aws

入門ガイド

Amazon Redshift

Amazon Redshift: 入門ガイド

Table of Contents

Amazon Redshift は、2026 年 6 月 30 日以降、Python UDF の使用をサポートしなくなります。これは段階的に実施されます。Python のサポート終了と移行オプションの詳細については、2025 年 6 月 30 日に公開されたブログ記事を参照してください。

Amazon Redshift Serverless データウェアハウスの使用を開始

Amazon Redshift Serverless を初めて使用する場合、次のセクションをお読みになって Amazon Redshift Serverless の使用開始の参考にすることをお勧めします。Amazon Redshift Serverless の基本的な流れは、サーバーレスリソースの作成、Amazon Redshift Serverless への接続、サンプルデータのロード、データに対するクエリの実行です。このガイドでは、Amazon Redshift Serverless から、または Amazon S3 バケットからサンプルデータのロードを選択できます。サンプルデータは、Amazon Redshift ドキュメント全体で機能を実証するために使用されます。Amazon Redshift でプロビジョニングされたデータウェアハウスの使用を開始するには、「Amazon Redshift でプロビジョニングされたデータウェアハウスの使用を開始する」を参照してください。

- the section called “Amazon Redshift Serverless によるデータウェアハウスの作成”
- the section called “Amazon S3 からデータをロードする”

AWS アカウントへのサインアップ

AWS を使い始めるには、AWS アカウントが必要です。AWS アカウントの作成の詳細については、「AWS アカウント管理 リファレンスガイド」の「AWS アカウントの開始方法」を参照してください。

Amazon Redshift Serverless によるデータウェアハウスの作成

Amazon Redshift Serverless コンソールに初めてログインすると、サーバーレスリソースの作成と管理に使用できる、入門エクスペリエンスにアクセスするように求められます。このガイドでは、Amazon Redshift Serverless のデフォルト設定を使用してサーバーレスリソースを作成します。

設定をより細かく制御するには、[Customize settings] (設定をカスタマイズ) を選択します。

> **Note**
>
> Redshift Serverless には、3 つの異なるアベイラビリティーゾーンに 3 つのサブネットを持つ Amazon VPC が必要です。Redshift Serverless には、少なくとも 3 個の使用可能な IP アドレスも必要です。Redshift Serverless で Amazon VPC を使用する前に、3 つの異なるアベイラビリティーゾーンに 3 つのサブネットがあり、少なくとも 3 個の使用可能な IP アドレスがあることを確認してください。Amazon VPC のサブネットを作成する方法の詳細につい

ては、「Amazon Virtual Private Cloud ユーザーガイド」の「サブネットの作成」を参照してください。Amazon VPC の IP アドレスの詳細については、「VPC とサブネットの IP アドレス指定」を参照してください。

デフォルト設定で設定するには:

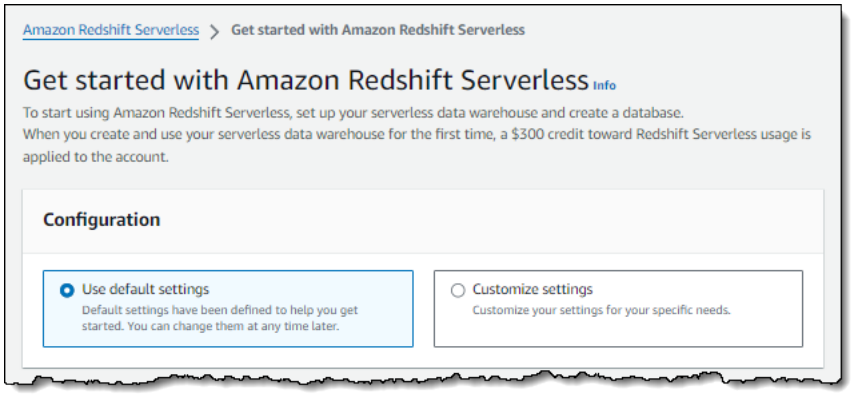
1. AWS マネジメントコンソール にサインインして、https://console.aws.amazon.com/redshiftv2/ で Amazon Redshift コンソールを開きます。

   [Redshift Serverless の無料トライアルをお試しください] を選択します。

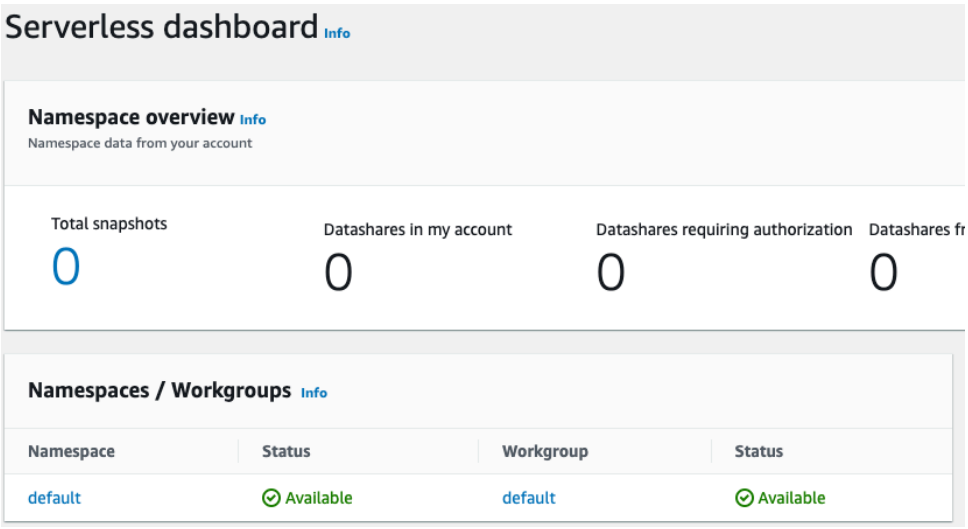
2. [Configuration] (設定) で、[Use default settings] (デフォルト設定を使用) を選択します。Amazon Redshift Serverless は、デフォルトの名前空間と、この名前空間に関連するデフォルトのワークグループを作成します。[設定の保存] を選択します。

   > **Note**
   >
   > 名前空間は、データベースオブジェクトとユーザーのコレクションです。名前空間は、スキーマ、テーブル、ユーザー、データ共有、スナップショットなど、Redshift Serverless で使用するすべてのリソースをグループ化します。
   > ワークグループは、コンピューティングリソースのコレクションです。ワークグループには、Redshift Serverless が計算タスクを実行するために使用するコンピューティングリソースが含まれます。

   次のスクリーンショットは、Amazon Redshift Serverless のデフォルト設定を示しています。



3. セットアップが完了したら、[続行] を選択し、[サーバーレスダッシュボード] に移動します。サーバーレスワークグループと名前空間が使用可能であることがわかります。



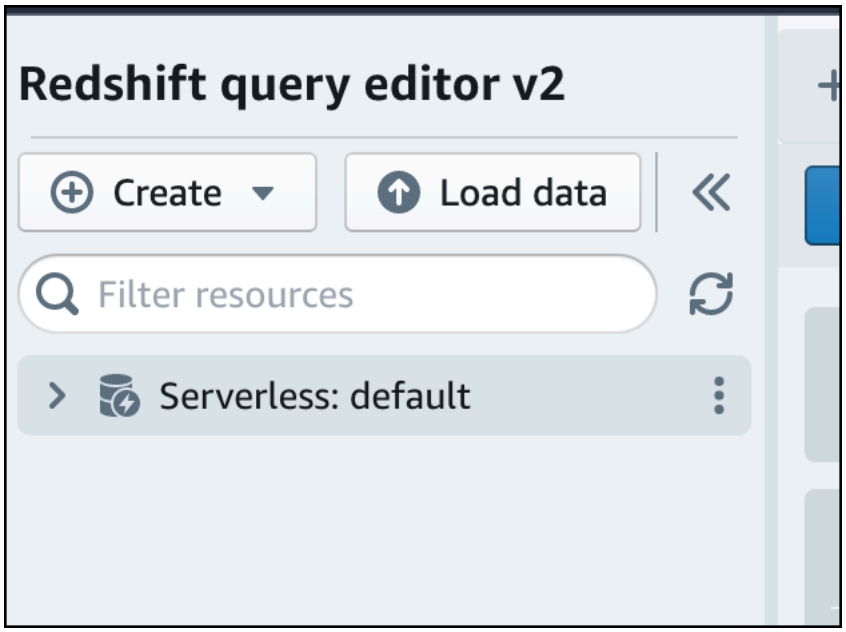
> **Note**
>
> Redshift Serverless がワークグループを正常に作成しない場合は、次の操作を実行できます。
>
> - Amazon VPC 内のサブネット数が少なすぎるなど、Redshift Serverless が報告するすべてのエラーに対処します。
> - Redshift Serverless ダッシュボードで [default-namespace]、[アクション]、[名前空間を削除] の順に選択して名前空間を削除します。名前空間の削除には数分かかります。
> - Redshift Serverless コンソールを再度開くと、ウェルカム画面が表示されます。

