {
               "type": "AssumedRole | FederatedUser | IAMUser | Root | SAMLUser |
WebIdentityUser",
               "principalId": "principal ID",
               "arn": "ARN",
               "accountId": "account ID",
               "accessKeyId": "access key ID",
               "sessionContext": {
                   "sessionIssuer": {
                       "type": "Role",
                       "principalId": "principal Id",
```

```
                    "arn": "ARN",
                    "accountId": "account ID",
                    "userName": "user name"
                },
                "webIdFederationData": {

                },
                "attributes": {
                    "mfaAuthenticated": false,
                    "creationDate": "timestamp"
                }
            }
        },
        "eventTime": "timestamp",
        "eventSource": "kendra.amazonaws.com",
        "eventName": "Query",
        "awsRegion": "region",
        "sourceIPAddress": "source IP address",
        "userAgent": "user agent",
        "requestParameters": {
            "indexId": "index ID"
        },
        "responseElements": null,
        "requestID": "request ID",
        "eventID": "event ID",
        "eventType": "AwsApiCall",
        "recipientAccountId": "account ID"
    },
```

AWS CloudTrail ログでの Amazon Kendra インテリジェントランキング API コールのログ記録

Amazon Kendra インテリジェントランキングは、Amazon Kendra インテリジェントランキング内のユーザー、ロール、または AWS サービスによって実行されたアクションのレコードを提供するサービスである AWS CloudTrail と統合されています。CloudTrail は、Amazon Kendra インテリジェントランキング API へのコード呼び出しを含む、Amazon Kendra インテリジェントランキングからのすべての API コールをイベントとしてキャプチャします。追跡を作成する場合は、Amazon Kendra インテリジェントランキングのイベントなど、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。追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (イベント履歴) で最新のイベントを表示できます。CloudTrail で収集された情報を使用し

て、Amazon Kendra インテリジェントランキング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。

設定や有効化の方法など、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 内の Amazon Kendra インテリジェントランキングの情報

CloudTrail は、AWS アカウント作成時にアクティブ化されます。Amazon Kendra インテリジェントランキングでアクティビティが発生すると、そのアクティビティは CloudTrail [イベント履歴] の他の AWS サービスイベントとともに CloudTrail イベントに記録されます。最近のイベントは、AWS アカウントで表示、検索、ダウンロードできます。詳細については、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 Kendra インテリジェントランキングのイベントなど、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] は、指定された S3 バケットに CloudTrail がイベントをログファイルとして配信できるようにする設定です。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。証跡では、AWS パーティションのすべてのリージョンからのイベントがログに記録され、指定した S3 バケットにログファイルが配信されます。さらに、CloudTrail ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS のサービスを設定できます。詳細については、次を参照してください。

- 証跡を作成するための概要
- CloudTrail のサポート対象サービスと統合
- Amazon SNS の CloudTrail の通知の設定
- 「複数のリージョンから CloudTrail ログファイルを受け取る」および「複数のアカウントから CloudTrail ログファイルを受け取る」

CloudTrail では、API リファレンスで文書化されているすべての Amazon Kendra インテリジェントランキングのアクションがログに記録されます。例えば、`CreateRescoreExecutionPlan` への呼び出しによって、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、リクエストの生成者に関する情報が含まれます。詳細については、CloudTrail userIdentity 要素を参照してください。

例: Amazon Kendra インテリジェントランキングのログファイルのエントリ

[証跡] は、指定された S3 バケットにイベントをログファイルとして配信できるようにする設定です。CloudTrail のログファイルには、単一か複数のログエントリがあります。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、パブリック 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。

`CreateRescoreExecutionPlan` 演算を呼び出すと、以下のエントリが作成されます。

```
{
            "eventVersion": "1.08",
            "userIdentity": {
                "type": "AssumedRole",
                "principalId": "principal ID",
                "arn": "ARN",
                "accountId": "account ID",
                "accessKeyId": "access key ID",
                "sessionContext": {
                    "sessionIssuer": {
                        "type": "Role",
                        "principalId": "principal ID",
                        "arn": "ARN",
                        "accountId": "account ID",
                        "userName": "user name"
                    },
                    "webIdFederationData": {},
                    "attributes": {
                        "creationDate": "yyyy-mm-ddThh:mm:ssZ",
                        "mfaAuthenticated": "false"
                    }
                }
            },
            "eventTime": "yyyy-mm-ddThh:mm:ssZ",
            "eventSource": "kendra-ranking.amazonaws.com",
            "eventName": "CreateRescoreExecutionPlan",
            "awsRegion": "region",
            "sourceIPAddress": "source IP address",
            "userAgent": "user agent",
            "requestParameters": {
                "name": "name",
```

```
            "description": "description",
            "clientToken": "client token"
        },
        "responseElements": {
            "id": "rescore execution plan ID",
            "arn": "rescore execution plan ARN"
        },
        "requestID": "request ID",
        "eventID": "event ID",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "account ID",
        "eventCategory": "Management",
        "tlsDetails": {
            "tlsVersion": "TLS version",
            "cipherSuite": "cipher suite",
            "clientProvidedHostHeader": "kendra-ranking.[region].api.aws"
        }
    }
```

Amazon CloudWatch による Amazon Kendra のモニタリング

インデックスの正常性を追跡するには、Amazon CloudWatch を使用します。CloudWatch では、インデックスのドキュメント同期のメトリクスを取得できます。定義したしきい値を 1 つ以上のメトリクスが超えたときに通知するよう CloudWatch アラームを設定することもできます。例えば、インデックス作成するために送信されたドキュメントの数やインデックス作成に失敗したドキュメントの数をモニタリングできます。

CloudWatch で Amazon Kendra をモニタリングするには、適切な CloudWatch アクセス許可が必要です。詳細については、Amazon CloudWatch ユーザーガイドのAmazon CloudWatch に対する認証とアクセスコントロールを参照してください。

Amazon Kendra メトリクスの表示

Amazon CloudWatch コンソールを使用して Amazon Kendra メトリクスを表示します。

メトリクスを表示する方法 (CloudWatch コンソール)

1. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。

2. [Metrics] (メトリクス)、[All Metrics] (すべてのメトリクス)、[Kendra] の順に選択します。
3. ディメンションを選択してメトリクスの名前を選んだら、[Add to graph] (グラフへ追加) を選択します。
4. 日付範囲の値を選択します。選択した日付範囲のメトリクスカウントがグラフに表示されます。

アラームを作成する

CloudWatch アラームは指定期間中に単一のメトリクスを監視し、1 つ以上のアクションを実行して Amazon Simple Notification Service (Amazon SNS) トピックまたは Auto Scaling ポリシーに通知を送信します。アクションは、複数の指定期間にわたって特定のしきい値を基準としたメトリクスの値に応じて実行されます。アラームの状態が変わったときにも、CloudWatch は Amazon SNS メッセージを送信できます。

CloudWatch アラームがアクションを呼び出すのは、状態が変わってから指定期間が経過するまで、その新しい状態が続いた場合に限ります。

アラームを設定するには

1. AWS Management Console にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。
2. [アラーム] を選択し、次に [アラームの作成] を選択します。
3. メトリクスを選択します。インデックスとデータソースの [Kendra] メトリクスを選択します。また、時間を時間、日、週、またはカスタムから設定します。
4. 統計情報を選択します。例えば、[平均] などです。また、アラームのトリガー時間を分単位、時間単位、1 日単位、またはカスタムから選択します。
5. アラームをトリガーするために使用するしきい値 (固定値、幅) と、しきい値に該当する条件を選択します。
6. トリガーのアラーム状態 (メトリクスが設定したしきい値を外れる必要があるか、または別の状態か) を選択します。アラームの通知先/E メールを選択します。
7. アラームに問題がなければ、[アラームの作成] を選択します。

> **Note**
>
> CloudWatch アラームの名前を入力する必要があります。

インデックス同期ジョブの CloudWatch メトリクス

次の表は、データソース同期ジョブの Amazon Kendra メトリクスの説明です。

API または CLI を使用する場合は、GetMetricStatistics API を使用するときに選択した `MetricName` に加えて、`Namespace` を「AWS/Kendra」として指定する必要があります。

メトリクス	説明
`DocumentsCrawled`	同期ジョブの実行中にスキャンまたは検出されたドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
`DocumentsSubmittedForIndexing`	同期ジョブがインデックスに送信したドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
`DocumentsSubmittedForIndexingFailed`	インデックス作成に失敗したドキュメントの数。詳細については、同期ジョブの CloudWatch ログの内容を確認してください。 ディメンション: • IndexId • DataSourceId

メトリクス	説明
	単位: 個
DocumentsSubmittedForDeletion	同期ジョブがインデックスから削除するよう指示したドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsSubmittedForDeletionFailed	削除に失敗したドキュメントの数。詳細については、同期ジョブの CloudWatch ログの内容を確認してください。 ディメンション: • IndexId • DataSourceId 単位: 個

Amazon Kendra データソースのメトリクス

次の表は、データソース同期ジョブの Amazon Kendra メトリクスの説明です。アスタリスク (*) が付いたメトリクスは、Amazon S3 データソースにのみ使用されます。

API または CLI を使用する場合は、GetMetricStatistics API を使用するときに選択した MetricName に加えて、Namespace を「AWS/Kendra」として指定する必要があります。

メトリクス	説明
DocumentsSkippedNoChange *	調査され、インデックス作成のために送信されなかったために変更されていないことが判明したドキュメント数。

メトリクス	説明
	ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsSkippedInvalidMetadata *	関連するメタデータファイルに問題があったため、スキップされたドキュメントの数。詳細については、同期実行の CloudWatch ログの内容を確認してください。 ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsCrawled	検査されたドキュメントファイルの数。 ディメンション: • IndexId • DataSourceId 単位: 個

メトリクス	説明
DocumentsSubmittedForDeletion	データソースから削除され、削除のために送信された調査済みドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsSubmittedForDeletionFailed	データソースからの削除に失敗したドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsSubmittedForIndexing	インデックス作成のために調査および送信されたドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個

メトリクス	説明
DocumentsSubmittedForIndexingFailed	インデックス作成のために送信され、インデックス作成できなかったドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個

インデックス作成されたドキュメントのメトリクス

次の表は、インデックス作成されたドキュメントの Amazon Kendra メトリクスの説明です。BatchPutDocument オペレーションを使用してインデックス作成されたドキュメントの場合、IndexId ディメンションのみがサポートされています。

API または CLI を使用する場合は、GetMetricStatistics API を使用するときに選択した MetricName に加えて、Namespace を「AWS/Kendra」として指定する必要があります。

メトリクス	説明
DocumentsIndexed	インデックスが作成されたドキュメントの数。 ディメンション: • IndexId • DataSourceId 単位: 個
DocumentsFailedToIndex	インデックスが作成できなかったドキュメントの数。詳細については、CloudWatch ログの内容を確認してください。 ディメンション:

メトリクス	説明
	• IndexId • DataSourceId 単位: 個
IndexQueryCount	1 分あたりのインデックスクエリの数。 ディメンション: • IndexId 単位: 個

Amazon CloudWatch Logs による Amazon Kendra のモニタリング

Amazon Kendra は Amazon CloudWatch Logs を使用して、データソースの操作に関するインサイトを提供します。Amazon Kendra のログは、インデックスが作成されたドキュメントの詳細をログに記録します。ドキュメントのインデックス作成中に発生したデータソースからのエラーをログに記録します。CloudWatch Logs を使用して、 ログファイルをモニタリング、保存、およびアクセスできます。

CloudWatch Logs は、ロググループの一部であるログストリームにログイベントを保存します。Amazon Kendra はこれらの機能を次のように使用します。

- ロググループ - Amazon Kendra は、すべてのログストリームをインデックスごとに 1 つのロググループに保存します。Amazon Kendra は、インデックスの作成時にロググループを作成します。ロググループ識別子は常に「aws/kendra/」で始まります。
- ログストリーム - Amazon Kendra は、実行するインデックス同期ジョブごとに、ロググループに新しいデータソースログストリームを作成します。また、ストリームが約 500 エントリに達すると、新しいドキュメントログストリームが作成されます。
- ログエントリ - Amazon Kendra は、ドキュメントのインデックス作成時にログストリームにログエントリを作成します。各エントリは、ドキュメントの処理または発生したエラーに関する情報を提供します。

CloudWatch Logs の使用方法の詳細については、Amazon CloudWatch Logs ユーザーガイドの Amazon CloudWatch Logs とはを参照してください。

Amazon Kendra は 2 つのタイプのログストリームを作成します。

- データソースログストリーム
- ドキュメントログストリーム

データソースログストリーム

データソースログストリームは、インデックス同期ジョブに関するエントリを公開します。各同期ジョブは、エントリの公開に使用する新しいログストリームを作成します。ログストリーム名は次のとおりです。

```
data source id/YYYY-MM-DD-HH/data source sync job ID
```

同期ジョブの実行ごとに、新しいログストリームが作成されます。

データソースログストリームに発行されるログメッセージには 3 つのタイプがあります。

- インデックス作成の送信に失敗したドキュメントのログメッセージ。S3 データソース内のドキュメントに対するこのメッセージの例を以下に示します。

```
{
    "DocumentId": "document ID",
    "S3Path": "s3://bucket/prefix/object",
    "Message": "Failed to ingest document via BatchPutDocument.",
    "ErrorCode": "InvalidRequest",
    "ErrorMessage": "No document metadata configuration found for document attribute
 key  city."
}
```

- 削除の送信に失敗したドキュメントのログメッセージ。以下は、このメッセージの例です。

```
{
    "DocumentId": "document ID",
    "Message": "Failed to delete document via BatchDeleteDocument.",
    "ErrorCode": "InvalidRequest",
    "ErrorMessage": "Document can't be deleted because it doesn't exist."
}
```

- Amazon S3 バケット内のドキュメントの無効なメタデータファイルが見つかった場合のログメッセージ。以下は、このメッセージの例です。

```
{
    "Message": "Found invalid metadata
 file bucket/prefix/filename.extension.metadata.json."
}
```

- SharePoint およびデータベースコネクタの場合、Amazon Kendra はドキュメントにインデックスを作成できない場合にのみメッセージをログストリームに書き込みます。Amazon Kendra がログに記録するエラーメッセージの例を以下に示します。

```
{
    "DocumentID": "document ID",
    "IndexID": "index ID",
    "SourceURI": "",
    "CrawlStatus": "FAILED",
    "ErrorCode": "403",
    "ErrorMessage": "Access Denied",
    "DataSourceErrorCode": "403"
}
```

ドキュメントログストリーム

Amazon Kendra は、インデックス作成中にドキュメントの処理に関する情報をログに記録します。Amazon S3 データソースに保存されたドキュメントの一連のメッセージをログに記録します。Microsoft SharePoint またはデータベースデータソースに保存されているドキュメントについてのみエラーを記録します。

ドキュメントが BatchPutDocument 演算を使用してインデックスに追加された場合、ログストリームには次のように名前が付けられます。

```
YYYY-MM-DD-HH/UUID
```

ドキュメントがデータソースを使用してインデックスに追加された場合では、ログストリームは次のように名前が付けられます。

```
dataSourceId/YYYY-MM-DD-HH/UUID
```

各ログストリームには最大 500 個のメッセージが含まれます。

ドキュメントのインデックス作成に失敗すると、次のメッセージがログストリームに出力されます。