サンプルデータをロードする

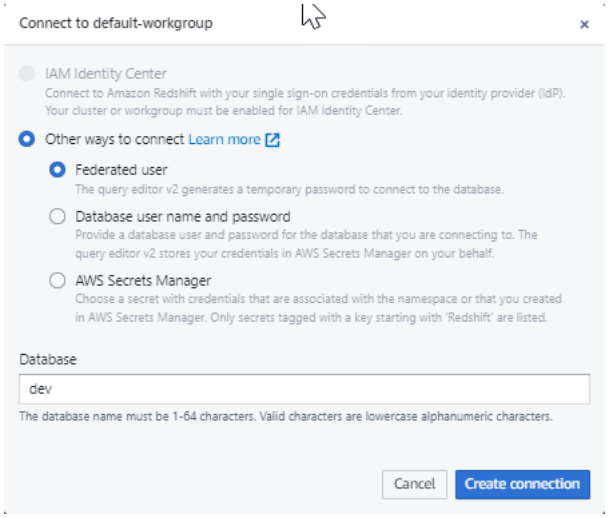
Amazon Redshift Serverless でデータウェアハウスをセットアップしたので、Amazon Redshift クエリエディタ v2 を使用してサンプルデータをロードできます。

1. Amazon Redshift Serverless コンソールからクエリエディタ v2 を起動するには、[データをクエリ] を選択します。Amazon Redshift Serverless コンソールで クエリエディタ v2 を呼び出すと、ブラウザ上に新しいタブが開きクエリーエディタが表示されます。クエリエディタ v2 により、クライアントマシンから Amazon Redshift Serverless 環境に接続されます。

2. このガイドでは、AWS 管理者アカウントとデフォルトの AWS KMS keyを使用します。Amazon Redshift クエリエディタ v2 の設定 (必要なアクセス許可など) については、「Amazon Redshift 管理ガイド」の「AWS アカウント の設定」を参照してください。カスタマーマネージドキーを使用するように Amazon Redshift を設定する方法、または Amazon Redshift で使用する KMS キーを変更する方法については、「名前空間の AWS KMS キーの変更」を参照してください。
3. ワークグループに接続するには、ツリービューパネルでワークグループ名を選択します。

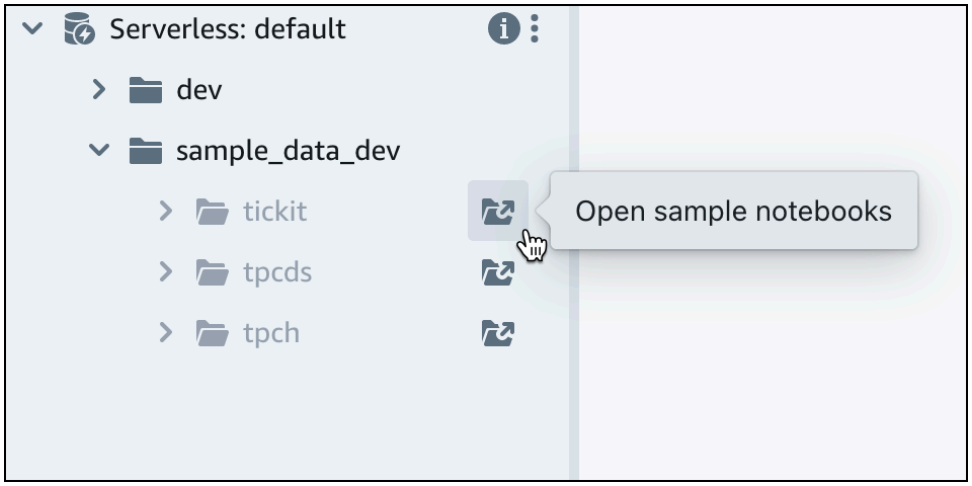


4. クエリエディタ v2 で初めて新しいワークグループに接続するときは、ワークグループへの接続に使用する認証タイプを選択する必要があります。このガイドでは、[フェデレーションユーザー] を選択したままにして、[接続の作成] を選択します。



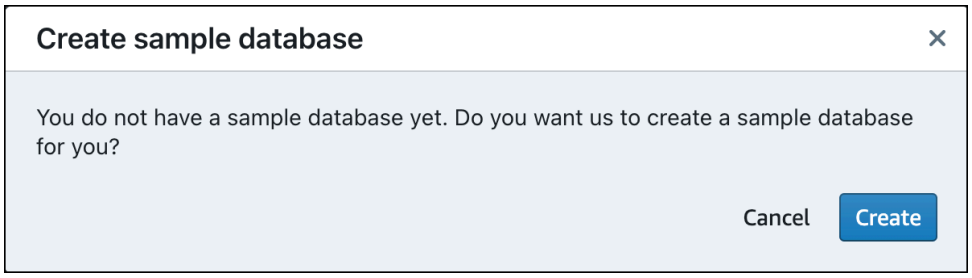
接続したら、Amazon Redshift Serverless から、または Amazon S3 バケットからサンプルデータのロードを選択できます。

5. Amazon Redshift サーバーレスのデフォルトワークグループで、sample_data_dev データベースを展開します。3 つのサンプルデータセットに対応する 3 つのサンプルスキーマがあり、これらを Amazon Redshift Serverless データベース内にロードできます。ロードするサンプルデータセットを選択して、[サンプルノートブックを開く] を選択します。



> **Note**
>
> SQL ノートブックは、SQL セルと Markdown セルのコンテナです。ノートブックを使用して、複数の SQL コマンドを 1 つのドキュメントにまとめて、注釈を付け、共有することができます。

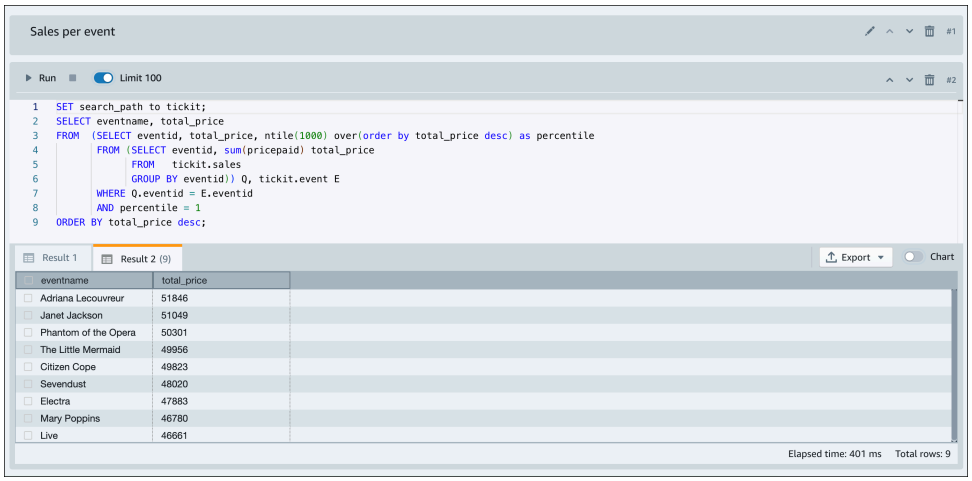
6. 初めてデータをロードするとき、クエリエディタ v2 からサンプルデータベースを作成するように促しされます。[作成] を選択します。



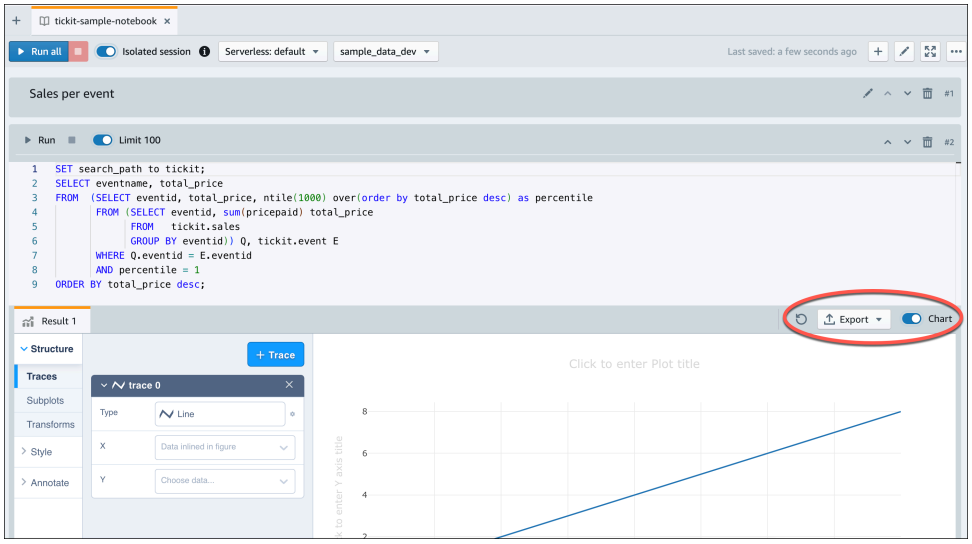
サンプルクエリの実行

Amazon Redshift Serverless をセットアップすると、Amazon Redshift Serverless 内のサンプルデータセットの使用を開始できます。Amazon Redshift Serverless は tickit データセットなどのサンプルデータセットを自動的にロードし、直ちにデータをクエリすることができます。

- Amazon Redshift Serverless がサンプルデータのロードを完了すると、すべてのサンプルクエリがエディタにロードされます。[すべて実行] を選択すると、サンプルノートブックのクエリをすべて実行できます。



結果を JSON または CSV ファイルとしてエクスポートしたり、結果をグラフで表示したりすることもできます。



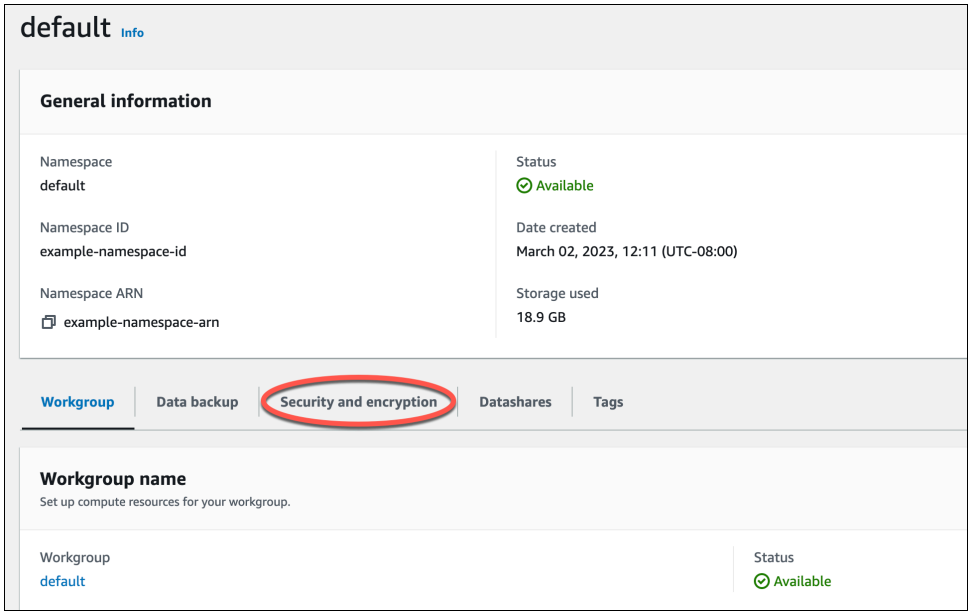
Amazon S3 バケットからデータをロードすることもできます。詳細については、「the section called “Amazon S3 からデータをロードする”」を参照してください。

Amazon S3 からデータをロードする

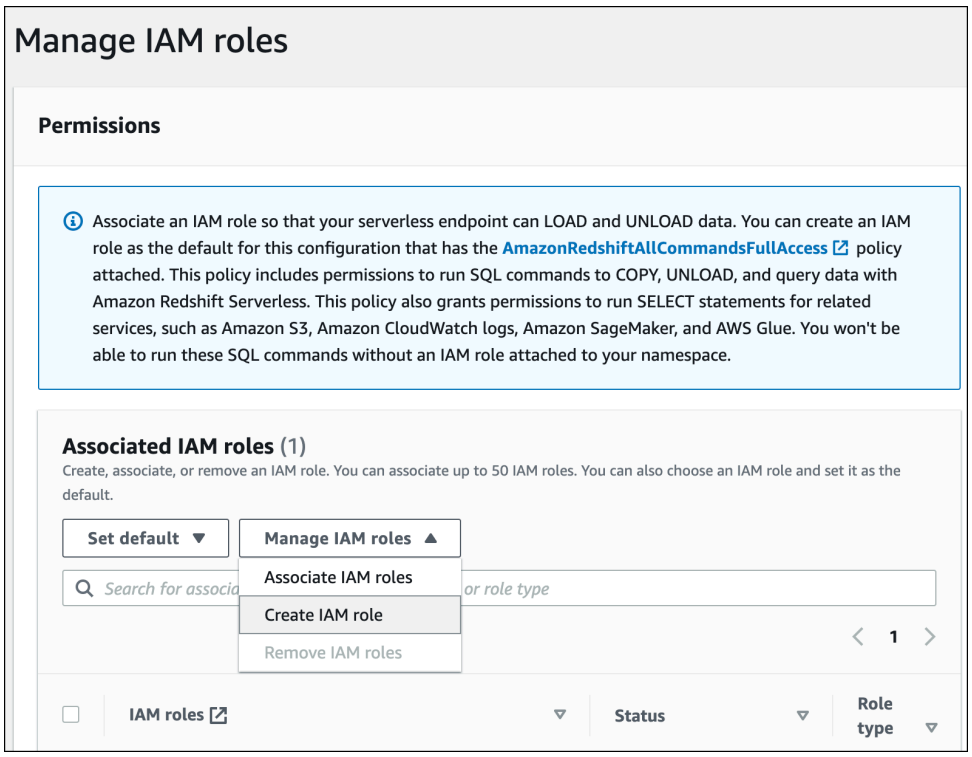
データウェアハウスを作成すると、Amazon S3 からデータをロードできます。

この時点で、dev という名前のデータベースがあります。次に、このデータベースにいくつかのテーブルを作成し、テーブルにデータをアップロードして、クエリを実行してみます。すぐにロードして使えるサンプルデータを Amazon S3 バケットに用意しました。