```
{
    "DocumentId": "document ID",
    "IndexName": "index name",
    "IndexId": "index ID"
    "SourceURI": "source URI"
    "IndexingStatus": "DocumentFailedToIndex",
    "ErrorCode": "400 | 500",
    "ErrorMessage": "message"
}
```

Amazon Kendra でのセキュリティ

AWS クラウドセキュリティは最優先事項です。 AWS 顧客は、最もセキュリティに敏感な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の恩恵を受けることができます。

セキュリティは、 AWS お客様とお客様との間で共有される責任です。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 AWS AWS クラウド内でサービスを実行するインフラストラクチャを保護する責任があります。 AWS また、安全に使用できるサービスも提供します。第三者監査人は、AWS、当社のセキュリティの有効性を定期的にテストおよび検証しています。Amazon Kendra に適用されるコンプライアンスプログラムについては、「AWS コンプライアンスプログラム別の対象サービス」「」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、 AWS 使用する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 Kendra 使用時における責任共有モデルの適用法を理解するのに役立ちます。以下のトピックでは、セキュリティとコンプライアンスの目的を満たすように Amazon Kendra を設定する方法について説明します。また、Amazon Kendra AWS リソースのモニタリングと保護に役立つ他のサービスの使用方法についても学びます。

トピック

- Amazon Kendra のデータ保護
- Amazon Kendra とインターフェイス VPC エンドポイント (AWS PrivateLink)
- Amazon Kendra 用の Identity and Access Management
- セキュリティに関するベストプラクティス
- Amazon Kendra でのログ記録とモニタリング
- Amazon Kendra のコンプライアンス検証
- Amazon Kendra の耐障害性
- Amazon Kendra でのインフラストラクチャセキュリティ
- の設定と脆弱性の分析 AWS Identity and Access Management

Amazon Kendra のデータ保護

AWS https://aws.amazon.com/compliance/shared-responsibility-model/、Amazon Kendra のデータ保護に適用されます。このモデルで説明されているように、 AWS はすべて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 クラウドお客様は、このインフラストラクチャでホストされてい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、使用する AWS のサービス のセキュリティ設定と管理タスクもユーザーの責任となります。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」のブログ記事を参照してください。

データ保護のため、 AWS アカウント 認証情報を保護し、 AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。こうすると、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、以下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 を使用してリソースと通信します。 AWS TLS 1.2、できれば TLS 1.3 が必要です。
- を使用して API とユーザーアクティビティの口ギングを設定します。 AWS CloudTrail
- AWS 暗号化ソリューションと、 AWS のサービスその中に含まれるデフォルトのセキュリティコントロールをすべて使用してください。
- Amazon Macie 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ます。これらは、Amazon S3 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。
- コマンドラインインターフェイスまたは API AWS を介してアクセスするときに FIPS 140-2 で検証された暗号モジュールが必要な場合は、FIPS エンドポイントを使用してください。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機密情報やセンシティブ情報は、タグや名前フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これには、コンソール、API AWS CLI、または AWS SDK AWS のサービス を使用して Amazon Kendra やその他のユーザーと作業する場合も含まれます。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

保管中の暗号化

Amazon Kendra は、選択した暗号化キーを使用して、保管中のデータを暗号化します。次のいずれかを選択できます。

- AWS所有している KMS キー AWS 。暗号化キーを指定しない場合、デフォルトでは、データはこのキーを使用して暗号化されます。
- AWSアカウント内の管理対象の KMS キー。このキーは、Amazon Kendra によってお客様に代わって作成、管理、使用されます。キー名は、`aws/kendra` です。
- カスタマーマネージドキー。アカウントで作成した暗号化キーの ARN を指定できます。カスタマーマネージド KMS キーを使用する場合、キーに Amazon Kendra がキーを使用できるようにするキーポリシーを付与する必要があります。対称暗号化カスタマーマネージド KMS キーを選択してください。Amazon Kandra は非対称 KMS キーはサポートしていません。詳細については、「キー管理」を参照してください。

転送中の暗号化

Amazon Kendra は HTTPS プロトコルを使用して、クライアントアプリケーションと通信します。HTTPS AWS と署名を使用して、アプリケーションに代わって他のサービスと通信します。VPC を使用している場合は、 AWS PrivateLink を使用して VPC と Amazon Kendra の間にプライベート接続を確立できます。

キー管理

Amazon Kendra は、3 種類のキーのいずれかを使用してインデックスの内容を暗号化します。次のいずれかを選択できます。

- 所有する KMS。 AWS AWS これがデフォルトです。
- AWS管理対象の KMS キー。このキーは、Amazon Kendra によってお客様に代わってお客様のアカウントで作成され、管理および使用されます。
- カスタマーマネージド KMS キー。Amazon Kendra インデックスまたはデータソースを作成するときにキーを作成するか、 AWS KMS コンソールを使用してキーを作成できます。対称暗号化カスタマーマネージド KMS キーを選択します。Amazon Kendra は非対称 KMS キーをサポートしていません。詳細については、AWS Key Management Service Developer Guide の対称キーと非対称キーの使用を参照してください。

Amazon Kendra とインターフェイス VPC エンドポイント (AWS PrivateLink)

VPC と Amazon Kendra 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AWS PrivateLink、NAT デバイス、VPN 接続、または AWS Direct Connect 接続なしで Amazon Kendra API にプライベートにアクセスできるテクノロジーを利用しています。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 Kendra API と通信できます。VPC と Amazon Kendra との間のトラフィックは、Amazon ネットワークを離れません。

各インターフェースエンドポイントは、サブネット内の 1 つ以上の 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 Kendra VPC エンドポイントに関する考慮事項

Amazon Kendra のインターフェイス 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 Kendra は、VPC からのすべての API アクションの呼び出しをサポートしています。

Amazon Kendra 用のインターフェイス VPC エンドポイントの作成

Amazon Kendra サービス用の VPC エンドポイントは、Amazon VPC コンソールまたは () のいずれかを使用して作成できます。 AWS Command Line Interface AWS CLI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。

Amazon Kendra 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*.kendra

VPC エンドポイントを作成したら、`endpoint-url`パラメータを使用して Amazon Kendra API AWS CLI へのインターフェイスエンドポイントを指定する以下のコマンド例を使用できます。

```
aws kendra list-indices --endpoint-url https://VPC endpoint
```

[VPC endpoint] (VPC エンドポイント) は、インターフェイスエンドポイントの作成時に生成される DNS 名です。この名前には、VPC エンドポイント ID、Amazon Kendra サービス名、およびリージョン名が含まれます。例えば `vpce-1234-abcdef.kendra.us-west-2.vpce.amazonaws.com` です。

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 (`kendra.us-east-1.amazonaws.com` など) を使用して、Amazon Kendra 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。

Amazon Kendra 用の VPC エンドポイントポリシーの作成

Amazon KendraへのアクセスをコントロールするVPC エンドポイントにエンドポイントポリシーをアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- このアクションを実行できるリソース。

詳細については、Amazon VPC ユーザーガイドの VPC エンドポイントによるサービスのアクセスコントロールを参照してください。

例: Amazon Kendra アクションの VPC エンドポイントポリシー

Amazon Kendraのエンドポイントポリシーの例を次に示します。このポリシーは、エンドポイントに添付されると、すべてのリソースのすべてのプリンシパルに対して、登録されている Amazon Kendra アクションへのアクセスを許可します。

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "kendra:Query"
         ],
         "Resource":"*"
      }
```

```
    ]
}
```

Amazon Kendra 用の Identity and Access Management

AWS Identity and Access Management (IAM) は、 AWS のサービス 管理者がリソースへのアクセスを安全に制御できるようにするものです。 AWS IAM 管理者は、誰を認証 (サインイン) し、誰に Amazon Kendra リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM AWS のサービス は追加料金なしで使用できるアプリです。

トピック

対象者

AWS Identity and Access Management (IAM) の使用方法は、Amazon Kendra で行う作業によって異なります。

サービスユーザー - ジョブを実行するために Amazon Kendra サービスを使用する場合は、管理者から必要なアクセス許可と認証情報が与えられます。さらに多くの Amazon Kendra 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。Amazon Kendra の機能にアクセスできない場合は、Amazon Kendra アイデンティティとアクセスのトラブルシューティング を参照してください。

サービス管理者 - 社内の Amazon Kendra リソースを担当している場合は、Amazon MQ に対する完全なアクセス権があると思われます。サービスのユーザーがどの Amazon Kendra 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概

念を理解してください。会社で Amazon Kendra と IAM を併用する方法の詳細については、Amazon Kendra で IAM が機能する仕組み を参照してください。

IAM 管理者 - 管理者は、Amazon Kendra 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 で使用できる Amazon Kendra アイデンティティベースのポリシーの例を表示するには、Amazon Kendra のアイデンティティベースポリシーの例 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 AWS 認証情報を使用してサインインする方法です。IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証 (サインイン AWS) する必要があります。

ID ソースを通じて提供された認証情報を使用して、フェデレーション ID AWS としてサインインできます。 AWS IAM Identity Center フェデレーテッド ID の例としては、(IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google や Facebook の認証情報などがあります。フェデレーションアイデンティティとしてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。 AWS フェデレーションを使用してアクセスすると、間接的にロールを引き継ぐことになります。

ユーザーのタイプによっては、 AWS Management Console AWS またはアクセスポータルにサインインできます。へのサインインについて詳しくは AWS、『AWS サインイン ユーザーガイド』の「AWS アカウントにサインインする方法」を参照してください。

AWS プログラムでアクセスする場合は、認証情報を使用してリクエストに暗号署名するためのソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨方法を使用して自分でリクエストに署名する方法の詳細については、IAM ユーザーガイドの「 AWS API リクエストへの署名」を参照してください。

使用する認証方法を問わず、セキュリティ情報の提供を追加でリクエストされる場合もあります。たとえば、アカウントのセキュリティを強化するために多要素認証 (MFA) AWS を使用することを推奨しています。詳細については、「AWS IAM Identity Center ユーザーガイド」の「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 root ユーザー

を作成するときは AWS アカウント、 AWS のサービス アカウント内のすべてのリソースに完全にアクセスできる 1 つのサインイン ID から始めます。この ID は AWS アカウント root ユーザーと呼ば

れ、アカウントの作成に使用した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されます。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、それらを使用してルートユーザーのみが実行できるタスクを実行してください。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。

IAM ユーザーとグループ

IAM ユーザーは、1 人のユーザーまたはアプリケーションに対して特定の権限を持つ社内の AWS アカウント ID 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保有する IAM 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 ユーザーでの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は、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、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する許可を与えることができます。

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、ロールでは一時的な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 (ロールではなく) の作成が適している場合」を参照してください。

IAM ロール

IAM ロールは、 AWS アカウント 特定の権限を持つ社内の ID です。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。 AWS Management Console ロールを切り替えることで、の IAM ロールを一時的に引き受けることができます。 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。

一時的な認証情報を持った IAM 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 – フェデレーションアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーションアイデンティティが認証

されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。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 アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。

- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (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、ロールをプロキシとして使用する代わりに AWS のサービス、ポリシーをリソースに直接アタッチできるものもあります。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は、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。
- クロスサービスアクセス — AWS のサービス AWS のサービス他の機能を使用するものもあります。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、または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。
  - 転送アクセスセッション (FAS) — IAM ユーザーまたはロールを使用してアクションを実行する場合 AWS、あなたはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 AWS のサービス を呼び出したプリンシパルの権限をリクエスト元と組み合わせて使用して AWS のサービス、ダウンストリームサービスにリクエストを行います。FASリクエストは、AWS のサービス サービスが他のユーザーとのやりとりやリソースとのやり取りを必要とするリクエストを受信したときにのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。
  - サービスロール - 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、にリンクされている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。 AWS アカウント サービスにリンクされたロールはに表示され、そのサービスが所有します。IAM 管理者は、サービスリンクロールの許可を表示できますが、編集することはできません。
- Amazon EC2 で実行されるアプリケーション — IAM ロールを使用して、EC2 インスタンスで実行され、 AWS API AWS CLI 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、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。EC2 AWS インスタンスにロールを割り当て、そのロールをそのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされる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、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与する」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、「IAM ユーザーガイド」の「(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか」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

AWS ポリシーを作成して AWS ID またはリソースにアタッチすることで、アクセスを制御します。ポリシーとは、ID またはリソースに関連付けると権限を定義するオブジェクトです。 AWS AWS 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、これらのポリシーを評価します。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 AWS ドキュメント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 AWS JSON 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。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの権限を定義します。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は AWS Management Console、、 AWS CLI、または AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

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のポリシーを作成する方法については、「IAM ユーザーガイド」の「IAM ポリシーの作成」を参照してください。

アイデンティティベースのポリシーは、さらに インラインポリシー または マネージドポリシー に分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、内の複数のユーザー、グループ、および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 管理ポリシーと顧客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いずれかを選択する方法については、「IAM ユーザーガイド」の「マネージドポリシーとインラインポリシーの比較」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、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーが添付されているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザ、ロール、フェデレーティッドユーザ、またはを含めることができます。 AWS のサービス

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。IAM AWS の管理ポリシーをリソースベースのポリシーで使用することはできません。

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、またはロール) がリソースにアクセスするための権限を持つかをコントロールします。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。

ACL をサポートするサービスの例としては AWS WAF、Amazon S3、、および Amazon VPC があります。ACL の詳細については、「Amazon Simple Storage Service デベロッパーガイド」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 あまり一般的ではないポリシータイプもサポートしています。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる許可の上限を設定する高度な機能です。エンティティに権限の境界を設定できます。結果として得られる権限は、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権限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、権限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、許可は無効になります。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 (SCP) — SCP は、組織または組織単位 (OU) の最大権限を指定する JSON ポリシーです。 AWS Organizations AWS Organizations は、 AWS アカウント 企業が所有する複数のもの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、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、メンバーアカウントのエンティティ (各エンティティを含む) の権限を制限します。 AWS アカウントのルートユーザー 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ティッド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。結果としてセッションの権限される範囲は、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になります。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、許可は無効になります。詳細については、 IAM ユーザーガイド の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。 AWS 複数のポリシータイプが関係している場合にリクエストを許可するかどうか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 Kendra で IAM が機能する仕組み

IAM を使用して Amazon Kendra へのアクセスを管理する前に、Amazon Kendra で使用できる IAM 機能について理解しておく必要があります。Amazon Kendra AWS やその他のサービスが IAM とどのように連携するかを大まかに把握するには、IAM ユーザーガイドの「IAM AWS と連携するサービス」を参照してください。

トピック

- Amazon Kendra アイデンティティベースのポリシー
- Amazon Kendra リソースベースのポリシー
- アクセスコントロールリスト (ACL)
- Amazon Kendra タグに基づいた認可
- Amazon Kendra IAM ロール

Amazon Kendra アイデンティティベースのポリシー

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、アクションを許可または拒否する条件を指定できます。Amazon Kendra は、特定のアクション、リソース、および条件キーをサポートしています。JSON ポリシーで使用するすべての要素については、「IAM ユーザーガイド」の「IAM JSON ポリシー要素のリファレンス」を参照してください。

アクション

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action`要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションには通常、関連する AWS API オペレーションと同じ名前が付けられます。一致する API オペレーションのない 許可のみのアクション など、いくつかの例外があります。また、ポリシーに複数アクションが必要なオペレーションもあります。これらの追加アクションは、 依存アクション と呼ばれます。

このアクションは、関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

Amazon Kendra のポリシーアクションは、アクションの前にプレフィックス `kendra:` を使用します。たとえば、ListIndicesAPI オペレーションで Amazon Kendra インデックスを一覧表示するアク

セス権限を誰かに付与するには、kendra:ListIndicesそのアクションをそのユーザーのポリシーに含めます。ポリシーステートメントには、Action または NotAction 要素を含める必要があります。Amazon Kendra は、このサービスで実行できるタスクを記述する独自のアクションのセットを定義します。

単一ステートメントに複数アクションを指定するには、次のようにカンマで区切ります:

```
"Action": [
      "kendra:action1",
      "kendra: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、Describe 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": "kendra:Describe*"
```

Amazon Kendra アクションのリストを確認するには、IAM ユーザーガイドの Amazon MQ で定義されるアクションを参照してください。

リソース

管理者は AWS JSON 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 ポリシーの要素は、オブジェクトあるいはアクションが適用されるオブジェクトを指定します。ステートメントには、Resource または NotResource要素を含める必要があります。ベストプラクティスとしては、Amazon リソースネーム (ARN) を使用してリソースを指定します。これは、 リソースレベルの許可 と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの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 Kendra インデックスリソースには次のネーム ARN があること。

```
arn:${Partition}:kendra:${Region}:${Account}:index/${IndexId}
```

ARN の形式の詳細については、「Amazon リソースネーム (ARN) AWS とサービス名前空間」を参照してください。

例えば、ステートメントでインデックスを指定するには、次の ARN のインデックスの GUID を使用します。

```
"Resource": "arn:aws:kendra:${Region}:${Account}:index/${GUID}"
```

特定のアカウントに属するすべてのインデックスを指定するには、ワイルドカード (*) を使用します。

```
"Resource": "arn:aws:${Region}:${Account}:index/*"
```