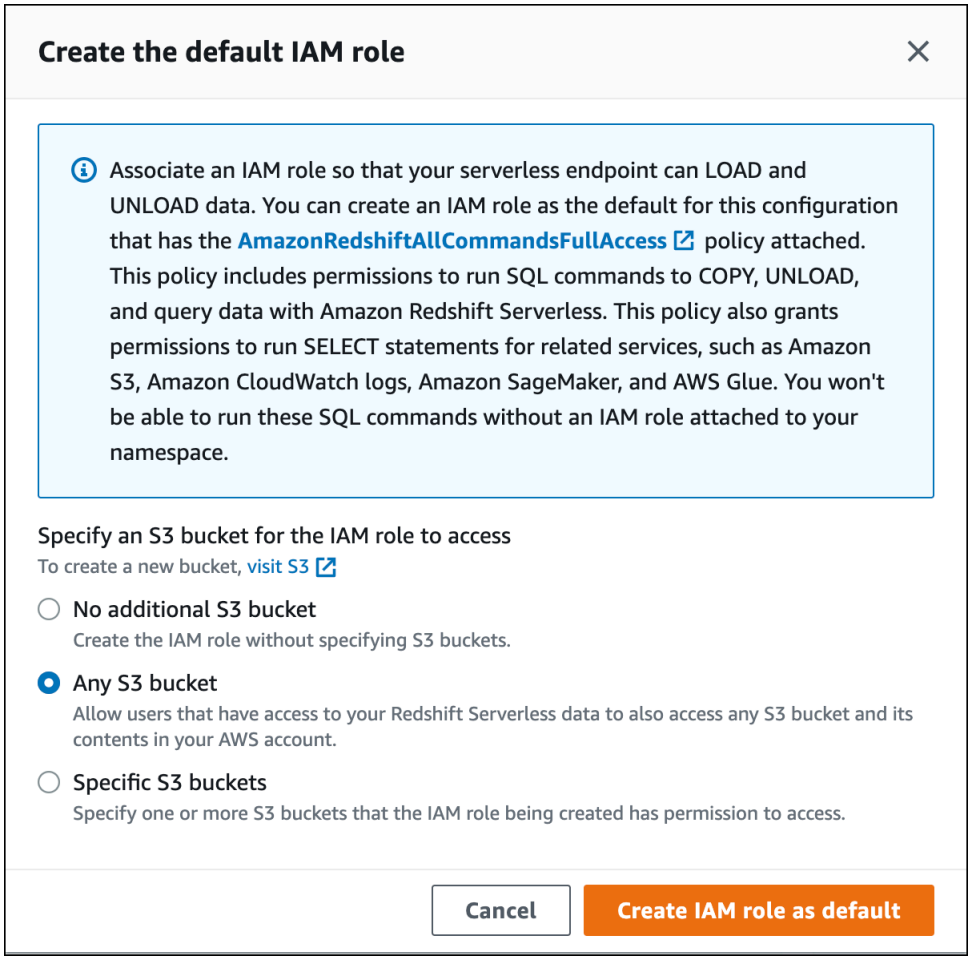
1. Amazon S3 からデータをロードする前に、必要な権限を持つ IAM ロールを作成し、それをサーバーレス名前空間にアタッチする必要があります。これを行うには、Redshift Serverless コンソールに戻り、[名前空間設定] を選択します。ナビゲーションメニューから [名前空間] を選択して、[セキュリティと暗号化] を選択します。次に、[IAM ロールの管理] を選択します。



2. [IAM ロールを管理] メニューを展開して、[IAM ロールの作成] を選択します。



3. このロールに付与する S3 バケットアクセスのレベルを選択して、[デフォルトとして IAM ロールを作成する] を選択します。

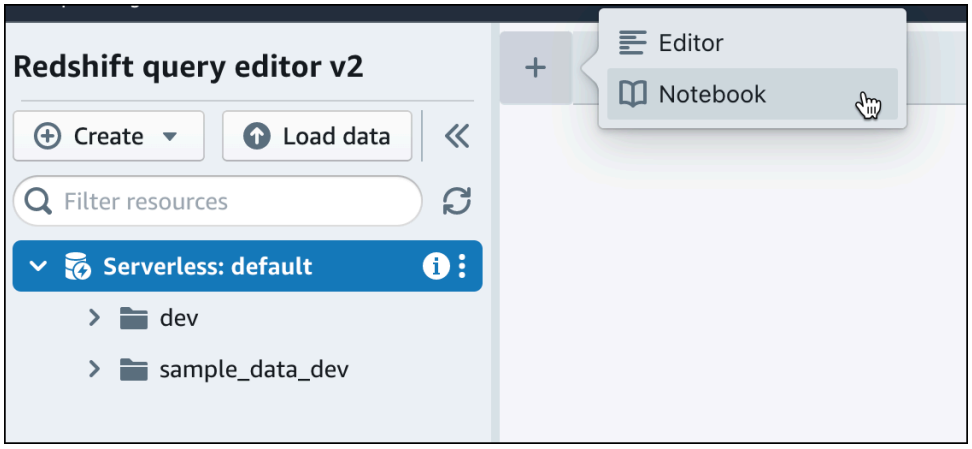


4. [変更を保存] をクリックします。これで Amazon S3 のサンプルデータをロードできるようになりました。

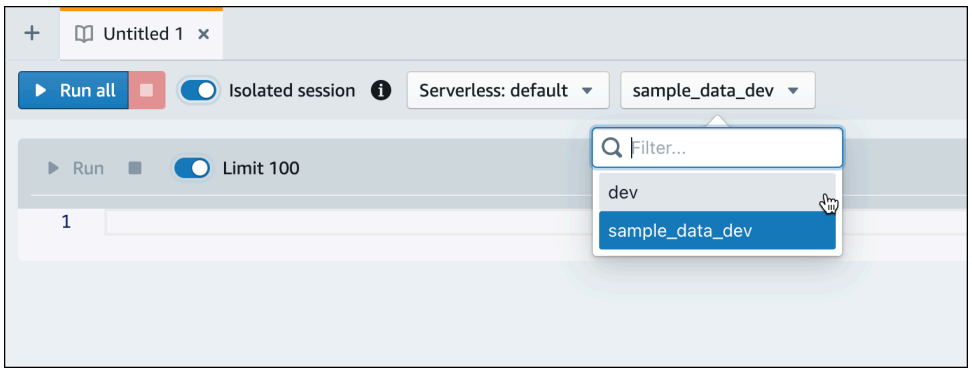
以下の手順ではパブリック Amazon Redshift S3 バケット内のデータを使用しますが、自分の S3 バケットと SQL コマンドを使用して同じ手順を再現することができます。

Amazon S3 からサンプルデータをロードする

1. クエリエディタ v2 で、
   [+
   追加] を選択してから、[ノートブック] を選択して新しい SQL ノートブックを作成します。



2. dev データベースに切り替えます。



3. テーブルを作成します。

クエリエディタ v2 使用している場合は、次のテーブル作成ステートメントを個別にコピーして実行し、dev データベースにテーブルを作成します。構文の詳細については、「Amazon Redshift データベースデベロッパーガイド」の「CREATE TABLE」を参照してください。

```
create table users(
userid integer not null distkey sortkey,
username char(8),
firstname varchar(30),
lastname varchar(30),
city varchar(30),
state char(2),
email varchar(100),
phone char(14),
likesports boolean,
liketheatre boolean,
likeconcerts boolean,
likejazz boolean,
likeclassical boolean,
```

```
likeopera boolean,
likerock boolean,
likevegas boolean,
likebroadway boolean,
likemusicals boolean);

create table event(
eventid integer not null distkey,
venueid smallint not null,
catid smallint not null,
dateid smallint not null sortkey,
eventname varchar(200),
starttime timestamp);

create table sales(
salesid integer not null,
listid integer not null distkey,
sellerid integer not null,
buyerid integer not null,
eventid integer not null,
dateid smallint not null sortkey,
qtysold smallint not null,
pricepaid decimal(8,2),
commission decimal(8,2),
saletime timestamp);
```

4. クエリエディタ v2 で、ノートブックに新しい SQL セルを作成します。

5. Amazon S3 または Amazon DynamoDB から大容量のデータセットを Amazon Redshift にロードする場合は、クエリエディタ v2 の COPY コマンドの使用をお勧めします。COPY 構文の詳細については、「Amazon Redshift データベースデベロッパーガイド」の「COPY」を参照してください。

   COPY コマンドは、パブリック S3 バケットにあるサンプルデータを使用して実行できます。クエリエディタ v2 で次の SQL コマンドを実行します。

```
COPY users
FROM 's3://redshift-downloads/tickit/allusers_pipe.txt'
DELIMITER '|'
TIMEFORMAT 'YYYY-MM-DD HH:MI:SS'
IGNOREHEADER 1
REGION 'us-east-1'
IAM_ROLE default;

COPY event
FROM 's3://redshift-downloads/tickit/allevents_pipe.txt'
DELIMITER '|'
TIMEFORMAT 'YYYY-MM-DD HH:MI:SS'
IGNOREHEADER 1
REGION 'us-east-1'
IAM_ROLE default;

COPY sales
FROM 's3://redshift-downloads/tickit/sales_tab.txt'
DELIMITER '\t'
TIMEFORMAT 'MM/DD/YYYY HH:MI:SS'
IGNOREHEADER 1
REGION 'us-east-1'
IAM_ROLE default;
```

6. データをロードしてから、ノートブックに別の SQL セルを作成して、いくつかのクエリ例を試します。SELECT コマンドの使用に関する詳細については、Amazon Redshift データベースデベロッパーガイドの SELECT を参照してください。サンプルデータの構造とスキーマを理解するため、クエリエディタ v2 を使用してみてください。

```
-- Find top 10 buyers by quantity.
SELECT firstname, lastname, total_quantity
FROM   (SELECT buyerid, sum(qtysold) total_quantity
        FROM  sales
        GROUP BY buyerid
        ORDER BY total_quantity desc limit 10) Q, users
WHERE Q.buyerid = userid
ORDER BY Q.total_quantity desc;

-- Find events in the 99.9 percentile in terms of all time gross sales.
SELECT eventname, total_price
FROM  (SELECT eventid, total_price, ntile(1000) over(order by total_price desc) as
 percentile
```

```
        FROM (SELECT eventid, sum(pricepaid) total_price
              FROM   sales
              GROUP BY eventid)) Q, event E
       WHERE Q.eventid = E.eventid
       AND percentile = 1
ORDER BY total_price desc;
```

データをロードし、いくつかのサンプルクエリを実行したので、Amazon Redshift Serverless の他の領域を調べることができます。Amazon Redshift Serverless の使用方法の詳細については、次のリストを参照してください。

- Amazon S3 バケットからデータをロードできます。詳細については、「Amazon S3 からデータをロードする」を参照してください。
- クエリエディタ v2 を使用すると、5 MB 未満のローカルの文字区切りされたファイルからデータをロードできます。詳細については、「ローカルファイルからのデータのロード」を参照してください。
- Amazon Redshift Serverless には、JDBC ドライバーと ODBC ドライバーを備えたサードパーティー製 SQL ツールで接続できます。詳細については、「Amazon Redshift Serverless への接続」を参照してください。
- Amazon Redshift Data API を使用して、Amazon Redshift Serverless に接続することもできます。詳細については、「Amazon Redshift Data API の使用」を参照してください。
- Amazon Redshift Serverless 内のデータと Redshift ML を使用して、CREATE MODEL コマンドで機械学習モデルを作成できます。Redshift ML モデルの構築方法については、「チュートリアル: カスタマーチャーンモデルの構築」を参照してください。
- Amazon Redshift Serverless にデータをロードせずに Amazon S3 データレイクからデータをクエリすることができます。詳細については、「データレイクのクエリ」を参照してください。

Amazon Redshift でプロビジョニングされたデータウェアハウスの使用を開始する

Amazon Redshift を初めて使用する場合は、次のセクションをお読みになり、プロビジョニングされたクラスターの使用開始方法を確認することをお勧めします Amazon Redshift の基本的な流れは、プロビジョニングされたリソースの作成、Amazon Redshift への接続、サンプルデータのロード、そのデータに対するクエリの実行です。このガイドでは、Amazon Redshift から、または Amazon S3 バケットからサンプルデータのロードを選択できます。サンプルデータは、Amazon Redshift ドキュメント全体で機能を実証するために使用されます。

このチュートリアルでは、システムリソースを管理する AWS データウェアハウスオブジェクトである、Amazon Redshift でプロビジョニングされたクラスターを使用する方法について説明します。Amazon Redshift は、使用状況に応じて自動的にスケールするデータウェアハウスオブジェクトであるサーバーレスワークグループでも使用できます。Redshift Serverless の使用を開始するには、「Amazon Redshift Serverless データウェアハウスの使用を開始」を参照してください。

Amazon Redshift プロビジョンドコンソールを作成してサインインすると、クラスター、ノード。データベースなどの Amazon Redshift オブジェクトを作成および管理できます。また、SQL クライアントでは、クエリを実行または表示したり、その他の SQL データ定義言語 (DDL) やデータ操作言語 (DML) のオペレーションを実行したりできます。

> **Important**
>
> この演習用にプロビジョニングしたクラスターは、ライブ環境で実行します。実行中は、お客様の AWS アカウントに課金されます。料金については、Amazon Redshift の料金表ページを参照してください。
> 不要な課金を回避するため、作業が完了したクラスターは削除してください。この章の最後のセクションでは、その方法について説明します。

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/redshiftv2/ で Amazon Redshift コンソールを開きます。

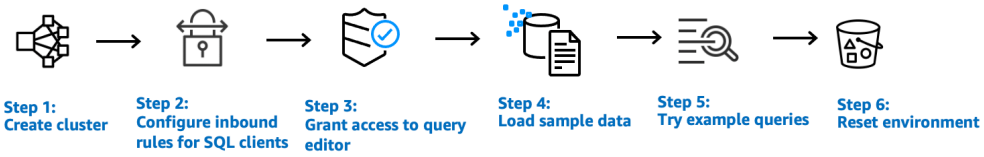
Amazon Redshift コンソールの使用を開始するには、まず [プロビジョニングされたクラスターダッシュボード] に移動することをお勧めします。

設定に応じて、Amazon Redshift プロビジョンドコンソールのナビゲーションペインに以下の項目が表示されます。

- Redshift Serverless — Amazon Redshift でプロビジョニングされたクラスターのセットアップ、チューニング、管理を行うことなく、データにアクセスして分析します。
- プロビジョニングされたクラスターダッシュボード — AWS リージョン のクラスターを一覧表示し、[クラスターメトリクス] および [クエリの概要] でメトリクスデータ (CPU 使用率など) へのインサイトやクエリ情報を確認します。これらを使用すると、パフォーマンスデータの異常が指定した時間範囲で発生しているかどうかを判断できます。
- クラスター — この AWS リージョンのクラスターを一覧表示し、クエリを開始するか、クラスターに関連するアクションを実行するためのクラスターを選択します。また、このページで新しいクラスターを作成することもできます。
- クエリエディタ - Amazon Redshift クラスターがホストするデータベースにクエリを実行します。クエリエディタ v2 を代わりに使用することをお勧めします。
- クエリエディタ v2 — Amazon Redshift クエリエディタ v2 は、独立したウェブベースの SQL クライアントアプリケーションであり、Amazon Redshift データウェアハウスへのクエリを作成および実行します。結果をグラフで視覚化し、チーム内の他のユーザーとクエリを共有することで共同作業を行うことができます。
- クエリとロード — 最近のクエリのリストや各クエリの SQL テキストなど、リファレンスまたはトラブルシューティングに使用するための情報を取得します。
- データ共有 – プロデューサアカウントの管理者は、コンシューマアカウントがデータ共有にアクセスすることを許可するか、アクセスを許可しないかのいずれかを選択できます。承認されたデータ共有を使用する場合、コンシューマーアカウントの管理者は AWS アカウント アカウント全体またはアカウント内の特定のクラスター名前空間にデータ共有を関連付けることができます。また管理者は、データ共有を拒否することもできます。
- ゼロ ETL 統合 — サポートされているソースに書き込んだトランザクションデータを Amazon Redshift で利用可能にする統合を管理します。
- IAM アイデンティティセンター接続 — Amazon Redshift と IAM アイデンティティセンター間の接続を設定します。
- 設定 – Java Database Connectivity (JDBC) および Open Database Connectivity (ODBC) 接続を介して、SQL クライアントツールから Amazon Redshift クラスターに接続します。Amazon RedShift が管理する仮想プライベートクラウド (VPC) エンドポイントを設定することもできます。これにより、Amazon VPC サービスをベースにしており、クラスターが置かれている VPC と、クライアントツールを実行している別の VPC との間にプライベート接続が提供されます。
- AWS パートナー統合 — サポートされている AWS パートナーとの統合を作成します。
- アドバイザー – Amazon Redshift クラスターに加えることができる変更に関する、具体的な推奨事項を入手して、最適化の優先順位を決定します。