リソースを作成するためのアクションなど、Amazon Kendra アクションには特定のリソースで実行できないものがあります。このような場合は、ワイルドカード *を使用する必要があります。

```
"Resource": "*"
```

Amazon Kendra のリソースタイプとそれらの ARN のリストを確認するには、IAM ユーザーガイドの Amazon Kendra で定義されるリソースタイプを参照してください。どのアクションで各リソースの ARN を指定できるかについては、Amazon Kendra で定義されるアクションを参照してください。

条件キー

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どの プリンシパル がどの リソース に対してどのような条件下 で アクション を実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。イコールや未満などの 条件演算子 を使用して条件式を作成することで、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition要素を指定する場合、または 1 つの Condition要素に複数のキーを指定する場合、 AWS では AND論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 AWS OR論理演算を使用して条件を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、 IAM ユーザーガイド の「IAM ポリシーの要素: 変数およびタグ」を参照してください。

AWS グローバル条件キーとサービス固有の条件キーをサポートします。 AWS すべての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 Kendra にはサービス固有条件キーがありませんが、いくつかのグローバル条件キーの使用をサポートしています。 AWS すべての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。

例

Amazon Kendra のアイデンティティベースポリシーの例を確認するには、Amazon Kendra のアイデンティティベースポリシーの例 を参照してください。

Amazon Kendra リソースベースのポリシー

Amazon Kendra では、リソースベースのポリシーはサポートされていません。

アクセスコントロールリスト (ACL)

Amazon Kendra では、 AWS のサービスとリソースへのアクセスのためのアクセスコントロールリスト (ACL) をサポートしていません。

Amazon Kendra タグに基づいた認可

特定のタイプの Amazon Kendra リソースにタグを関連付けると、これらのリソースへのアクセスを認可できます。タグに基づいてアクセスをコントロールするには、`aws:RequestTag/key-name`、または `aws:TagKeys` 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。

次の表に、タグベースのアクセスコントロールのアクション、対応するリソースタイプおよび条件キーを示します。各アクションは、対応するリソースタイプに関連付けられたタグに基づいて許可されます。

アクション	リソースタイプ	条件キー
CreateDataSource		`aws:RequestTag` , `aws:TagKeys`

アクション	リソースタイプ	条件キー
CreateFaq		aws:RequestTag , aws:TagKeys
CreateIndex		aws:RequestTag , aws:TagKeys
API_ ListTagsForResource	データソース、よくある質問、インデックス	
TagResource	データソース、よくある質問、インデックス	aws:RequestTag , aws:TagKeys
UntagResource	データソース、よくある質問、インデックス	aws:TagKeys

Amazon Kendra リソースのタグ付けの詳細については、タグ を参照してください。リソースタグに基づいてリソースへのアクセスを制限するアイデンティティベースのポリシーの例については、タグベースのポリシーの例 を参照してください。リソースへのアクセスを制限するためのタグの使用の詳細については、IAM ユーザーガイドのタグを使用したアクセス制御を参照してください。

Amazon Kendra IAM ロール

IAM ロールは、 AWS 特定の権限を持つアカウント内のエンティティです。

Amazon Kendra での一時的な認証情報の使用

一時的な認証情報を使用して、フェデレーションでサインインする、IAM ロールを引き受ける、またはクロスアカウントロールを引き受けることができます。一時的なセキュリティ認証情報は、AWS STS AssumeRoleGetFederationTokenやなどの API オペレーションを呼び出して取得します。

Amazon Kendra は、一時的な認証情報の使用をサポートします。

サービスロール

この機能により、ユーザーに代わってサービスがサービスロールを引き受けることが許可されます。このロールにより、サービスがお客様に代わって他のサービスのリソースにアクセスし、アクションを完了することが許可されます。サービスロールは、IAM アカウントに表示され、アカウントに

よって所有されます。つまり、IAM 管理者は、このロールの権限を変更できます。ただし、それにより、サービスの機能が損なわれる場合があります。

Amazon Kendra ではサービスロールがサポートされています。

Amazon Kendra での IAM ロールの選択

インデックスの作成、`BatchPutDocument` 演算の呼び出し、データソースの作成、またはよくある質問の作成を行う場合は、Amazon Kendra がユーザーに代わって必要なリソースにアクセスするために使用するアクセスロールの Amazon リソースネーム (ARN) を指定する必要があります。以前に作成したロールがある場合、Amazon Kendra コンソールにより、選択できるロールのリストが提示されます。必要なリソースへのアクセスを許可するロールを選択することが重要です。詳細については、「IAM の アクセスロール Amazon Kendra」を参照してください。

Amazon Kendra のアイデンティティベースポリシーの例

デフォルトで、ユーザーとロールには Amazon Kendra リソースを作成または変更する許可がありません。また、 AWS Management Console AWS CLI、または AWS API を使用してタスクを実行することもできません。IAM 管理者は、ユーザーとロールに必要な、指定されたリソースで特定の API オペレーションを実行する権限をユーザーとロールに付与する IAM ポリシーを作成する必要があります。続いて、管理者はそれらのアクセス許可が必要なユーザーまたはグループにそのポリシーをアタッチします。

JSON ポリシードキュメントのこれらの例を使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「JSON タブでのポリシーの作成」を参照してください。

トピック

- ポリシーのベストプラクティス
- Amazon Kendra のAWS 管理 (事前定義) ポリシー
- 自分の許可の表示をユーザーに許可する
- 1 つの Amazon Kendra インデックスへのアクセス
- タグベースのポリシーの例

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウント内で誰かが Amazon Kendra リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションを実行すると、 AWS アカウントに

料金が発生する可能性があります。アイデンティティベースの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 管理ポリシーから始めて、最小権限の権限に移行する — ユーザーとワークロードへの権限の付与を開始するには、AWS 多くの一般的なユースケースで権限を付与する管理ポリシーを使用してください。これらのポリシーは、で利用できます。 AWS アカウント AWS 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、権限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「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、 最小特権アクセス許可 とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「IAM でのポリシーとアクセス許可」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する – ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。サービスアクションがなどの特定の用途で使用された場合は AWS のサービス、条件を使用してサービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、「IAM ユーザーガイド」の「IAM Access Analyzer ポリシーの検証」を参照してください。
- 多要素認証 (MFA) が必要 — IAM ユーザーまたは root 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、セキュリティを強化するために MFA をオンにしてください。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 保護 API アクセスの設定」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。

Amazon Kendra のAWS 管理 (事前定義) ポリシー

AWS によって作成および管理されるスタンドアロンの IAM ポリシーを提供することで、多くの一般的なユースケースに対応します。 AWSこれらのポリシーは管理ポリシーと呼ばれます AWS 。 AWS 管理ポリシーを使用すると、ポリシーを自分で作成するよりも簡単にユーザー、グループ、ロールにアクセス許可を割り当てることができます。詳細については、「IAM ユーザーガイド」の「ユーザーへのアクセス許可の追加」を参照してください。

AWS アカウントのグループとロールにアタッチできる以下の管理ポリシーは、Amazon Kendra 固有のものです。

- AmazonKendraReadOnly— Amazon Kendra リソースへの読み取り専用アクセスを許可します。
- AmazonKendraFullAccess— すべての Amazon Kendra リソースを作成、読み取り、更新、削除、タグ付け、実行するためのフルアクセスを付与します。

コンソールでは、ロールにも `iam:CreateRole`、`iam:CreatePolicy`、`iam:AttachRolePolicy`、および `s3:ListBucket` アクセス許可が必要です。

> **Note**
>
> これらのアクセス許可については、IAM; コンソールにサインインして特定のポリシーを検索することで確認できます。

独自のカスタム ポリシーを作成して、Amazon Kendra API アクションにアクセス権限を付与することもできます。これらのカスタムポリシーは、それらのアクセス許可が必要な IAM ロールまたはグループにアタッチできます。Amazon Kendra の IAM ポリシーの例については、Amazon Kendra のアイデンティティベースポリシーの例 を参照してください。

自分の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添付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を作成する方法を示します。このポリシーには、コンソールで、またはまたは API を使用してこのアクションをプログラム的に実行するためのアクセス権限が含まれています。 AWS CLI AWS

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

1 つの Amazon Kendra インデックスへのアクセス

この例では、 AWS アカウント内のユーザーにインデックスをクエリするためのアクセス権限を付与する必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "QueryIndex",
            "Effect": "Allow",
```

```
            "Action": [
                "kendra:Query"
            ],
            "Resource": "arn:aws:kendra:${Region}:${Account}:index/${Index ID}"
        }
    ]
}
```

タグベースのポリシーの例

タグベースのポリシーとは、タグ付きリソースに対してプリンシパルとしての実行できるアクションを指定する JSON ポリシードキュメントです。

例: タグを使用したリソースへのアクセス

このサンプルポリシーは、 AWS アカウント内のユーザーまたはロールに、Query**departmentfinance**キーと値がタグ付けされた任意のリソースでオペレーションを使用する権限を付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kendra:Query"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/department": "finance"
                }
            }
        }
   ]
}
```

例: タグを使用して Amazon Kendra 演算を有効にする

このポリシー例は、 AWS アカウントのユーザーまたはロールに、**departmentfinance**キーと値がタグ付けされたリソースでのオペレーションを除く任意の Amazon Kendra TagResource オペレーションを使用するアクセス権限を付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "kendra:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "kendra:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/department": "finance"
                }
            }
        }
    ]
}
```

例: タグを使用して演算へのアクセスを制限する

このサンプルポリシーでは、ユーザーがタグを提供し、**department**`finance`タグに許容値とが設定されている場合を除き、 AWS `CreateIndex`アカウント内のユーザーまたはロールがオペレーションを使用する際のアクセスを制限しています。**IT**

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "kendra:CreateIndex",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": "kendra:CreateIndex",
            "Resource": "*",
            "Condition": {
                "Null": {
```

```
                    "aws:RequestTag/department": "true"
                }
            }
        },
        {
            "Effect": "Deny",
            "Action": "kendra:CreateIndex",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringNotEquals": {
                    "aws:RequestTag/department": [
                        "finance",
                        "IT"
                    ]
                }
            }
        }
    ]
}
```

AWS Amazon Kendra 管理ポリシー

ユーザー、グループ、ロールにアクセス権限を追加するには、 AWS 自分でポリシーを作成するよりも管理ポリシーを使用する方が簡単です。チームに必要な許可のみを提供する IAM カスタマーマネージドポリシーを作成するには、時間と専門知識が必要です。すぐに始めるには、 AWS 管理ポリシーをご利用ください。これらのポリシーは一般的なユースケースを対象としており、 AWS お客様のアカウントで利用できます。 AWS 管理ポリシーの詳細については、IAM ユーザーガイドの「AWS 管理ポリシー」を参照してください。

AWS AWS サービスは管理ポリシーを維持および更新します。 AWS 管理ポリシーの権限は変更できません。サービスでは、新しい機能を利用できるようにするために、 AWS マネージドポリシーに権限が追加されることがあります。この種類の更新は、ポリシーがアタッチされている、すべてのアイデンティティ (ユーザー、グループおよびロール) に影響を与えます。新しい機能が立ち上げられた場合や、新しいオペレーションが使用可能になった場合に、各サービスが AWS マネージドポリシーを更新する可能性が最も高くなります。 AWS サービスは管理ポリシーから権限を削除しないため、ポリシーを更新しても既存の権限が損なわれることはありません。

さらに、 AWS 複数のサービスにまたがるジョブ機能の管理ポリシーもサポートされます。たとえば、ReadOnlyAccess AWS AWS 管理ポリシーはすべてのサービスとリソースへの読み取り専用ア

クセスを提供します。サービスが新しい機能を起動すると、 AWS 新しい操作やリソースに対する読み取り専用権限が追加されます。ジョブ機能のポリシーの一覧および詳細については、「IAM ユーザーガイド」の「AWS のジョブ機能のマネージドポリシー」を参照してください。

AWS 管理ポリシー: AmazonKendraReadOnly

Amazon Kendra リソースに読み取り専用アクセスを付与します。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。

- kendra - ユーザーは、アイテムのリストまたはアイテムに関する詳細を返すアクションを実行することができます。これには、Describe、List、Query、BatchGetDocumentStatus、GetQuerySuggestions、または GetSnapshots で始まる API 演算が含まれます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "kendra:Describe*",
                "kendra:List*",
                "kendra:Query",
                "kendra:BatchGetDocumentStatus",
                "kendra:GetQuerySuggestions",
                "kendra:GetSnapshots"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

AWS 管理ポリシー: AmazonKendraFullAccess

すべての Amazon Kendra リソースの作成、読み取り、更新、削除、タグ付け、および実行を行うためのフルアクセスを付与します。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。

- kendra - プリンシパルに Amazon Kendra 内のすべてのアクションへの読み取りおよび書き込みアクセスを許可します。
- s3 - プリンシパルに Amazon S3 バケットの場所を取得してバケットを一覧表示できるようにします。
- iam - プリンシパルがロールを渡して一覧表示できるようにします。
- kms AWS KMS —プリンシパルがキーとエイリアスを記述して一覧表示できるようにします。
- secretsmanager - プリンシパルが、シークレットの作成、記述、一覧表示ができるようにします。
- ec2 - プリンシパルがセキュリティグループ、VCP (仮想プライベートクラウド)、およびサブネットを記述できるようにします。
- cloudwatch - プリンシパルが Cloud Watch メトリクスを表示できるようにします。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "kendra.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets"
            ],
```

```
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:ListKeys",
                "kms:ListAliases",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:GetBucketLocation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:ListSecrets"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricData"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret",
                "secretsmanager:DescribeSecret"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonKendra-*"
        },
        {
            "Effect": "Allow",
            "Action": "kendra:*",
```

```
            "Resource": "*"
        }
    ]
}
```

Amazon Kendra AWS による管理ポリシーの更新

このサービスが変更の追跡を開始して以降の Amazon Kendra AWS の管理ポリシーの更新に関する詳細を表示します。このページの変更に関する自動通知を入手するには、Amazon Kendra ドキュメントの履歴ページから、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
AmazonKendraReadOnly— サポート GetSnapshots、API にアクセス許可を追加 BatchGetDocumentStatus	Amazon Kendra には新しい API `GetSnapshots` および `BatchGetDocumentStatus` が追加されました。`GetSnapshots` はユーザーの検索アプリケーションとのやり取りを示すデータを提供します。`BatchGetDocumentStatus` はドキュメントのインデックス作成の進行状況を監視します。	2022 年 1 月 3 日
AmazonKendraReadOnly— サポート業務への権限の追加 GetQuerySuggestions	Amazon Kendra に、一般的な検索クエリのクエリ提案を取得でき、ユーザーの検索をガイドできる新しい `GetQuerySuggestions` を追加しました。ユーザーが検索クエリを入力すると、提案されるクエリが検索のオートコンプリートを手助けします。	2021 年 5 月 27 日

変更	説明	日付
Amazon Kendra が変更の追跡を開始しました。	Amazon Kendra は、 AWS 管理ポリシーの変更の追跡を開始しました。	2021 年 5 月 27 日

Amazon Kendra アイデンティティとアクセスのトラブルシューティング

次の情報は、Amazon Kendra と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

トピック

- Amazon Kendra でアクションを実行する認可がありません
- 私には IAM を実行する権限がありません:PassRole
- 管理者として Amazon Kendra へのアクセスを他のユーザーに許可したいです
- AWS アカウント外のユーザーが自分の Amazon Kendra リソースにアクセスできるようにしたい

Amazon Kendra でアクションを実行する認可がありません

アクションを実行する権限がないと表示された場合、管理者に連絡して支援を求める必要があります。 AWS Management Console 管理者とは、サインイン認証情報を提供した担当者です。

次のエラー例は、`mateojackson` ユーザーがコンソールを使用してインデックスの詳細を表示しようとする際に、`kendra:DescribeIndex` アクセス許可を持っていない場合に発生します。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 kendra:DescribeIndex on resource: index ARN
```

この場合、Mateo は、`kendra:DescribeIndex` アクションを使用して `index` リソースへのアクセスが許可されるように、管理者にポリシーの更新を依頼します。

私には IAM を実行する権限がありません:PassRole

`iam:PassRole` アクションを実行する権限がないというエラーが表示された場合は、Amazon Kendra にロールを渡すことを許可するようにポリシーを更新する必要があります。

新しいサービスロールやサービスにリンクされたロールを作成する代わりに、 AWS のサービス 既存のロールをそのサービスに渡すことができるものもあります。そのためには、サービスにロールを渡すアクセス許可が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して Amazon Kendra でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、メアリーのポリシーを更新してメアリーに iam:PassRoleアクションの実行を許可する必要があります。

サポートが必要な場合は、管理者に問い合わせてください。 AWS サインイン資格情報を提供した担当者が管理者です。

管理者として Amazon Kendra へのアクセスを他のユーザーに許可したいです

Amazon Kendra へのアクセスを他のユーザーに許可するには、アクセスを必要とする人またはアプリケーション用に IAM エンティティ (ユーザーまたはロール) を作成する必要があります。ユーザーまたはアプリケーションは、そのエンティティの認証情報を使用して AWSにアクセスします。次に、Amazon Kendra の適切な許可を付与するエンティティにポリシーをアタッチする必要があります。

すぐに開始するには、IAM ユーザーガイドの IAM が委任した最初のユーザーおよびグループの作成を参照してください。

AWS アカウント外のユーザーが自分の Amazon Kendra リソースにアクセスできるようにしたい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下にご相談ください。

- Amazon Kendra がこれらの機能をサポートしているかどうかを確認するには、Amazon Kendra で IAM が機能する仕組み を参照してください。
- AWS アカウント 所有しているリソース全体のリソースへのアクセスを提供する方法については、『IAM ユーザーガイド』の「AWS アカウント 所有する別の IAM ユーザーへのアクセスを提供する」を参照してください。
- リソースへのアクセスを第三者に提供する方法については AWS アカウント、IAM ユーザーガイドの「AWS アカウント 第三者が所有するリソースへのアクセスの提供」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、「IAM ユーザーガイド」の「外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可」を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

セキュリティに関するベストプラクティス

Amazon Kendra には、独自のセキュリティポリシーを策定および実装する際に考慮すべきさまざまなセキュリティ機能が用意されています。以下のベストプラクティスは一般的なガイドラインであり、完全なセキュリティソリューションを提供するものではありません。これらのベストプラクティスは顧客の環境に必ずしも適切または十分でない可能性があるため、処方箋ではなく、あくまで有用な検討事項とお考えください。

最小特権の原則を適用する

Amazon Kendra は、 IAM ロールを使用するアプリケーション用のきめ細かなアクセスポリシーを提供します。ロールには、アプリケーションのカバーやログ送信先へのアクセスなど、ジョブに必要な最小限の特権セットのみを付与することをお勧めします。定期的に、またアプリケーションに変更があったときに、ジョブの権限を監査することもお勧めします。

ロールベースのアクセスコントロール (RBAC) の許可

管理者は、Amazon Kendra アプリケーションに対するロールベースのアクセスコントロール (RBAC) の許可を厳密に制御する必要があります。

Amazon Kendra でのログ記録とモニタリング

モニタリングは、Amazon Kendra アプリケーションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。Amazon Kendra API の呼び出しをモニタリングするには、を使用できます

AWS CloudTrail。ジョブのステータスをモニタリングするには、Amazon CloudWatch Logs を使用してください。

- Amazon CloudWatch アラーム — CloudWatch アラームを使用して、指定した期間にわたって 1 つのメトリックスを監視します。メトリックスがポリシーを超える場合。 CloudWatch メトリックスが特定の状態にある場合、アラームはアクションを呼び出しません。状態が変わり、それが指定した期間だけ維持される必要があります。詳細については、「Amazon CloudWatch による Amazon Kendra のモニタリング」を参照してください。
- AWS CloudTrail ログ — Amazon Kendra または Amazon Kendra インテリジェントランキングでユーザー、ロール、CloudTrail AWS またはサービスが実行したアクションの記録を提供します。によって収集された情報を使用して CloudTrail、Amazon Kendra に対して行われたリクエスト、リクエストが行われた IP アドレス、リクエストの実行者、実行日時、その他の詳細を判断できます。詳細については、AWS CloudTrail ログでの Amazon Kendra API コールのログ記録およびAWS CloudTrail ログでの Amazon Kendra インテリジェントランキング API コールのログ記録を参照してください。

Amazon Kendra のコンプライアンス検証

Amazon Kendra のセキュリティとコンプライアンスは、Amazon Kendra のさまざまなコンプライアンスプログラムの一環として、サードパーティー監査機関によって評価されます。Amazon Kendra は以下のものに準拠しています。

- Health Insurance Portability and Accountability Act (HIPAA)
- System and Organization Controls (SOC) 2
- Information Security Registered Assessors Program (IRAP)
- 米国東部/西部地域における Federal Risk and Authorization Management Program (FedRAMP) Moderate
- AWS（米国西部）地域における連邦リスクおよび承認管理プログラム GovCloud （FedRAMP）の上位

AWS 特定のコンプライアンスプログラムの対象となるサービスのリストについては、「AWS」を参照してください。一般的な情報については、「AWS」を参照してください。

サードパーティの監査レポートはを使用してダウンロードできます AWS Artifact。詳細については、「Artifact」の「レポートのダウンロード」「 AWS」を参照してください。

Amazon Kendra を使用する際のユーザーのコンプライアンス責任は、ユーザーのデータの機密性や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。 AWS では、コンプライアンスに役立つ以下のリソースを提供しています。

- セキュリティとコンプライアンスのクイックスタートガイドセキュリティとコンプライアンスのクイックスタートガイド 、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境をにデプロイする手順を示します。 AWS
- HIPAA セキュリティとコンプライアンスのためのアーキテクチャに関するホワイトペーパー — このホワイトペーパーでは、企業が HIPAA 準拠のアプリケーションをどのように作成できるかを説明しています。 AWS
- AWS — この一連のワークブックとガイドは、お客様の業界や地域に当てはまる場合があります。
- AWS Config 開発者ガイドのルールによるリソースの評価 — AWS Config このサービスでは、リソース構成が社内の慣行、業界のガイドライン、規制にどの程度準拠しているかを評価します。
- AWS Security Hub AWS —このサービスでは、内部のセキュリティ状態を包括的に把握できるため、 AWS セキュリティ業界の標準やベストプラクティスに準拠しているかどうかを確認できます。

Amazon Kendra の耐障害性

AWS AWS グローバルインフラストラクチャはリージョンとアベイラビリティーゾーンを中心に構築されています。 AWS リージョンには、物理的に分離され隔離された複数のアベイラビリティーゾーンがあり、低レイテンシー、高スループット、冗長性の高いネットワークで接続されています。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョンとアベイラビリティーゾーンの詳細については、「グローバルインフラストラクチャ」を参照してください。AWS

AWS グローバルインフラストラクチャを備えた Amazon Kendra エンタープライズエディションは、耐障害性、スケーラビリティ、高可用性を備えています。インデックスの以前のバージョンへのロールバックは現在サポートされていませんが、既存のデータソースを削除する、およびインデックスに追加することにより、インデックスの一部を更新または再作成することができます。

Amazon Kendra でのインフラストラクチャセキュリティ

マネージド型サービスとして、Amazon Kendra AWS はグローバルネットワークセキュリティによって保護されています。 AWS AWS セキュリティサービスとインフラストラクチャを保護する方法については、「AWS Cloud Security」を参照してください。 AWS 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設計するには、「Security Pillar AWS Well-Architected Framework におけるインフラストラクチャ保護」を参照してください。

AWS 公開されている API 呼び出しを使用して、ネットワーク経由で Amazon Kendra 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、できれば TLS 1.3 が必要です。
- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token Service (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

の設定と脆弱性の分析 AWS Identity and Access Management

AWS ゲストオペレーティングシステム (OS) やデータベースのパッチ、ファイアウォールの設定、障害復旧などの基本的なセキュリティタスクを処理します。これらの手順は適切な第三者によって確認され、証明されています。詳細については、以下のリソース を参照してください。

- 責任共有モデル
- AWS: セキュリティプロセスの概要 (ホワイトペーパー)

以下のリソースは AWS Identity and Access Management (IAM) の設定と脆弱性分析にも対応しています。

- のコンプライアンス検証 AWS Identity and Access Management
- セキュリティのベストプラクティスとユースケースは、を参照してください AWS Identity and Access Management。

のクォータ Amazon Kendra

サポートされるリージョン

AWS 利用可能なリージョンのリストについては、Amazon Web Services 全般のリファレンスの「Amazon Kendra リージョンとエンドポイント」を参照してください。 Amazon Kendra

クォータ

サービスクォータは制限とも呼ばれ、アカウントのサービスリソースの最大数です。 AWS 詳細については、「AWS 全般のリファレンス」の「Amazon Kendra Service Quotas」を参照してください。

インデックスクォータ

説明	デフォルト値	エディション	調整可能
アカウントあたりの最大インデックス数	10	開発者、エンタープライズ	Yes
1 つの単位 (開発者) でインデックス用に抽出されるテキストの量。Developer Edition では、テキスト抽出用のユニットは追加できません。	3 GB	開発者	No
1 つの単位 (Enterprise) でインデックス用に抽出されたテキストの量。Enterprise Edition では、テキストの抽出用に最大 100 ユニットまで追加できます。それ以外の場合はサポートに	30 GB	エンタープライズ	Yes

説明	デフォルト値	エディション	調整可能
お問い合わせください。			

データソースコネクターのクォータ

説明	デフォルト値	エディション	調整可能
インデックス 1 つあたりのデータソースコネクタの最大数 (開発者)	5	開発者	No
インデックス 1 つあたりのデータソースコネクタの最大数 (Enterprise)	50	エンタープライズ	Yes
データソースコネクタを使用する場合の 1 つのドキュメントまたは RAW ファイルの最大サイズ	50 MB	開発者、エンタープライズ	Yes
Amazon S3 データソースコネクタに含まれるアクセスコントロールリスト設定ファイルの S3 プレフィックスの最大数	100	開発者、エンタープライズ	No
Amazon S3 データソースコネクタに含まれるアクセス制御リスト設定ファイルの最大サイズ	50 MB	開発者、エンタープライズ	Yes

よくある質問-クォータ

説明	デフォルト値	エディション	調整可能
インデックスあたりのよくある質問の最大数	30	開発者、エンタープライズ	Yes
1 つのよくある質問の最大サイズ	5 MB	開発者、エンタープライズ	Yes
よくある質問に対して返される結果の最大数。	4	開発者、エンタープライズ	Yes
FAQ の質問に入力できる最大文字数	300	開発者、エンタープライズ	No
よくある質問の回答の最大文字数	2000	開発者、エンタープライズ	No

シソーラスクォータ

説明	デフォルト値	エディション	調整可能
インデックスあたりのシソーラスの最大数	1	開発者、エンタープライズ	No
シソーラスファイルの最大サイズ	5 MB	開発者、エンタープライズ	Yes
シソーラスあたりのシノニムルールの最大数	10,000	開発者、エンタープライズ	Yes
インデックス内のすべてのシソーラスの	10	開発者、エンタープライズ	No

説明	デフォルト値	エディション	調整可能
用語あたりのシノニムの最大数			

Amazon Kendra エクスペリエンスクォータ

説明	デフォルト値	エディション	調整可能
1 Amazon Kendra つのインデックスあたりのエクスペリエンスの最大数	50	開発者、エンタープライズ	Yes

クエリと検索結果のクォータ

説明	デフォルト値	エディション	調整可能
1 つのユニット (Developer) での 1 秒あたりのインデックスクエリ数。Developer Edition では、クエリ用のユニットは追加できません。	0.05	開発者	No
1 単位のインデックスに対する 1 秒あたりのクエリ数 (Enterprise)。Enterprise Edition では、クエリ用に最大 100 ユニットまで追加できます。それ以外の場	0.1	エンタープライズ	Yes

説明	デフォルト値	エディション	調整可能
合はサポートにお問い合わせください。			
クエリテキストあたりの最大文字数	1,000	開発者、エンタープライズ	Yes
クエリあたりの検索結果の最大数。デフォルトは 100 です。100 件を超える結果を指定する場合は、サポートにお問い合わせください。	100	開発者、エンタープライズ	Yes
ページあたりの検索結果の最大数	100	開発者、エンタープライズ	Yes
切り捨て前の、クエリテキストあたりのトークンの最大単語数。デフォルトは 30 です。30 語を超える結果を指定する場合は、サポートにお問い合わせください。	30	開発者、エンタープライズ	Yes
クエリ属性ごとのユーザーグループリストの最大サイズ	10	開発者、エンタープライズ	Yes
クエリ属性ごとの文字列リストの最大サイズ	10	開発者、エンタープライズ	Yes

クエリ、提案、クォータ

説明	デフォルト値	エディション	調整可能
1 回の呼び出しで返されるクエリ候補の最大数 GetQuerySuggestions	10	開発者、エンタープライズ	Yes
1 回の呼び出しで表示されるクエリ候補の最大フィールド/属性数 GetQuerySuggestions	10	開発者、エンタープライズ	Yes
1 回の呼び出しでクエリ候補に追加できるフィールド/属性の最大数 GetQuerySuggestions	5	開発者、エンタープライズ	Yes
インデックスあたりのブロックリストの最大数	1	開発者、エンタープライズ	No
ブロックリストテキストファイルの最大サイズ	2 MB	開発者、エンタープライズ	Yes
ブロックリスト内の項目 (単語または語句) の最大数	20,000	開発者、エンタープライズ	Yes
Query API コールで返されるスペル修正のクエリの提案の最大数。	1	開発者、エンタープライズ	Yes

ドキュメントクォータ

説明	デフォルト値	エディション	調整可能
1 つの単位 (Developer) でインデックス用に抽出されたテキストの量。Developer Edition では、テキスト抽出用のユニットは追加できません。	3 GB	開発者	No
1 つの単位 (Enterprise) でインデックス用に抽出されたテキストの量。Enterprise Edition では、テキストの抽出用に最大 100 ユニットまで追加できます。それ以外の場合はサポートにお問い合わせください。	30 GB	エンタープライズ	Yes
データソースコネクタを使用する場合の 1 つのドキュメントまたは RAW ファイルの最大サイズ	50 MB	開発者、エンタープライズ	Yes
`BatchPutDocument` API を使用する場合の 1 つのドキュメントまたは RAW ファイルの最大サイズ	5 MB	開発者、エンタープライズ	Yes

説明	デフォルト値	エディション	調整可能
1 つのドキュメントから抽出されるテキストの最大量	5 MB	開発者、エンタープライズ	No
インデックス 1 つあたりのカスタムフィールド/属性の最大数	500	開発者、エンタープライズ	No

おすすめの検索結果クォータ

説明	デフォルト値	エディション	調整可能
主要な結果セットあたりの主要なドキュメントの最大数	4	エンタープライズ	Yes
主要な結果セットあたりのクエリテキストの最大数	49	エンタープライズ	No
主要な結果セットのクエリテキストあたりの最大文字数	1,000	エンタープライズ	Yes
インデックスあたりの主要な結果セットの最大数	50	エンタープライズ	Yes

検索結果のクォータの再スコア/再ランク付け

説明	デフォルト値	エディション	調整可能
再スコアリング実行計画または 1 ユニットの容量に対する 1 秒あたりの最大の Rescore リクエスト数。最大 1,000 ユニットの容量が追加できます。	0.01	エンタープライズ	No
アカウントあたりの再スコアリング実行プランの最大数。	50	エンタープライズ	Yes
1 つの Rescore リクエストのドキュメントに対する Title の最大トークン数。	100	エンタープライズ	No
1 つの Rescore リクエストのドキュメントに対する Body の最大トークン数。	200	エンタープライズ	No
1 つの Rescore リクエスト内のドキュメントの最大数。	25	エンタープライズ	No
1 つの Rescore リクエスト内のグループあたりのドキュメントの最大数。	3	エンタープライズ	No