- AWS Marketplace – 他のツール、もしくは Amazon Redshift と連携する AWS サービスに関する情報を取得します。
- アラーム – パフォーマンスデータを表示するためにクラスターメトリクスのアラームを作成し、指定した期間中のメトリクスを追跡します。
- イベント – イベントを追跡し、そのイベントが発生した日付、その説明、およびイベントソースなどの情報に関するレポートを取得します。
- 最新情報 – Amazon Redshift の新機能と製品アップデートを表示します。

本チュートリアルでは、次のステップを実行します。



トピック

- AWS アカウント へのサインアップ
- ファイアウォールルールの決定
- ステップ 1: サンプルの Amazon Redshift クラスターを作成する
- ステップ 2: SQL クライアントのインバウンドルールを設定する
- ステップ 3: SQL クライアントにアクセスを許可し、クエリを実行する
- ステップ 4: Amazon S3 から Amazon Redshift にデータをロードする
- ステップ 5: クエリエディタを使用してクエリ例を試す
- ステップ 6: 環境をリセットする

AWS アカウント へのサインアップ

AWS を使い始めるには、AWS アカウントが必要です。AWS アカウントの作成の詳細については、「AWS アカウント管理 リファレンスガイド」の「AWS アカウントの開始方法」を参照してください。

ファイアウォールルールの決定

> **Note**
>
> このチュートリアルでは、クラスターがデフォルトのポート 5439 を使用すること、および Amazon Redshift クエリエディタ v2 を使用して SQL コマンドを実行できることを前提としています。環境で必要になる可能性があるネットワーク設定や SQL クライアント設定については詳しく説明しません。

一部の環境では、Amazon Redshift クラスターの起動時にポートを指定します。このポートをクラスターのエンドポイント URL とともに使用して、クラスターにアクセスします。また、このポートを経由するクラスターへのアクセスを許可するために、インバウンド進入ルールをセキュリティグループに作成します。

クライアントコンピュータがファイアウォールの内側にある場合、使用可能な開いているポートを把握していることを確認してください。この開いているポートを使用して、SQL クライアントツールからクラスターに接続してクエリを実行できます。このポートが分からない場合は、ネットワークファイアウォールのルールを把握している担当者の協力を得て、ファイアウォールで開いているポートを確認してください。

Amazon Redshift はデフォルトでポート 5439 を使用しますが、お使いのファイアウォールでこのポートが開いていない場合は接続できません。作成後の Amazon Redshift クラスターでは、ポート番号を変更することはできません。したがって、必ず起動プロセス中にお使いの環境で機能する開いているポートを指定してください。

ステップ 1: サンプルの Amazon Redshift クラスターを作成する

このチュートリアルでは、1 つのデータベースを含む Amazon Redshift クラスターを作成する手順について説明します。次に、Amazon S3 からデータベース内のテーブルにデータセットをロードします。このサンプルクラスターは、Amazon Redshift サービスを評価するのに使用できます。

Amazon Redshift クラスターの設定を開始する前に、「ファイアウォールルールの決定」などの必要な前提条件を必ず完了してください。

別の AWS リソースからデータにアクセスするオペレーションの場合、クラスターには、そのリソースとリソース上のデータに対しユーザーに代わってアクセスするためのアクセス許可が必要です。

この場合の例としては、SQL COPY コマンドを使用して Amazon Simple Storage Service (Amazon S3) からデータをロードすることが挙げられます。AWS Identity and Access Management (IAM) を使用してアクセス権限を提供できます。これを行うには、クラスターに対して作成してアタッチした IAM ロールを使用できます。認証情報とアクセス許可の詳細については、「Amazon Redshift データベース開発者ガイド」の「認証情報とアクセス許可」を参照してください。

Amazon Redshift クラスターを作成するには

1. AWS マネジメントコンソール にサインインして、https://console.aws.amazon.com/redshiftv2/ で Amazon Redshift コンソールを開きます。

   > **Important**
   >
   > IAM ユーザー認証情報を使う場合は、クラスターオペレーションを実行するために必要なアクセス許可を持っていることを確認してください。詳細については、「Amazon Redshift 管理ガイド」の「Amazon Redshift のセキュリティ」を参照してください。

2. AWS コンソールで、クラスターを作成する先の AWS リージョンを選択します。
3. ナビゲーションメニューで [Clusters] (クラスター)、[Create cluster] (クラスターを作成) の順に選択します。[クラスターの作成] ページが表示されます。
4. [Cluster configuration (クラスター構成)] セクションで、[Cluster identifier (クラスター識別子)]、[Node type (ノードタイプ)]、および [Nodes (ノード)] の値を指定します。

   - クラスター識別子: このチュートリアルでは **`examplecluster`** を入力します。この識別子は一意である必要があります。識別子は、有効な文字として a ~ z (小文字のみ) および - (ハイフン) を使って、1 ~ 63 文字にする必要があります。
   - クラスターのサイズを設定するには、次のいずれかの方法を選択します。

     > **Note**
     >
     > 次のステップでは、RG ノードタイプをサポートしている AWS リージョンを前提としています。RG または RA3 ノードタイプをサポートする AWS リージョンのリストについては、「Amazon Redshift 管理ガイド」の「AWS リージョンでの RG ノードタイプの可用性」および「AWS リージョンでの RA3 ノードタイプの可用性」を参照してください。各ノードタイプのノード仕様とサイズの詳細については、「ノードタイプの詳細を参照してください」。

- クラスターのサイズがわからない場合は、[選択のヘルプ] を選んでください。これにより、データウェアハウスに保存するデータのサイズやクエリ特性について質問するサイジング計算ツールが開きます。

  クラスターの必須サイズ (ノードタイプとノード数) がわかっている場合は、[自分で選択します] を選んでください。次に、[Node type] (ノードタイプ) と [Nodes] (ノード) の数を選択して、クラスターのサイズを設定します。

  このチュートリアルでは、[ノードタイプ] に [rg.4xlarge] を、[ノード数] に [2] をそれぞれ選択します。

  [AZ 設定] を選択できる場合は、[シングル AZ] を選択します。

- Amazon Redshift が提供するサンプルデータセットを使用するには、[Sample data] (サンプルデータ) で、[Load sample data] (サンプルデータをロード) をクリックします。Amazon Redshift は、サンプルデータセットの Tickit をデフォルトの`dev` データベースと `public` スキーマにロードします。

5. [データベース設定] セクションで、[管理者ユーザー名] の値を指定します。[管理者パスワード] で、次のオプションから選択します。

   - [パスワードの生成] – Amazon Redshift によって生成されたパスワードを使用します。
   - [管理者パスワードを手動で追加する] – 独自のパスワードを使用します。
   - [AWS Secrets Manager での管理者認証情報の管理] – Amazon Redshift は管理者パスワードの生成と管理に AWS Secrets Manager を使用します。AWS Secrets Manager を使用してパスワードのシークレットの生成と管理を行うには料金がかかります。AWS Secrets Manager の料金の詳細については、「AWS Secrets Manager の料金」を参照してください。

   このチュートリアルでは以下の値を使用します。

   - [管理者ユーザー名]: **`awsuser`** を入力します。
   - [管理者ユーザーパスワード]: パスワードに **`Changeit1`** を入力します。