サービスクォータの詳細とクォータの増額リクエストについては、「 Amazon Kendra Service Quotas」を参照してください。

トラブルシューティング

このセクションは、作業中に発生する可能性のある一般的な問題の解決に役立ちます Amazon Kendra。

トピック

データソースのトラブルシューティング

このセクションは、 Amazon Kendra データソースコネクタを設定して使用するときによくある問題の解決に役立ちます。

マイドキュメントにインデックスが作成されませんでした

Amazon Kendra インデックスをデータソースと同期すると、ドキュメントのインデックス作成を妨げる問題が発生する可能性があります。インデックス作成は 2 ステップのプロセスです。まず、データソースは、インデックスを作成する新しいドキュメントと更新されたドキュメントを確認し、インデックスから削除する文書を検出します。次に、ドキュメントレベルで各ドキュメントにアクセスし、インデックスが付けられます。

次のいずれかのステップでエラーが発生する可能性があります。データソースレベルのエラーは、データソースの詳細ページの [Sync run history] (実行履歴を同期) セクションのコンソールに報告されます。同期ジョブのステータスは、[Succeeded] (成功)、[Incomplete] (不完全)、または [Failed] (失敗) になります。ジョブ中にインデックス作成され、削除されたドキュメントの数も確認できます。ステータスが [Failed] (失敗) の場合、メッセージが [Details] (詳細) 列で表示されます。

ドキュメントレベルのエラーはで報告されます。 Amazon CloudWatch Logs CloudWatch コンソールを使用してエラーを確認できます。

ドキュメント同期ステータスレポートを生成するには、「ドキュメントの同期ステータスレポートを生成したい」を参照してください。

同期ジョブが失敗しました

同期ジョブは通常、インデックスまたはデータソースの構成エラーがある場合に失敗します。コンソールで、データソースの詳細ページの [詳細] 列の [実行履歴を同期] セクションにエラーメッセージが表示されます。ドキュメントレベルのエラーは、 Amazon CloudWatch Logsで報告されます。エラーメッセージには、問題点に関する情報が表示されます。問題は通常、 IAM インデックスまたはデータソースに適切な権限がないことです。エラーメッセージには、欠落しているアクセス許可が示されます。受け取る可能性のあるエラーメッセージをいくつか以下に示します。

```
Failed to create log group for job. Please make sure that the IAM role
provided has sufficient permissions.
```

インデックスロールに使用権限がない場合 CloudWatch、 CloudWatch データソースはログを作成できません。このエラーが発生した場合は、 CloudWatch インデックスロールに権限を追加する必要があります。

```
Failed to access Amazon S3 file prefix (bucket name) while trying to crawl
your metadata files. Please make sure the IAM role (ARN) provided has
sufficient permissions.
```

Amazon S3 データソースを使用するときは、 Amazon Kendra ドキュメントを含むバケットにアクセスする権限が必要です。 Amazon Kendra IAM バケットを読み取る権限をデータソースロールに追加する必要があります。

```
The provided IAM role (ARN) could not be assumed. Please make sure Amazon
Kendra is a trusted entity that is allowed to assume the role.
```

Amazon Kendra IAM インデックスロールとデータソースロールを引き受ける権限が必要です。`sts:AssumeRole` アクションの許可を持つロールに信頼ポリシーを追加する必要があります。

IAM Amazon Kendra データソースのインデックスを作成する必要があるポリシーについては、「IAM ロール」を参照してください。

ドキュメント同期ステータスレポートを生成するには、「ドキュメントの同期ステータスレポートを生成したい」を参照してください。

同期ジョブが不完全です

通常、ジョブは、データソースレベルのプロセスを完了したが、ドキュメントレベルのプロセス中に何らかのエラーが発生した場合、不完全になります。ジョブが不完全な場合、ドキュメントの一部で

正常にインデックスが作成されていない可能性があります。 Amazon S3 データソースの場合、不完全なジョブは通常、次の原因によって引き起こされます。

- 1 つまたは複数のドキュメントのメタデータが無効だった。
- インデックス作成のためにドキュメントが送信されたが、少なくとも 1 つのドキュメントが送信されなかった場合。
- インデックスから削除するドキュメントが送信されたが、少なくとも 1 つのドキュメントが送信されなかった場合。

不完全な同期ジョブのトラブルシューティングを行うには、 CloudWatch まずログを調べてください。

1. 詳細列から [詳細を表示] を選択します。 CloudWatch
2. エラーメッセージを確認して、ドキュメントが失敗した原因を確認します。

ドキュメント同期ステータスレポートを生成するには、「ドキュメントの同期ステータスレポートを生成したい」を参照してください。

同期ジョブは成功しましたが、インデックス付きドキュメントがありません

場合によっては、インデックス同期ジョブが [成功] とマークされていても、想定したときに新しいドキュメントや更新されたドキュメントでインデックスが作成されていないことがあります。考えられる理由は以下のとおりです。

- CloudWatch `DocumentsSubmittedForIndexingFailed`メトリクスをチェックして、同期に失敗したドキュメントがないか確認します。 CloudWatch 詳細についてはログを確認してください。
- Amazon S3 データソースには、 Amazon Kendra 間違ったバケット名またはプレフィックスを指定した可能性があります。使用しているバケットが、インデックスを作成するドキュメントを含むバケットであることを確認してください。 Amazon Kendra
- 以前のジョブでインデックス作成に失敗したドキュメントのインデックスを再作成する場合、Amazon Kendra はドキュメントまたは関連するメタデータファイルを変更しない限り、そのドキュメントのインデックスを作成しません。

ドキュメント同期ステータスレポートを生成するには、「ドキュメントの同期ステータスレポートを生成したい」を参照してください。

データソースの同期中にファイル形式の問題が発生しました。

データソースへのファイルの追加、またはデータソースの同期中にファイル形式の問題が発生した場合は、使用しているドキュメントタイプで Amazon Kendra がサポートされていることを確認してください。サポートされているドキュメントタイプのリストについては、 Amazon Kendra 「ドキュメントタイプまたはフォーマット」を参照してください。

プレーンテキストファイルで BatchPutDocument API を使用する場合は、PLAIN_TEXT をコンテンツタイプとして指定してください。

ドキュメントの同期履歴レポートを生成したい

Amazon Kendra データソースコネクタを同期すると、データソース内の各ドキュメントの同期ステータスレポートを生成し、 Amazon Kendra Amazon S3 バケットにコピーできます。このプロセス中、データは AWS KMS キーを使用して暗号化され、ユーザーだけが表示できます。報告されたドキュメントのステータスは、[失敗]、[完了]、または [成功 (エラーあり)] のいずれかになります。

同期ステータスレポートを生成する前に、次の操作を行う必要があります。

- Amazon Kendra Amazon S3 次のサービスプリンシパルをアクセスポリシーに追加します。

```
{
    “Version”: “2012-10-17",
    “Statement”: [
        {
            “Sid”: “KendraS3Access”,
            “Effect”: “Allow”,
            “Principal”: {
                “Service”: “kendra.amazonaws.com”
            },
            “Action”: “s3:PutObject”,
            “Resource”: “arn:aws:s3:::your-manifest-bucket-name/*”
        }
    ]
}
```

- Amazon S3 以下のアクセス権限を持つバケットを作成します。 Amazon Kendra

コンソールを使用して同期ステータスレポートを生成する場合は、[データソースの詳細] ページから [同期履歴の生成] オプションの有効化を選択します。次に、 Amazon S3 バケットの場所を入力し、

使用可能な設定オプションから選択します。レポート生成を有効にすると、次回の同期からレポートが生成されます。

Amazon S3 バケットを削除するとログデータが失われるため、新しい同期レポートを保存する新しいバケットを設定する必要があります。

現在、同期レポートのステータス生成は Amazon S3 コネクタでのみサポートされています。

データソースの同期にはどのくらいの時間がかかりますか?

ドキュメントが更新されていない場合、 Amazon Kendra インデックスの同期時間はドキュメントの数に比例して増加します。例えば、更新されていない 1,000 件のドキュメントの同期には約 5 分かかり、更新されていない 2,000 件のドキュメントの同期には約 10 分かかります。ドキュメントが更新されると、更新されたドキュメントの数に応じて同期時間が長くなります。

データソースの同期にかかる料金はいくらですか?

インデックスを同期すると、ウォームアップして必要な接続が確立されるまで 2 分かかります。 Amazon EC2 この処理中は課金されません。使用量の計測は、同期ジョブが開始してから開始されます。 Amazon Kendra 料金の詳細については、「Amazon Kendra 価格設定」を参照してください。

Amazon EC2 認証エラーが発生します。

仮想プライベートクラウド (VPC) Amazon EC2 データソースの同期中に不正操作エラーが発生した場合、VPC IAM ロールには必要な権限がない可能性があります。 IAM データソースに使用するロールに権限が付与されていることを確認してください。詳細については、「IAM 仮想プライベートクラウドの役割」を参照してください。

Amazon S3 検索インデックスリンクを使用してオブジェクトを開くことができません。

Amazon Kendra インデックスは、 Amazon S3 データソースからアクセス権限を付与されたファイルにのみアクセスできます。たとえば、 Amazon Kendra Amazon S3 オブジェクトをパブリックにするか暗号化するかを決定する権限は変更できません。 Amazon Kendra また、 Amazon S3 オブジェクトの署名付きリンクを作成または返すためのデフォルトの権限もありません。 Amazon S3 Amazon Kendra インデックス内のオブジェクトに対して署名付きリンクを有効にする場合は、次の 2 つの方法があります。

- 検索ページに結果を返す前に、ソース URI オブジェクトでインデックスクエリの結果に署名できます。 step-by-stepこのプロセスの手順については、「署名付き URL を使ったオブジェクトの共有」を参照してください。
- Amazon S3 オブジェクトメタデータのソース URI をオーバーライドして、 CloudFront バケットに接続されたコンテンツ配信ネットワーク (CDN) を通じてサービスを利用できるようにすることができます。 Amazon S3 または、署名済み URL API Gateway を返してリダイレクトするプロキシエンドポイントを使用することもできます。

「SSL AccessDenied 証明書ファイル使用時」というエラーメッセージが表示されます。

データソースで SSL 証明書を使用しているときにアクセス拒否エラーが発生する場合は、指定した場所にある SSL IAM 証明書ファイルにアクセスする権限がロールに付与されていることを確認してください。証明書がキーで暗号化されている場合、 AWS KMS IAM そのキーを使用して復号化する権限もロールに付与する必要があります。 AWS KMS 詳細については、「 AWS KMSに対する認証とアクセスコントロール」を参照してください。

データソースを使用すると認証エラーが発生します。 SharePoint

SharePoint インデックスをデータソースと同期しているときに認証エラーが発生する場合は、にサイト管理者ロールが割り当てられていることを確認してください。 SharePoint

インデックスが Confluence データソースからのドキュメントにクロールされません

Amazon Kendra 同期処理中にインデックスが Confluence データソースからドキュメントをクロールしていない場合は、自分が Confluence の管理者グループに属していることを確認してください。

ドキュメントの検索結果のトラブルシューティング

このセクションは、検索結果の問題の修正に役立ちます。 Amazon Kendra

検索結果が検索クエリと無関係です

検索結果が関係ないと思われる場合は、次の理由が考えられます。

- 結果に信頼度 LOW の結果が含まれています。QueryResultItem's ScoreAttributes フィールドを使用して値がのすべての結果を除外することで、LOW自信を持って結果を除外できますLOW。

Amazon Kendra 各結果に、VERY_HIGHHIGH、MEDIUMのいずれかの信頼度バケット値を割り当てますLOW。これらの値は、結果がクエリに関連しているかどうかの信頼度のレベルを示しています。また、信頼度バケットに関係なく、ANSWER (推奨回答の抜粋)、(FAQ)、QUESTION_ANSWER (文書の抜粋) の順で DOCUMENT 3 Amazon Kendra 種類の結果を返します。そのため、LOW の信頼度の QUESTION_ANSWER 結果が VERY_HIGH の信頼度の DOCUMENT の結果よりも上位に位置する可能性があります。ただし、必ずしも LOW の信頼度の QUESTION_ANSWER が VERY_HIGH の信頼度の DOCUMENT よりも、よい結果になるとは限りません。

- 特定のメタデータフィールドまたは属性は非常に高い値にブーストされ、結果のランキングに影響します。 Amazon Kendra ドキュメントのタイトル、テキスト、日付、カスタムテキストフィールドや属性など、複数のパラメータを使用してインデックスを検索します。すべてのクエリで最良の結果が得られるように、さまざまなブースト値を試すことができます。クエリレベルで動的な関連性のチューニングを使用して、クエリごとに異なるブースト値を使用することもできます。
- ユーザーは情報を検索する際に専門用語を使用していますが、これらの専門用語を処理するためのカスタムシノニムがインデックスに設定されていません。シノニムの使用方法と使用タイミングの詳細については、「Adding custom synonyms to an index」を参照してください。

なぜ 100 件しか表示されないのですか。

Amazon Kendra 関連文書の総数を返します。デフォルトでは、クエリごとに上位 100 件が返されます。結果はページ分割されます。PageNumber を使用して、別のページにアクセスできます。

クエリごとに最大 1,000 件のドキュメントまたは検索結果を返し、1 ページあたり最大 100 件の結果を返すように設定できます Amazon Kendra 。100 件を超える結果を返す場合は、クォータサポートに連絡してリクエストできます。検索結果の数を増やすと、レイテンシーに影響を与える可能性があります。

見ようとしているドキュメントがないのはなぜですか?

Amazon Kendra ユーザーとグループに基づくアクセス制御リスト (ACL) をサポートします。Amazon Kendra ACL ポリシーをコネクタ経由で取り込みます。インデックスで ACL が設定されていない場合、ユーザーとグループの属性フィルターに一致するドキュメントのみが表示されます。ユーザーまたはグループ属性フィルターが指定されている場合、ACL のないドキュメントは表示されません。

トークンベースのアクセス制御を使用している場合、ACL ポリシーのないドキュメントと、ユーザーおよびグループに一致するドキュメントが表示されます。

ACL ポリシーが設定されているドキュメントが表示されるのはなぜですか。

インデックスがアクセスコントロールポリシーを設定しない場合、フィルターによってユーザーとグループを指定できます。ユーザーとグループのフィルターが適用されていない場合は、すべての関連ドキュメントが返されます。ACL ポリシーは無視されます。

一般的な問題のトラブルシューティング

Amazon Kendra CloudWatch メトリクスとログを使用して、データソースの同期に関する洞察を提供します。メトリクスとログを使用して、同期の実行で何が問題になったかを判断し、どのように修正すればよいかを判断できます。

一般的なトラブルシューティングは、 CloudWatch メトリクスから始めてください。

- `DocumentsCrawled` メトリクスをクリックしてデータソースがチェックしたドキュメントの数を確認します。 Amazon S3 バケットの数値が予想よりも少ない場合は、データソースが正しいバケットを指していることを確認してください。
- `DocumentsSkippedNoChange` メトリクスをチェックして、前回の同期以降に変更されていないためにスキップされたドキュメントの数を確認します。数字が想定したものと一致しない場合は、リポジトリが正しく更新されていることを確認してください。
- `DocumentsSkippedInvalidMetadata` メトリクスをチェックして無効なメタデータを含むドキュメントの数を確認します。 CloudWatch ログをチェックして、発生した特定のエラーを確認してください。
- `DocumentsSubmittedForIndexingFailed` メトリクスを確認して、データソースからインデックスに送信されたが、インデックス作成に失敗したドキュメントの数を確認します。例えば、カスタムインデックスフィールドとして定義されていない Amazon S3 データソースでメタデータ属性を使用すると、ドキュメントはインデックスを作成しません。 CloudWatch ログをチェックして、発生した特定のエラーを確認してください。
- `DocumentsSubmittedForDeletionFailed` メトリクスをチェックして、インデックスから削除できなかったドキュメントをデータソースがインデックスから削除しようとしたドキュメントの数を確認します。 CloudWatch ログをチェックして、発生した特定のエラーを確認してください。

CloudWatch 特定の同期実行のログを調べると、実行中に発生したエラーの詳細を確認できます。 CloudWatch の ログの詳細については Amazon Kendra、を参照してくださいCloudWatch Logs。

Amazon Kendra インテリジェントランキング

Amazon Kendra インテリジェントランキングは、Amazon Kendra のセマンティック検索機能を使用して、検索サービスの結果をインテリジェントにランク付けし直します。

トピック

Amazon Kendra セルフマネージド向けのインテリジェント・ランキング OpenSearch

Amazon Kendraのセマンティック検索機能を活用するとOpenSearch、Apache 2.0 ライセンスに基づくセルフマネージドのオープンソース検索サービスからの検索結果を改善できます。 Amazon Kendra インテリジェント・ランキング・プラグインは、を使用して結果をセマンティックに再ランク付けします。 OpenSearch Amazon Kendraこれは、 OpenSearch デフォルトの検索結果からドキュメント本文やタイトルなどの特定のフィールドを使用した検索クエリの意味とコンテキストを理解することで行われます。

例えば、「メインキーノートアドレス」というクエリを考えてみましょう。「住所」には複数の意味があるため、 Amazon Kendra クエリの背後にある意味を推測して、意図した意味に沿った関連情報を返すことができます。このコンテキストでは、これは会議の基調講演です。単純な検索サービスでは、その意図が考慮されず、例えばメインストリートの住所の結果が返される可能性があります。

OpenSearch 用のインテリジェント・ランキング・プラグインは OpenSearch (セルフマネージド) バージョン 2.4.0 以降で使用できます。クイックスタート Bash スクリプトを使用してプラグインをインストールし、Intelligent Ranking プラグインが含まれた新しい Docker イメージを構築できます。 OpenSearch 「インテリジェント検索プラグインの設定」を参照してください - これはすぐに使い始めるためのセットアップの例です。

インテリジェント検索プラグインの仕組み

OpenSearch (セルフマネージド) 用のインテリジェント・ランキング・プラグインの全体的なプロセスは以下の通りです。

1. OpenSearch ユーザーはクエリーを発行し、 OpenSearch クエリーレスポンスまたはクエリーに関連するドキュメントのリストを提供します。

2. インテリジェントランキングプラグインはクエリレスポンスを受け取り、ドキュメントから情報を抽出します。
3. インテリジェント・ランキングプラグインは、 Amazon Kendra インテリジェント・ランキングの Rescore APIを呼び出します。
4. Rescore API はドキュメントから抽出された情報を取得し、検索結果をセマンティックに再度ランク付けします。
5. Rescore API は再ランク付けされた検索結果をプラグインに送り返します。プラグインは、OpenSearch 新しいセマンティックランキングを反映するように検索レスポンス内の検索結果を並べ替えます。

インテリジェントランキングプラグインは、「本文」と「タイトル」フィールドを使用して結果を再度ランク付けします。これらのプラグインフィールドは、 OpenSearch ドキュメントの本文とタイトルの定義に最も適合するインデックス内のフィールドにマップできます。例えば、索引に「chapter_heading」や「chapter_contents」のようなフィールドを含む本の章が含まれている場合、前者を「タイトル」に、後者を「本文」にマッピングすると、最良の結果が得られます。

インテリジェント検索プラグインの設定

Intelligent Ranking プラグインを使って素早く設定 OpenSearch (自己管理) する方法を以下に概説します。

インテリジェント・ランキングプラグイン OpenSearch (クイックセットアップ) を使ったセットアップ (セルフマネージド)

既に Docker イメージを使用している場合はopensearch:2.4.0、この Dockerfile を使用して、インテリジェントランキングプラグインで OpenSearch 2.4.0 の新しいイメージを構築できます。新しいイメージ用のコンテナを docker-compose.yml ファイルまたは opensearch.yml ファイルに含めます。また、リスコア実行プランの作成時に生成されたリスコア実行プラン ID を、リージョンとエンドポイントの情報とともに含めます。リスコア実行プランの作成については、ステップ 2 を参照してください。

2.4.0 より古いバージョンの opensearch Docker イメージを以前にダウンロードしていた場合は、Docker opensearch:2.4.0 イメージ以降を使用し、インテリジェントランキングプラグインが含まれた新しいイメージを構築する必要があります。

1. オペレーティングシステム用の Docker デスクトップ をダウンロードしてインストールします。Docker デスクトップには Docker Compose と Docker Engine が含まれています。お使いの

コンピュータが Docker インストールの詳細に記載されているシステム要件を満たしているかどうかを確認することをお勧めします。

Docker デスクトップの設定でメモリ使用量の要件を増やすこともできます。Docker サービスの無料使用制限以外の Docker の使用要件については、お客様の責任となります。「Docker サブスクリプション」を参照してください。

Docker デスクトップのステータスが「実行中」であることを確認します。

2. Amazon Kendra インテリジェント・ランキングと必要なキャパシティをプロビジョニングします。 Amazon Kendra インテリジェントランキングをプロビジョニングすると、設定したキャパシティーユニットに基づいて時間単位で課金されます。無料利用枠と料金表情報をご覧ください。

   CreateRescoreExecutionPlanAPI を使用してをプロビジョニングしますRescore API。単一ユニットのデフォルトよりも多くのキャパシティーユニットが必要ない場合は、ユニットを追加せず、再スコア実行プランの名前だけを指定してください。UpdateRescoreExecutionPlanAPI を使用してキャパシティ要件を更新することもできます。詳細については、「検索サービスの結果をセマンティックにランク付けする」を参照してください。

   オプションで、ステップ 3 に進んで、クイックスタート Bash スクリプトを実行するときのデフォルトの再スコア実行プランを作成できます。

   ステップ 4 では、レスポンスに含まれる再スコア実行プラン ID をメモしておきます。

   CLI

   ```
   aws kendra-ranking create-rescore-execution-plan \
     --name MyRescoreExecutionPlan \
     --capacity-units '{"RescoreCapacityUnits":<integer number of additional
    capacity units>}'

   Response:

   {
       "Id": "<rescore execution plan ID>",
       "Arn": "arn:aws:kendra-ranking:<region>:<account-id>:rescore-execution-plan/
   <rescore-execution-plan-id>"
   }
   ```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra_ranking = boto3.client("kendra-ranking")

print("Create a rescore execution plan.")

# Provide a name for the rescore execution plan
name = "MyRescoreExecutionPlan"
# Set your required additional capacity units
# Don't set capacity units if you don't require more than 1 unit given by
 default
capacity_units = 1

try:
    rescore_execution_plan_response =
 kendra_ranking.create_rescore_execution_plan(
        Name = name,
        CapacityUnits = {"RescoreCapacityUnits":capacity_units}
    )

    pprint.pprint(rescore_execution_plan_response)

    rescore_execution_plan_id = rescore_execution_plan_response["Id"]

    print("Wait for Amazon Kendra to create the rescore execution plan.")

    while True:
        # Get the details of the rescore execution plan, such as the status
        rescore_execution_plan_description =
 kendra_ranking.describe_rescore_execution_plan(
            Id = rescore_execution_plan_id
        )
        # When status is not CREATING quit.
        status = rescore_execution_plan_description["Status"]
        print(" Creating rescore execution plan. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break
```

```
except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

3. OpenSearchメインブランチのドロップダウンからバージョンブランチを選択して、 GitHub ご使用のバージョンのクイックスタート Bash スクリプトをダウンロードします。

   このスクリプトは、 GitHub スクリプトのリポジトリで選択したバージョンの Docker OpenSearch OpenSearch イメージとダッシュボードを使用します。Intelligent Ranking プラグインの zip ファイルをダウンロードし、 OpenSearch そのプラグインを含む新しい Docker Dockerfile イメージを構築するためのファイルを生成します。また、 OpenSearch インテリジェントランキングプラグインとダッシュボード用のコンテナを含む docker-compose.yml ファイルも作成します。 OpenSearch このスクリプトは、スコア実行プラン ID、リージョン情報、およびエンドポイント (リージョンを使用) を docker-compose.yml ファイルに追加します。その後、スクリプトが実行され、docker-compose upインテリジェント・ランキングとダッシュボードを含むコンテナーが起動します。 OpenSearch OpenSearch コンテナを削除せずに停止するには、docker-compose stop を実行します。コンテナを削除するには、docker-compose down を実行します。

4. ターミナルを開き、Bash スクリプトのディレクトリで、次のコマンドを実行します。

   ```
   bash search_processing_kendra_quickstart.sh -p <execution-plan-id> -r <region>
   ```

   このコマンドを実行するときは、 Amazon Kendra Intelligent Ranking をプロビジョニングしたときにステップ 2 でメモした再スコア実行プラン ID を、地域情報とともに提供します。オプションで、代わりに --create-execution-plan オプションを使用して Amazon Kendra インテリジェントランキングをプロビジョニングすることもできます。これにより、デフォルトの名前とデフォルトのキャパシティーで再スコア実行プランが作成されます。

   デフォルトのエフェメラルコンテナが削除されてもインデックスが失われないようにするには、--volume-name オプションを使用してデータボリューム名を指定して、実行後もインデックスを保持できます。以前にインデックスを作成している場合は、docker-compose.yml ファイルまたは opensearch.yml ファイルでボリュームを指定できます。ボリュームをそのまま残すには、docker-compose down -v を実行しないでください。

   クイックスタート Bash スクリプトは、Intelligent Ranking AWS OpenSearch に接続するようにキーストア内の認証情報を設定します。 Amazon Kendra AWS 認証情報をスクリプトに提供

するには、--profileオプションを使用してプロファイルを指定します。 AWS --profileオプションが指定されていない場合、クイックスタート Bash AWS スクリプトは認証情報 (アクセス/シークレットキー、オプションのセッショントークン) を環境変数から読み取り、次にデフォルトプロファイルから読み取ろうとします。 AWS --profileオプションが指定されておらず、認証情報も見つからない場合、スクリプトは認証情報をキーストアに渡しません。 OpenSearch OpenSearch キーストアに認証情報が指定されていない場合でも、プラグインは Default Credential Provider Chain 内の認証情報 (Amazon ECS メタデータサービスを通じて配信されるコンテナ認証情報やインスタンスプロファイル認証情報など) をチェックします Amazon EC2 。

Intelligent Ranking IAM Amazon Kendra を起動するのに必要な権限を持つロールを作成したことを確認してください。以下は、特定の再スコア実行プランに Rescore API IAM を使用する権限を付与するポリシーの例です。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "kendra-ranking:Rescore",
            "Resource": "arn:aws:kendra-ranking:${Region}:${Account}:rescore-
execution-plan/${RescoreExecutionPlanId}"
        }
    ]
}
```

docker-compose.yml の例

OpenSearch 2.4.0以降とインテリジェントランキングプラグインとダッシュボード2.4.0以降を使用したdocker-compose.ymlファイルの例。 OpenSearch

```
version: '3'
networks:
  opensearch-net:
volumes:
  <volume-name>:
services:
  opensearch-node:
    image: <Docker image tag name of OpenSearch with Intelligent Ranking plugin>
    container_name: opensearch-node
```

```
    environment:
      - cluster.name=opensearch-cluster
      - node.name=opensearch-node
      - discovery.type=single-node
      - kendra_intelligent_ranking.service.endpoint=https://kendra-
ranking.<region>.api.aws
      - kendra_intelligent_ranking.service.region=<region>
      - kendra_intelligent_ranking.service.execution_plan_id=<rescore-execution-plan-
id>
    ulimits:
      memlock:
        soft: -1
        hard: -1
      nofile:
        soft: 65536
        hard: 65536
    ports:
      - 9200:9200
      - 9600:9600
    networks:
      - opensearch-net
  volumes:
    <docker-volume-name>:/usr/share/opensearch/data
  opensearch-dashboard:
   image: opensearchproject/opensearch-dashboards:<your-version>
   container_name: opensearch-dashboards
   ports:
     - 5601:5601
   environment:
     OPENSEARCH_HOSTS: '["https://opensearch-node:9200"]'
   networks:
     - opensearch-net
```

Dockerfile とイメージの構築の例

インテリジェント・ランキング・プラグインで 2.4.0 以降を使用する場合の例。Dockerfile OpenSearch

```
FROM opensearchproject/opensearch:<your-version>
RUN /usr/share/opensearch/bin/opensearch-plugin install --batch  https://github.com/
opensearch-project/search-processor/releases/download/<your-version>/search-
processor.zip
```

OpenSearch インテリジェント・ランキング・プラグインを使用した Docker イメージの構築。

```
docker build --tag=<Docker image tag name of OpenSearch with Intelligent Ranking
 plugin>
```

インテリジェント検索プラグインとのやり取り

Intelligent Ranking プラグインを使用してセットアップ OpenSearch (自己管理) すると、curl OpenSearch コマンドまたはクライアントライブラリを使用してプラグインを操作できます。Intelligent Ranking OpenSearch プラグインでアクセスするためのデフォルトの認証情報は、ユーザー名「admin」、パスワード「admin」です。

インテリジェント・ランキングプラグインの設定をインデックスに適用するには OpenSearch :

Curl

```
curl -XPUT "https://localhost:9200/<your-docs-index>/_settings" -u 'admin:admin' --
insecure -H 'Content-Type: application/json' -d'
{
  "index": {
    "plugin" : {
      "searchrelevance" : {
        "result_transformer" : {
          "kendra_intelligent_ranking": {
              "order": 1,
              "properties": {
                "title_field": "title_field_name_here",
                "body_field": "body_field_name_here"
              }
          }
        }
      }
    }
  }
}
'
```

Python

```
pip install opensearch-py

from opensearchpy import OpenSearch
```

```
host = 'localhost'
port = 9200
auth = ('admin', 'admin')

client = OpenSearch(
    hosts = [{'host': host, 'port': port}],
    http_compress = True, # enables gzip compression for request bodies
    http_auth = auth,
    # client_cert = client_cert_path,
    # client_key = client_key_path,
    use_ssl = True,
    verify_certs = False,
    ssl_assert_hostname = False,
    ssl_show_warn = False,
    ca_certs = ca_certs_path
)

setting_body = {
    "index": {
        "plugin" : {
            "searchrelevance" : {
                "result_transformer" : {
                    "kendra_intelligent_ranking": {
                            "order": 1,
                            "properties": {
                                "title_field": "title_field_name_here",
                                "body_field": "body_field_name_here"
                            }
                    }
                }
            }
        }
    }
}

response = client.indices.put_settings(index_name, body=setting_body)
```

ドキュメント本文やドキュメントコンテンツフィールドなど、再ランク付けに使用するメインテキストフィールドの名前を含める必要があります。ドキュメントのタイトルや概要など、他のテキストフィールドも含めることができます。

これで、任意のクエリを発行できるようになり、その結果はインテリジェントランキングプラグインを使ってランク付けされます。

Curl

```
curl -XGET "https://localhost:9200/<your-docs-index>/_search?pretty" -u
 'admin:admin' --insecure -H 'Content-Type: application/json' -d'
{
  "query" : {
    "match" : {
      "body_field_name_here": "intelligent systems"
    }
  }
}
'
```

Python

```
from opensearchpy import OpenSearch
host = 'localhost'
port = 9200
auth = ('admin', 'admin')

client = OpenSearch(
    hosts = [{'host': host, 'port': port}],
    http_compress = True, # enables gzip compression for request bodies
    http_auth = auth,
    # client_cert = client_cert_path,
    # client_key = client_key_path,
    use_ssl = True,
    verify_certs = False,
    ssl_assert_hostname = False,
    ssl_show_warn = False,
    ca_certs = ca_certs_path
)

query = {
  'size': 10,
  "query" : {
    "match" : {
      "body_field_name_here": "intelligent systems"
    }
  }
```

```
}

response = client.search(
    body = query,
    index = index_name
)

print('\nSearch results:')
print(response)
```

OpenSearch インデックスのインテリジェント・ランキングプラグイン設定を削除するには:

Curl

```
curl -XPUT "http://localhost:9200/<your-docs-index>/_settings" -H 'Content-Type:
 application/json' -d'
{
  "index": {
    "plugin": {
      "searchrelevance": {
        "result_transformer": {
          "kendra_intelligent_ranking.*": null
        }
      }
    }
  }
}
'
```

Python

```
from opensearchpy import OpenSearch
host = 'localhost'
port = 9200
auth = ('admin', 'admin')

client = OpenSearch(
    hosts = [{'host': host, 'port': port}],
    http_compress = True, # enables gzip compression for request bodies
    http_auth = auth,
    # client_cert = client_cert_path,
    # client_key = client_key_path,
```