6. 以下で説明するように、このチュートリアルでは、 IAM ロールを作成し、そのロールをクラスターのデフォルトとして設定しています。クラスターにデフォルトとして設定できる IAM ロールセットは 1 つのみです。

   a. [Cluster permissions] (クラスターのアクセス許可) にある [Manage IAM roles] (IAM ロールの管理) で、[Create IAM role] (IAM ロールを作成) をクリックします。

b. 次のいずれかの方法で、 IAM ロールがアクセスする Amazon S3 バケットを指定します。

- 作成された IAM ロールに対し、`redshift` という名前の Amazon S3 バケットへのアクセスのみを許可するには、[No additional Amazon S3 bucket] (追加の Amazon S3 バケットはありません) を選択します。
- 作成された IAM ロールが、すべての Amazon S3 バケットにアクセスできるようにするには、[Any Amazon S3 bucket] (すべての Amazon S3 バケット) を選択します。
- 作成された IAM ロールがアクセスする 1 つ以上の Amazon S3 バケットを指定するには、[Specific Amazon S3 buckets] (特定の Amazon S3 バケット) を選択します 次に、テーブルから Amazon S3 バケットを 1 つ以上選択します。

c. [Create IAM role as default] (デフォルトとして IAM ロールを作成する) をクリックします。Amazon Redshift は、クラスター用に IAM ロールを自動的に作成し、デフォルトとして設定します。

コンソールから IAM ロールを作成したため、IAM ロールには `AmazonRedshiftAllCommandsFullAccess` ポリシーがアタッチされています。これにより Amazon Redshift は、IAM アカウント内の Amazon リソースからのデータのコピーやロード、そのデータに対するクエリおよび分析の実行が可能になります。

クラスターのデフォルトの IAM ロールを管理する詳しい方法については、「Amazon Redshift 管理ガイド」の「Amazon Redshift 用にデフォルトの IAM ロールを作成する」を参照してください。

7. (オプション) [追加設定] セクションで、[デフォルトを使用] をオフにして、[ネットワークとセキュリティ]、[データベース設定]、[メンテナンス]、[モニタリング]]、および [バックアップ] の各設定を変更します。

場合によっては、[Load sample data] (サンプルデータをロード) を使用してクラスターを作成し、拡張 Amazon VPC ルーティングを有効すること考えられます。その場合、仮想プライベートクラウド (VPC) のクラスターは Amazon S3 エンドポイントにアクセスし、データをロードする必要があります。

クラスターをパブリックにアクセスできるようにするには、以下の 2 つのうちいずれかを実行します。クラスターがインターネットにアクセスできるように、ネットワークアドレス変換 (NAT) アドレスを VPC 内で設定します。または、VPC で Amazon S3 VPC エンドポイントを設定します。拡張された Amazon VPC ルーティングの詳細については、「Amazon Redshift 管理ガイド」の「拡張された Amazon VPC ルーティングをオンにする」を参照してください。

8. [クラスターを作成] を選択します。[クラスター] ページでクラスターが作成され、ステータスが [Available] になるまで待ちます。

ステップ 2: SQL クライアントのインバウンドルールを設定する

> **Note**
>
> このステップをスキップし、Amazon Redshift クエリエディタ v2 を使用してクラスターにアクセスすることをお勧めします。

このチュートリアルの後半では、Amazon VPC サービスに基づく Virtual Private Cloud (VPC) 内からクラスターにアクセスします。ただし、ファイアウォールの外部から SQL クライアントを使用してクラスターにアクセスする場合は、必ずインバウンドアクセスを許可してください。

ファイアウォールを確認し、クラスターへのインバウンドアクセスを許可するには

1. ファイアウォールの外側からクラスターがアクセスされる必要がある場合、ファイアウォールのルールを確認します。この例としては、クライアントが Amazon Elastic Compute Cloud (Amazon EC2) インスタンスか、外部のコンピュータである場合などが挙げられます。

   ファイアウォールルールの詳細については、「Amazon EC2 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。
2. Amazon EC2 外部クライアントからアクセスするには、インバウンドトラフィックを許可するクラスターにアタッチされたセキュリティグループに進入ルールを追加します。Amazon EC2 コンソールに Amazon EC2 セキュリティグループのルールを追加します。例えば、CIDR/IP が 192.0.2.0/24 の場合、その IP アドレス範囲のクライアントはクラスターに接続できます。環境に適した CIDR/IP を見つけます。

ステップ 3: SQL クライアントにアクセスを許可し、クエリを実行する

Amazon Redshift クラスターがホストするデータベースを SQL クライアントからクエリするには、複数のオプションがあります。具体的には次のとおりです。

- Amazon Redshift クエリエディタ v2 を使用してクラスターに接続し、クエリを実行します。

クエリエディタ v2 を使用すると、SQL クライアントアプリケーションをダウンロードしてセットアップする必要はありません。Amazon Redshift クエリエディタ v2 は、Amazon Redshift コンソールから起動します。

- RSQL を使用してクラスターに接続します。詳細については、「Amazon Redshift 管理ガイド」の「Amazon Redshift RSQL を使用した接続」を参照してください。https://docs.aws.amazon.com/redshift/latest/mgmt/rsql-query-tool.html
- SQL Workbench/J などの SQL クライアントツールを介してクラスターに接続します。詳細については、「Amazon Redshift 管理ガイド」の「SQL Workbench/J を使用してクラスターに接続する」を参照してください。

このチュートリアルでは、Amazon Redshift クラスターがホストするデータベースにクエリを実行する簡単な方法として、Amazon Redshift クエリエディタ v2 を使用します。クラスターを作成したら、すぐにクエリを実行できます。Amazon Redshift クエリエディタ v2 を使用する際の詳しい考慮事項については、「Amazon Redshift 管理ガイド」の「クエリエディタ v2 を使用する際の考慮事項」を参照してください。

クエリエディタ v2 に対するアクセス許可を付与する

管理者が AWS アカウントのために最初にクエリエディタ v2 を設定するときは、クエリエディタ v2 のリソースを暗号化するために使用する AWS KMS key を選択します。Amazon Redshift クエリエディタ v2 のリソースには、保存されたクエリ、ノートブック、チャートなどがあります。デフォルトでは、AWS 所有キーは、リソースを暗号化するために使用されます。または、管理者はカスタマーマネージドキーを使用することもできます。その場合は、設定ページでキーの Amazon リソースネーム (ARN) を選択します。アカウントの設定後は、AWS KMS を使用した暗号化の設定は変更できません。詳細については、「Amazon Redshift 管理ガイド」の「AWS アカウントの設定」を参照してください。

クエリエディタ v2 にアクセスするには、アクセス許可が必要です。管理者は、Amazon Redshift クエリエディタ v2 の AWS マネージドポリシーのいずれかを、IAM ロールまたはユーザーにアタッチして、アクセス許可を付与できます。これらの AWS 管理ポリシーは、リソースのタグ付けでクエリを共有する方法を制御するさまざまなオプションを使用して記述されます。IAM コンソール (https://console.aws.amazon.com/iam/) を使用して IAM ポリシーをアタッチできます。詳細については、「Amazon Redshift 管理ガイド」の「クエリエディタ v2 へのアクセス」を参照してください。

また、提供された管理ポリシーで許可もしくは拒否されたアクセス権限に基づいて、独自のポリシーを作成することもできます。IAM コンソールのポリシーエディタを使用して独自のポリシーを作成する場合は、ビジュアルエディタでポリシーを作成する対象のサービスとして、[SQL Workbench]

を選択します。クエリエディタ v2 では、ビジュアルエディタ および IAM Policy Simulator の中で、サービス名として AWS SQL Workbench を使用します。

詳細については、「Amazon Redshift 管理ガイド」の「クエリエディタ v2 の使用」を参照してください。

ステップ 4: Amazon S3 から Amazon Redshift にデータをロードする

クラスターを作成したら、Amazon S3 からデータベーステーブルにデータをロードできます。Amazon S3 からデータをロードする方法は複数あります。

- SQL クライアントを使用して SQL CREATE TABLE コマンドを実行することで、データベースにテーブルを作成し、SQL COPY コマンドを使用して Amazon S3 からデータをロードできます。Amazon Redshift クエリエディタ v2 は、SQL クライアントです。
- Amazon Redshift クエリエディタ v2 のロードウィザードを使用できます。