```
    use_ssl = True,
    verify_certs = False,
    ssl_assert_hostname = False,
    ssl_show_warn = False,
    ca_certs = ca_certs_path
)

setting_body = {
  "index": {
    "plugin": {
      "searchrelevance": {
        "result_transformer": {
          "kendra_intelligent_ranking.*": null
        }
      }
    }
  }
}

response = client.indices.put_settings(index_name, body=setting_body)
```

インテリジェントランキングプラグインを特定のクエリでテストしたり、特定の本文やタイトルフィールドでテストするには:

Curl

```
curl -XGET "https://localhost:9200/<your-docs-index>/_search?pretty" -u
 'admin:admin' --insecure -H 'Content-Type: application/json' -d'
{
  "query": {
    "multi-match": {
      "query": "intelligent systems",
      "fields": ["body_field_name_here", "title_field_name_here"]
    }
  },
  "size": 25,
  "ext": {
    "search_configuration": {
      "result_transformer": {
        "kendra_intelligent_ranking": {
          "order": 1,
          "properties": {
```

```
            "title_field": "title_field_name_here",
            "body_field": "body_field_name_here"
          }
        }
      }
    }
  }
}
'
```

Python

```
from opensearchpy import OpenSearch
host = 'localhost'
port = 9200
auth = ('admin', 'admin')

client = OpenSearch(
    hosts = [{'host': host, 'port': port}],
    http_compress = True, # enables gzip compression for request bodies
    http_auth = auth,
    # client_cert = client_cert_path,
    # client_key = client_key_path,
    use_ssl = True,
    verify_certs = False,
    ssl_assert_hostname = False,
    ssl_show_warn = False,
    ca_certs = ca_certs_path
)

# Index settings null for kendra_intelligent_ranking

query = {
  "query": {
    "multi_match": {
      "query": "intelligent systems",
      "fields": ["body_field_name_here", "title_field_name_here"]
    }
  },
  "size": 25,
  "ext": {
    "search_configuration": {
      "result_transformer": {
```

```
        "kendra_intelligent_ranking": {
          "order": 1,
          "properties": {
            "title_field": "title_field_name_here",
            "body_field": "body_field_name_here"
          }
        }
      }
    }
  }
}

response = client.search(
    body = query,
    index = index_name
)

print('\nSearch results:')
print(response)
```

OpenSearch Amazon Kendra 結果と結果の比較

ランク付けされた結果と、再ランク付けされた結果を比較 side-by-side OpenSearch (自己管理) できます。 Amazon Kendra OpenSearch バージョン 2.4.0 side-by-side 以降のダッシュボードでは結果が表示されるため、 OpenSearch ドキュメントのランク付け方法と検索クエリでのドキュメントのランク付け方法を比較したり、 Amazon Kendra プラグインがドキュメントをランク付けしたりできるようになります。

OpenSearch Amazon Kendra ランク付けされた結果と再ランク付けされた結果を比較する前に、OpenSearch ダッシュボードが Intelligent Ranking OpenSearch プラグインを備えたサーバーによって支えられていることを確認してください。これは Docker とクイックスタート Bash スクリプトを使用して設定できます。インテリジェント検索プラグインの設定 を参照してください。

以下では、 OpenSearch Amazon Kendra ダッシュボードで結果を比較および検索する方法の概要を説明します。 OpenSearch 詳細については、ドキュメンテーションを参照してください。OpenSearch

OpenSearch ダッシュボードの検索結果の比較

1. http://localhost:5601 OpenSearch を開いてダッシュボードにサインインします。デフォルトの認証情報は、ユーザー名は「admin」、パスワードは「admin」です。

2. OpenSearch ナビゲーションメニューのプラグインから [検索関連性] を選択します。
3. 検索バーに検索テキストを入力します。
4. クエリ 1 のインデックスを選択し、クエリ DSL OpenSearch にクエリを入力します。この `%SearchText%` 変数を使用して、検索バーに入力した検索テキストを参照できます。このクエリの例については、「OpenSearch ドキュメント」を参照してください。このクエリで返される結果は、Intelligent Ranking OpenSearch プラグインを使用しない結果です。
5. クエリ 2 に同じインデックスを選択し、同じクエリを OpenSearch Query DSL に入力します。また、`kendra_intelligent_ranking` を持つ拡張子を含め、ランク付けに必須の `body_field` を指定します。タイトルフィールドも指定できますが、本文フィールドは必須です。このクエリの例については、「OpenSearch ドキュメント」を参照してください。このクエリで返される結果は、Intelligent Ranking Amazon Kendra プラグインを使用して再ランク付けされた結果です。このプラグインは最大 25 件の結果をランク付けします。
6. [検索] を選択すると、結果を返して比較できます。

検索サービスの結果をセマンティックにランク付けする

Amazon Kendra Amazon Kendraインテリジェント・ランキングはセマンティック検索機能を使用して、検索サービスの結果をランク付けし直します。これは、検索クエリのコンテキストと、検索サービスのドキュメントから入手可能なすべての情報を考慮して行われます。 Amazon Kendra インテリジェント・ランキングを使うと、単純なキーワードマッチングを改善できます。

CreateRescoreExecutionPlanこの API は Rescore API Amazon Kendra のプロビジョニングに使用されるインテリジェント・ランキングリソースを作成します。RescoreAPI は OpenSearch (セルフマネージド) などの検索サービスからの検索結果を再ランク付けします。

`CreateRescoreExecutionPlan` の呼び出し時に、検索サービスの結果を再ランク付けするのに必要なキャパシティーユニットを設定します。単一ユニットのデフォルトを超えるキャパシティーユニットが必要ない場合は、デフォルトを変更しないでください。再スコア実行プランの名前だけを指定してください。追加ユニットは 1000 件まで設定できます。単一のキャパシティーユニットに含まれる内容については、「キャパシティーの調整」を参照してください。 Amazon Kendra Intelligent Ranking をプロビジョニングすると、設定したキャパシティユニットに基づいて時間単位で課金されます。無料利用枠と料金表情報をご覧ください。

`CreateRescoreExecutionPlan` を呼び出すと、再スコア実行プラン ID が生成され、レスポンスとして返されます。`Rescore` API は再スコア実行プラン ID を使用して、設定したキャパシティーを使用して検索サービスの結果を再ランク付けします。再スコア実行プラン ID は検索サービスの設定ファイルに含めます。たとえば、 OpenSearch (セルフマネージド) を使用する場合は、docker-

compose.yml ファイルまたは opensearch.yml ファイルに再スコア実行プラン ID を含めます。「(セルフサービス) 結果のインテリジェントなランク付け」を参照してください。 OpenSearch

Amazon リソースネーム (ARN) は、CreateRescoreExecutionPlan 呼び出し時のレスポンスでも生成されます。この ARN を使用して AWS Identity and Access Management (IAM) にアクセス権限ポリシーを作成し、特定の再スコア実行プランの特定の ARN へのユーザーアクセスを制限できます。特定の再スコア実行プランで Rescore API IAM を使用する権限を付与するポリシーの例については、「自己管理型の Amazon Kendra Intelligent Ranking」を参照してください。 OpenSearch

以下は、キャパシティーユニットを 1 に設定して再スコア実行プランを作成する例です。

CLI

```
aws kendra-ranking create-rescore-execution-plan \
  --name MyRescoreExecutionPlan \
  --capacity-units '{"RescoreCapacityUnits":1}'

Response:

{
    "Id": "<rescore execution plan ID>",
    "Arn": "arn:aws:kendra-ranking:<region>:<account-id>:rescore-execution-plan/
<rescore-execution-plan-id>"
}
```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra_ranking = boto3.client("kendra-ranking")

print("Create a rescore execution plan.")

# Provide a name for the rescore execution plan
name = "MyRescoreExecutionPlan"
# Set your required additional capacity units
# Don't set capacity units if you don't require more than 1 unit given by default
capacity_units = 1
```

```
try:
    rescore_execution_plan_response = kendra_ranking.create_rescore_execution_plan(
        Name = name,
        CapacityUnits = {"RescoreCapacityUnits":capacity_units}
    )

    pprint.pprint(rescore_execution_plan_response)

    rescore_execution_plan_id = rescore_execution_plan_response["Id"]

    print("Wait for Amazon Kendra to create the rescore execution plan.")

    while True:
        # Get the details of the rescore execution plan, such as the status
        rescore_execution_plan_description =
 kendra_ranking.describe_rescore_execution_plan(
            Id = rescore_execution_plan_id
        )
        # When status is not CREATING quit.
        status = rescore_execution_plan_description["Status"]
        print(" Creating rescore execution plan. Status: "+status)
        time.sleep(60)
        if status != "CREATING":
            break

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

```
import java.util.concurrent.TimeUnit;

import software.amazon.awssdk.services.kendraranking.KendraRankingClient;
import
 software.amazon.awssdk.services.kendraranking.model.CapacityUnitsConfiguration;
import
 software.amazon.awssdk.services.kendraranking.model.CreateRescoreExecutionPlanRequest;
import
 software.amazon.awssdk.services.kendraranking.model.CreateRescoreExecutionPlanResponse;
import
 software.amazon.awssdk.services.kendraranking.model.DescribeRescoreExecutionPlanRequest;
```

```
import
 software.amazon.awssdk.services.kendraranking.model.DescribeRescoreExecutionPlanResponse;
import
 software.amazon.awssdk.services.kendraranking.model.RescoreExecutionPlanStatus;

public class CreateRescoreExecutionPlanExample {

  public static void main(String[] args) throws InterruptedException {

    String rescoreExecutionPlanName = "MyRescoreExecutionPlan";
    int capacityUnits = 1;

    KendraRankingClient kendraRankingClient = KendraRankingClient.builder().build();

    System.out.println(String.format("Creating a rescore execution plan named %s",
 rescoreExecutionPlanName));

    CreateRescoreExecutionPlanResponse createResponse =
 kendraRankingClient.createRescoreExecutionPlan(
        CreateRescoreExecutionPlanRequest.builder()
            .name(rescoreExecutionPlanName)
            .capacityUnits(
                CapacityUnitsConfiguration.builder()
                    .rescoreCapacityUnits(capacityUnits)
                    .build()
            )
            .build()
    );

    String rescoreExecutionPlanId = createResponse.id();
    System.out.println(String.format("Waiting for rescore execution plan with id %s
 to finish creating.", rescoreExecutionPlanId));
    while (true) {
      DescribeRescoreExecutionPlanResponse describeResponse =
 kendraRankingClient.describeRescoreExecutionPlan(
          DescribeRescoreExecutionPlanRequest.builder()
              .id(rescoreExecutionPlanId)
              .build()
      );
      RescoreExecutionPlanStatus rescoreExecutionPlanStatus =
 describeResponse.status();
      if (rescoreExecutionPlanStatus != RescoreExecutionPlanStatus.CREATING) {
        break;
      }
```

```
      TimeUnit.SECONDS.sleep(60);
    }

    System.out.println("Rescore execution plan creation is complete.");
  }
}
```

以下は、キャパシティーユニットを 2 に設定するように再スコア実行プランを更新する例です。

CLI

```
aws kendra-ranking update-rescore-execution-plan \
  --id <rescore execution plan ID> \
  --capacity-units '{"RescoreCapacityUnits":2}'
```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint
import time

kendra_ranking = boto3.client("kendra-ranking")

print("Update a rescore execution plan.")

# Provide the ID of the rescore execution plan
id = <rescore execution plan ID>
# Re-set your required additional capacity units
capacity_units = 2

try:
    kendra_ranking.update_rescore_execution_plan(
        Id = id,
        CapacityUnits = {"RescoreCapacityUnits":capacity_units}
    )

    print("Wait for Amazon Kendra to update the rescore execution plan.")

    while True:
        # Get the details of the rescore execution plan, such as the status
```

```
        rescore_execution_plan_description =
 kendra_ranking.describe_rescore_execution_plan(
            Id = id
        )
        # When status is not UPDATING quit.
        status = rescore_execution_plan_description["Status"]
        print(" Updating rescore execution plan. Status: "+status)
        time.sleep(60)
        if status != "UPDATING":
            break

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

```
import java.util.concurrent.TimeUnit;

import software.amazon.awssdk.services.kendraranking.KendraRankingClient;
import
 software.amazon.awssdk.services.kendraranking.model.CapacityUnitsConfiguration;
import
 software.amazon.awssdk.services.kendraranking.model.DescribeRescoreExecutionPlanRequest;
import
 software.amazon.awssdk.services.kendraranking.model.DescribeRescoreExecutionPlanResponse;
import
 software.amazon.awssdk.services.kendraranking.model.RescoreExecutionPlanStatus;
import
 software.amazon.awssdk.services.kendraranking.model.UpdateRescoreExecutionPlanRequest;
import
 software.amazon.awssdk.services.kendraranking.model.UpdateRescoreExecutionPlanResponse;

public class UpdateRescoreExecutionPlanExample {

  public static void main(String[] args) throws InterruptedException {

    String rescoreExecutionPlanId = <rescore execution plan ID>;
    int newCapacityUnits = 2;

    KendraRankingClient kendraRankingClient = KendraRankingClient.builder().build();
```

```
    System.out.println(String.format("Updating a rescore execution plan named %s",
 rescoreExecutionPlanId));

    UpdateRescoreExecutionPlanResponse updateResponse =
 kendraRankingClient.updateRescoreExecutionPlan(
        UpdateRescoreExecutionPlanRequest.builder()
            .id(rescoreExecutionPlanId)
            .capacityUnits(
                CapacityUnitsConfiguration.builder()
                    .rescoreCapacityUnits(newCapacityUnits)
                    .build()
            )
            .build()
    );

    System.out.println(String.format("Waiting for rescore execution plan with id %s
 to finish updating.", rescoreExecutionPlanId));
    while (true) {
      DescribeRescoreExecutionPlanResponse describeResponse =
 kendraRankingClient.describeRescoreExecutionPlan(
          DescribeRescoreExecutionPlanRequest.builder()
              .id(rescoreExecutionPlanId)
              .build()
      );
      RescoreExecutionPlanStatus rescoreExecutionPlanStatus =
 describeResponse.status();
      if (rescoreExecutionPlanStatus != RescoreExecutionPlanStatus.UPDATING) {
        break;
      }
      TimeUnit.SECONDS.sleep(60);
    }

    System.out.println("Rescore execution plan update is complete.");
  }
}
```

以下は Rescore API を使用したスクリプトの例です。

CLI

```
aws kendra-ranking rescore \
  --rescore-execution-plan-id <rescore execution plan ID> \
```

```
  --search-query "intelligent systems" \
  --documents "[{\"Id\": \"DocId1\",\"Title\": \"Smart systems\", \"Body\":
 \"intelligent systems in everyday life\",\"OriginalScore\": 2.0}, {\"Id\":
 \"DocId2\",\"Title\": \"Smarter systems\", \"Body\": \"living with intelligent
 systems\",\"OriginalScore\": 1.0}]"
```

Python

```
import boto3
from botocore.exceptions import ClientError
import pprint

kendra_ranking = boto3.client("kendra-ranking")

print("Use the Rescore API.")

# Provide the ID of the rescore execution plan
id = <rescore execution plan ID>
# The search query from the search service
query = "intelligent systems"
# The list of documents for Intelligent Ranking to rescore
document_list = [
    {"Id": "DocId1", "Title": "Smart systems", "Body": "intelligent systems in
 everyday life", "OriginalScore": 2.0},
    {"Id": "DocId2", "Title": "Smarter systems", "Body": "living with intelligent
 systems", "OriginalScore": 1.0}
]

try:
    rescore_response = kendra_ranking.rescore(
        rescore_execution_plan_id = id,
        search_query = query,
        documents = document_list
    )

    print(rescore_response["RescoreId"])
    print(rescore_resposne["ResultItems"])

except ClientError as e:
        print("%s" % e)

print("Program ends.")
```

Java

```
import java.util.ArrayList;
import java.util.List;

import software.amazon.awssdk.services.kendraranking.KendraRankingClient;
import software.amazon.awssdk.services.kendraranking.model.RescoreRequest;
import software.amazon.awssdk.services.kendraranking.model.RescoreResponse;
import software.amazon.awssdk.services.kendraranking.model.Document;

public class RescoreExample {

  public static void main(String[] args) {

    String rescoreExecutionPlanId = <rescore execution plan ID>;
    String query = "intelligent systems";

    List<Document> documentList = new ArrayList<>();
    documentList.add(
        Document.builder()
            .id("DocId1")
            .originalScore(2.0F)
            .body("intelligent systems in everyday life")
            .title("Smart systems")
            .build()
    );
    documentList.add(
        Document.builder()
            .id("DocId2")
            .originalScore(1.0F)
            .body("living with intelligent systems")
            .title("Smarter systems")
            .build()
    );

    KendraRankingClient kendraRankingClient = KendraRankingClient.builder().build();

    RescoreResponse rescoreResponse = kendraRankingClient.rescore(
        RescoreRequest.builder()
            .rescoreExecutionPlanId(rescoreExecutionPlanId)
            .searchQuery(query)
            .documents(documentList)
            .build()
    );
```

```
    System.out.println(rescoreResponse.rescoreId());
    System.out.println(rescoreResponse.resultItems());
  }
}
```

のドキュメント履歴 Amazon Kendra

- ドキュメントの最終更新日： 2024 年 2 月 27 日

次の表に、 の各リリースにおける重要な変更点を示します Amazon Kendra。このドキュメントの更新に関する通知については、RSS フィードにサブスクライブできます。

変更	説明	日付
新機能	Amazon Kendra は、 GitHub データソースコネクタの更新バージョンをサポートするようになりました。詳細については、「」を参照してくださいGitHub。	2024 年 2 月 27 日
新機能	Amazon Kendra は、データソースコネクタの更新 Amazon FSx バージョンをサポートするようになりました。詳細については、「Amazon FSx (Windows)」およびAmazon FSx 「 (NetApp ONTAP)」を参照してください。	2024 年 2 月 8 日
新機能	Amazon Kendra が Slack データソースコネクタの更新バージョンをサポートするようになりました。詳細については、「Slack」を参照してください。	2024 年 1 月 11 日
新機能	Amazon Kendra は、検索結果の折りたたみと拡張をサポートするようになりました。詳細については、「クエリ結果	2023 年 10 月 19 日

	の折りたたみ/展開」を参照してください。	
新機能	Amazon Kendra が Aurora (MySQL) データソースコネクタをサポートするようになりました。詳細については、「Aurora (MySQL)」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Aurora (PostgreSQL) データソースコネクタをサポートするようになりました。詳細については、「Aurora (PostgreSQL)」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Amazon RDS (MySQL) データソースコネクタをサポートするようになりました。詳細については、「Amazon RDS (MySQL)」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Amazon RDS (Microsoft SQL Server) データソースコネクタをサポートするようになりました。詳細については、「Amazon RDS (Microsoft SQL Server)」を参照してください。	2023 年 9 月 28 日

新機能	Amazon Kendra が Amazon RDS (Oracle) データソースコネクタをサポートするようになりました。詳細については、「Amazon RDS (Oracle)」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Amazon RDS (PostgreSQL) データソースコネクタをサポートするようになりました。詳細については、「Amazon RDS (PostgreSQL)」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が IBM DB2 データソースコネクタをサポートするようになりました。詳細については、「IBM DB2」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Microsoft SQL Server データソースコネクタをサポートするようになりました。詳細については、「Microsoft SQL Server」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が MySQL データソースコネクタをサポートするようになりました。詳細については、「MySQL」を参照してください。	2023 年 9 月 28 日

新機能	Amazon Kendra が Oracle Database データソースコネクタをサポートするようになりました。詳細については、「Oracle Database」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が PostgreSQL データソースコネクタをサポートするようになりました。詳細については、「PostgreSQL」を参照してください。	2023 年 9 月 28 日
新機能	Amazon Kendra が Drupal のデータソースコネクタを提供するようになりました。詳細については、「Drupal」を参照してください。	2023 年 9 月 6 日
新機能	検索拡張生成 (RAG) システム用の Amazon Kendra Retrieve API を使用して、意味的に関連するパッセージを取得します。	2023 年 6 月 22 日
新機能	Amazon Kendra が Web Crawler データソースコネクタの更新バージョン Amazon Kendra をサポートするようになりました。詳細については、「Amazon Kendra Web Crawler v2.0」を参照してください。	2023 年 6 月 21 日

リージョンの拡張	Amazon Kendra が欧州 (ロンドン) (eu-west-2) で利用可能になりました。	2023 年 6 月 5 日
新機能	Amazon Kendra が Alfresco データソースコネクタの更新バージョンをサポートするようになりました。詳細については、「Alfresco」を参照してください。	2023 年 5 月 16 日
新機能	Amazon Kendra が Adobe Experience Manager のデータソースコネクタを提供するようになりました。詳細については、「Adobe Experience Manager」を参照してください。	2023 年 5 月 11 日
新機能	Amazon Kendra では、 を呼び出すときのドキュメントフィールド/属性の設定がサポートされるようになりましたGetQuerySuggestions。ドキュメントフィールドの内容をクエリの提案の基準にできるようになりました。詳細については、「Query suggestions」を参照してください。	2023 年 5 月 2 日
新機能	Amazon Kendra が Gmail のデータソースコネクタを提供するようになりました。詳細については、「Gmail」を参照してください。	2023 年 4 月 13 日

新機能	Amazon Kendra は、Microsoft OneDrive データソースコネクタの更新バージョンをサポートするようになりました。詳細については、「Microsoft OneDrive v2.0」を参照してください。	2023 年 4 月 3 日
新機能	ユーザーが注目結果を使用して特定のクエリを入力したとき、新しいドキュメントの視認性を向上させたり、特定のドキュメントを昇格したりします。	2023 年 3 月 30 日
新機能	Amazon Kendra は、Microsoft の更新されたデータソースコネクタをサポートするようになりました SharePoint。詳細については、「Microsoft SharePoint」を参照してください。	2023 年 3 月 2 日
新機能	Amazon Kendra は Confluence データソースコネクタの更新バージョンをサポートするようになりました。詳細については、「Confluence」を参照してください。	2023 年 3 月 1 日
リージョンの拡張	Amazon Kendra がアジアパシフィック (東京) (ap-northeast-1) で利用可能になりました。	2023 年 2 月 7 日

新機能	Amazon Kendra が Microsoft Exchange のデータソースコネクタを提供するようになりました。詳細については、「Microsoft Exchange」を参照してください。	2023 年 1 月 12 日
新機能	Amazon Kendra が Microsoft Yammer のデータソースコネクタを提供するようになりました。詳細については、「Microsoft Yammer」を参照してください。	2023 年 1 月 12 日
新機能	Amazon Kendra は、RTF、XML、XSLT、MS_EXCEL、CSV、JSON、MD ドキュメントタイプのインデックス作成をサポートするようになりました。詳細については、「Types of documents」を参照してください。	2023 年 1 月 11 日
新機能	Amazon Kendra は、データソースコネクタの更新バージョン Amazon S3 をサポートするようになりました。詳細については、「Amazon S3」を参照してください。	2023 年 1 月 10 日
新機能	OpenSearch (セルフマネージド) 検索結果は、Amazon Kendra インテリジェントランキングを使用して意味的にランク付けできます。	2023 年 1 月 9 日

新機能	Amazon Kendra が Microsoft Teams のデータソースコネクタを提供するようになりました。詳細については、「Microsoft Teams」を参照してください。	2023 年 1 月 5 日
新機能	Amazon Kendra には、Google Drive 用の更新されたデータソースコネクタがあります。詳細については、「Google Drive」を参照してください。	2023 年 1 月 5 日
新機能	Amazon Kendra には、 用の更新されたデータソースコネクタがあります ServiceNow。詳細については、「」を参照してくださいServiceNow。	2022 年 12 月 21 日
新機能	Amazon Kendra には、Salesforce 用の更新されたデータソースコネクタがあります。詳細については、「Salesforce」を参照してください。	2022 年 12 月 21 日
リージョンの拡張	Amazon Kendra がアジアパシフィック (ムンバイ) (ap-south-1) で利用可能になりました。	2022 年 12 月 14 日
新機能	Amazon Kendra の表形式検索機能では、HTML ドキュメントに埋め込まれた表から回答を検索して抽出できます。	2022 年 11 月 27 日
新機能	Amazon Kendra は、選択した言語セットのセマンティック検索をサポートします。	2022 年 11 月 27 日

新機能	Amazon Kendra が Dropbox のデータソースコネクタを提供するようになりました。詳細については、「Dropbox」を参照してください。	2022 年 9 月 27 日
新機能	Amazon Kendra が Zendesk のデータソースコネクタを提供するようになりました。詳細については、「Zendesk」を参照してください。	2022 年 8 月 17 日
新機能	ドキュメントレベルのアクセス制御は、ドキュメントにインデックスを付けた後に再設定できるようになりました。詳細については、「アクセスコントロール設定」を参照してください。	2022 年 7 月 14 日
新機能	Amazon Kendra が Alfresco のデータソースコネクタを提供するようになりました。詳細については、「Alfresco」を参照してください。	2022 年 6 月 30 日
新機能	Amazon Kendra が のデータソースコネクタを提供するようになりました GitHub。詳細については、「」を参照してくださいGitHub。	2022 年 6 月 2 日
新機能	Amazon Kendra が Jira のデータソースコネクタを提供するようになりました。詳細については、「Jira」を参照してください。	2022 年 5 月 12 日

新機能	ファセット内のネストされたファセットを検索結果に表示できます。詳細については、「Facets」を参照してください。	2022 年 5 月 5 日
新機能	Amazon Kendra が Quip 用のデータソースコネクタを提供するようになりました。詳細については、「Quip」を参照してください。	2022 年 4 月 19 日
新機能	Amazon Kendra が Box のデータソースコネクタを提供するようになりました。詳細については、「Box」を参照してください。	2022 年 4 月 6 日
新機能	Amazon Kendra が Slack のデータソースコネクタを提供するようになりました。詳細については、「Slack」を参照してください。	2022 年 3 月 14 日
新機能	Amazon Kendra が のデータソースコネクタを提供するようになりました Amazon FSx。詳細については、「Amazon FSx」を参照してください。	2022 年 2 月 8 日
AWS マネージドポリシーの更新 - 新しいポリシー	Amazon Kendra は新しい AWS マネージドポリシーを追加しました。詳細については、「AWS Managed policies for Amazon Kendra」を参照してください。	2022 年 1 月 3 日

新機能	Amazon Kendra 検索アプリケーションは、フロントエンドコードを必要とせずに数回クリックするだけでデプロイできます。詳細については、コードなしで検索アプリケーションをデプロイするを参照してください。	2021 年 12 月 1 日
新機能	ドキュメントの取り込みプロセス中に、ドキュメントのメタデータおよび内容を改善できます。詳細については、取り込みプロセス中のドキュメントのメタデータのカスタマイズを参照してください。	2021 年 12 月 1 日
新機能	Amazon Kendra は、検索アプリケーションに関する有用なインサイトを得るために検索分析を提供します。詳細については、検索分析でインサイトを取得するを参照してください。	2021 年 12 月 1 日
リージョンの拡張	Amazon Kendra が AWS GovCloud （米国西部) (-1) us-gov-westで利用可能になりました。	2021 年 10 月 13 日

新機能	Amazon Kendra は、ドキュメントを複数の言語でインデックス化し、検索結果を言語別にフィルタリングできるようになりました。英語以外の言語でドキュメントを追加するおよび各言語での検索を参照してください。	2021 年 10 月 7 日
新機能	Amazon Kendra が Identity Center ディレクトリと統合され、ユーザーコンテキストフィルタリング のためにグループおよびユーザーのアクセスレベルを取得できるようになりました。「User-group configuration for IAM Identity Center」を参照してください。	2021 年 10 月 6 日
チュートリアルの新規追加	Amazon Kendra では、メタデータが豊富な検索ソリューションを構築する方法を説明するチュートリアルが提供されるようになりました。インテリジェント検索ソリューションの構築を参照してください。	2021 年 8 月 13 日
新機能	Amazon Kendra が のデータソースコネクタを提供するようになりました Amazon WorkDocs。詳細については、「Amazon WorkDocs」を参照してください。	2021 年 7 月 20 日

新機能	Amazon Kendra は、ウェブページをクロールしてインデックスを作成するためのウェブクローラーを提供するようになりました。詳細については、「Web crawler」を参照してください。	2021 年 6 月 17 日
リージョンの拡張	Amazon Kendra がカナダ (中部) (ca-central-1) で利用可能になりました。	2021年6月16日
リージョンの拡張	Amazon Kendra が米国東部 (オハイオ) (米国東部 2) で利用可能になりました。	2021 年 6 月 7 日
新機能	Amazon Kendra はクエリの提案をサポートするようになりました。ユーザーは検索に関連する一般的なクエリが提案されます。詳細については、人気のある検索クエリの提案を参照してください。	2021 年 5 月 27 日
AWS マネージドポリシーの更新 - 新しいポリシー	Amazon Kendra は新しい AWS マネージドポリシーを追加しました。詳細については、「AWS Managed policies for Amazon Kendra」を参照してください。	2021 年 5 月 27 日
リージョンの拡張	Amazon Kendra がアジアパシフィック (シンガポール) (ap-southeast-1) で利用可能になりました。	2021 年 5 月 5 日

新機能	Amazon Kendra では、インデックスレベルで設定された調整設定を上書きすることで、クエリ内の検索関連性の調整がサポートされるようになりました。詳細については、検索の関連性のチューニングおよびレスポンスのチューニングを参照してください。	2021 年 4 月 20 日
新機能	Amazon Kendra が OAuth 2.0 認証をサポートし、クエリを使用して ServiceNowインデックス作成用のドキュメントを選択できるようになりました。詳細については、「」を参照してくださいServiceNow。	2021 年 4 月 1 日
新機能	Amazon Kendra で、よくある質問ドキュメントの増分学習がサポートされるようになりました。詳細については、増分学習のフィードバックの送信を参照してください。	2021 年 2 月 17 日
新機能	Amazon Kendra がインデックスシノニムをサポートするようになりました。詳細については、インデックスへのシノニムの追加を参照してください。	2020 年 12 月 10 日

新機能	Amazon Kendra が Google Workspace Drive のデータベースコネクタを提供するようになりました。詳細については、Google Workspace Drive のデータソースを使用するを参照してください。	2020 年 12 月 8 日
新機能	Amazon Kendra は、 へのクエリフィードバックの提供を容易にする JavaScript ライブラリを提供するようになりました Amazon Kendra。詳細については、フィードバックの送信を参照してください。	2020 年 12 月 8 日
新機能	Amazon Kendra がトークンベースのユーザーアクセスコントロールをサポートするようになりました。詳細については、インデックス内のドキュメントへのアクセス制御を参照してください。	2020 年 11 月 5 日
新機能	Amazon Kendra Confluence データソースコネクタが Confluence クラウドで動作するようになりました。詳細については、Confluence データソースの使用を参照してください。	2020 年 11 月 5 日
リージョンの拡張	Amazon Kendra がアジアパシフィック (シドニー) (ap-southeast-2) で利用可能になりました。	2020 年 11 月 2 日

新機能	Amazon Kendra が Confluence サーバーのデータソースコネクタを提供するようになりました。詳細については、Confluence データソースの使用を参照してください。	2020 年 10 月 26 日
新機能	Amazon Kendra で、カスタムコネクタの統計を生成するために使用できるデータソースが提供されるようになりました。詳細については、データソースの使用を参照してください。	2020 年 10 月 21 日
新機能	Amazon Kendra は、よくある質問のカスタム属性をサポートするようになりました。詳細については、質問と回答を追加するを参照してください。	2020 年 9 月 17 日
新機能	Amazon Kendra は、クエリ結果の信頼スコアを返すようになりました。詳細については、「」を参照してくださいQueryResultItem。	2020 年 9 月 15 日
新機能	AWS CloudFormation が をサポートするようになりましたAmazon Kendra。詳細については、「 Amazon Kendra リソースタイプリファレンス - AWS CloudFormation」を参照してください。	2020 年 9 月 10 日

新機能	Amazon Kendra で のサポートが追加されました AWS PrivateLink。詳細については、「Amazon Kendra とインターフェイス VPC エンドポイント (AWS PrivateLink)」を参照してください。	2020 年 7 月 7 日
新しいガイド	これは Amazon Kendra デベロッパーガイドの初回リリースです。	2020 年 5 月 11 日

API リファレンス

「API リファレンスドキュメント」は別のガイドになりました。

AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「AWS 用語集」を参照してください。

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。