このチュートリアルでは、Amazon Redshift Query Editor V2 で SQL コマンドを実行してテーブルを作成し、データをコピーする方法を示します。Amazon Redshift コンソールのナビゲーションペインからクエリエディタ v2 を起動します。クエリエディタ v2 で、管理者ユーザー `awsuser` として `examplecluster` クラスターと `dev` という名前のデータベースへの接続を作成します。このチュートリアルでは、接続の作成時に [データベースユーザー名を使用した一時的な認証情報] を選択します。Amazon Redshift クエリエディタ v2 の使用の詳細については、「Amazon Redshift 管理ガイド」の「Amazon Redshift データベースに接続する」を参照してください。

SQL コマンドを使用して Amazon S3 からデータをロードする

クエリエディタ v2 のクエリエディタペインで、`examplecluster` クラスターと `dev` データベースに接続していることを確認します。次に、データベースにテーブルを作成し、テーブルにデータをロードします。このチュートリアルでは、Amazon S3 バケットからデータをロード可能にし、多くの AWS リージョンからアクセスできるようにします。

次の手順では、テーブルを作成し、公開の Amazon S3 バケットからデータをロードします。

Amazon Redshift クエリエディタ v2 を使用して次の create table ステートメントをコピーして実行し、`dev` データベースの `public` スキーマにテーブルを作成します。構文の詳細については、「Amazon Redshift データベースデベロッパーガイド」の「CREATE TABLE」を参照してください。

クエリエディタ v2 などの SQL クライアントを使用してデータを作成およびロードするには

1. 次の SQL コマンドを実行して sales テーブルを作成します。

```
  drop table if exists sales;
  create table sales(
salesid integer not null,
listid integer not null distkey,
sellerid integer not null,
buyerid integer not null,
eventid integer not null,
dateid smallint not null sortkey,
qtysold smallint not null,
pricepaid decimal(8,2),
commission decimal(8,2),
saletime timestamp);
```

2. 次の SQL コマンドを実行して date テーブルを作成します。

```
drop table if exists date;
create table date(
 dateid smallint not null distkey sortkey,
 caldate date not null,
 day character(3) not null,
 week smallint not null,
 month character(5) not null,
 qtr character(5) not null,
 year smallint not null,
 holiday boolean default('N'));
```

3. COPY コマンドを使用して、Amazon S3 から sales テーブルをロードします。

> **Note**
>
> Amazon S3 から Amazon Redshift に大容量のデータセットをロードする場合は、COPY コマンドを使用することをお勧めします。COPY 構文の詳細については、「Amazon Redshift データベースデベロッパーガイド」の「COPY」を参照してください。

サンプルデータをロードするために、ユーザーに代わってクラスターが Amazon S3 にアクセスするための認証を提供します。認証を行うには、クラスターの作成時に [IAM ロールをデフォルトとして作成する] を選択し、クラスターに対して作成して default として設定した IAM ロールを参照します。

次の SQL コマンドを使用して sales テーブルをロードします。オプションで、Amazon S3 バケットから sales テーブルのソースデータをダウンロードして表示できます。。

```
COPY sales
    FROM 's3://redshift-downloads/tickit/sales_tab.txt'
    DELIMITER '\t'
    TIMEFORMAT 'MM/DD/YYYY HH:MI:SS'
    REGION 'us-east-1'
    IAM_ROLE default;
```

4. 次の SQL コマンドを使用して date テーブルをロードします。オプションで、Amazon S3 バケットから date テーブルのソースデータをダウンロードして表示できます。。

```
COPY date
    FROM 's3://redshift-downloads/tickit/date2008_pipe.txt'
    DELIMITER '|'
    REGION 'us-east-1'
    IAM_ROLE default;
```

クエリエディタ v2 を使用して Amazon S3 からデータをロードする

このセクションでは、独自のデータを Amazon Redshift クラスターにロードする方法について説明します。クエリエディタ v2 では、[データのロード] ウィザードを使用してデータを簡単にロードできます。クエリエディタ v2 の [データのロード] ウィザードで生成および使用する COPY コマンドは、Amazon S3 からデータをロードするための COPY コマンド構文で使用できるパラメータの多くをサポートしています。Amazon S3 からコピーおよびロードをするために使用される COPY コマンドと、そのオプションの詳細については、Amazon Redshift データベース開発者ガイドの「Amazon S3 からの COPY」を参照してください。

Amazon S3 から Amazon Redshift に独自のデータをロードする際、Amazon Redshift には、指定された Amazon S3 バケットからデータをロードするための権限を持つ IAM ロールが必要です。

独自のデータを Amazon S3 から Amazon Redshift にロードするには、クエリエディタ v2 の [データのロード] ウィザードを使用できます。[データのロード] ウィザードの使用方法について詳しくは、「Amazon Redshift 管理ガイド」の「Amazon S3 からデータをロードする」を参照してください。

クラスターで TICKIT データを作成する

TICKIT は、Amazon Redshift でのデータのクエリ方法を学ぶ目的で、オプションで Amazon Redshift クラスターにロードできるサンプルデータベースです。完全な TICKIT テーブルセットを作成してクラスターにデータをロードするには、以下の方法を使用できます。

- Amazon Redshift コンソールでクラスターを作成する場合は、TICKIT のサンプルデータを同時にロードできます。Amazon Redshift コンソールで、[クラスター]、[クラスターを作成] の順に選択します。[サンプルデータ] セクションで、[サンプルデータをロード] を選択します。Amazon Redshift のサンプルデータセットは、クラスターの作成時に Amazon Redshift クラスターの `dev` データベースに自動的にロードされます。
- 既存のクラスターに接続するには、以下の手順を行います。
  - Amazon Redshift コンソールのナビゲーションバーで、[クラスター] を選択します。
  - [クラスター] ペインでクラスターを選択します。
  - [クエリデータ]、[クエリエディタ v2 でクエリ] の順に選択します。
  - リソースリストで examplecluster を展開します。クラスターに初めて接続する場合は、[examplecluster に接続] が表示されます。[データベースユーザー名とパスワード] を選択します。データベースは **`dev`** のままにします。ユーザー名に **`awsuser`** をパスワードに **`Changeit1`** を指定します。
  - [接続を作成] を選択します。
- Amazon Redshift クエリエディタ v2 を使用すると、TICKIT データを sample_data_dev という名前のサンプルデータベースにロードできます。リソースリストで [sample_data_dev] データベースを選択します。tickit ノードの横にある、[サンプルノートブックを開く] アイコンを選択します。サンプルデータベースを作成することを確認します。
- Amazon Redshift クエリエディタ v2 は、サンプルデータベースと共に tickit-sample-notebook という名前のサンプルノートブックを作成します。[すべてを実行] を選択してこのノートブックを実行し、サンプルデータベース内のデータをクエリできます。

TICKIT データの詳細については、「Amazon Redshift データベース開発者ガイド」の「サンプルデータベース」を参照してください。

ステップ 5: クエリエディタを使用してクエリ例を試す

データベースをクエリするように Amazon Redshift クエリエディタ v2 を設定して使用するには、「Amazon Redshift 管理ガイド」の「Amazon Redshift クエリエディタ v2の使用」を参照してください。

次に示すように、今、いくつかのクエリ例をお試しください。クエリエディタ v2 で新しいクエリを作成するには、クエリペインの右上にある [+] アイコンを選択し、[SQL] を選択します。新しいクエリページが表示されたら、次の SQL クエリをコピーして貼り付けます。

> **Note**
>
> まずノートブックで最初のクエリを実行し、次の SQL コマンドを使用して search_path サーバー設定値を tickit スキーマに設定します。
>
> ```
> set search_path to tickit;
> ```

SELECT コマンドの使用の詳細については、「Amazon Redshift データベース開発者ガイド」の「SELECT」を参照してください。

```
-- Get definition for the sales table.
SELECT *
FROM pg_table_def
WHERE tablename = 'sales';
```

```
-- Find total sales on a given calendar date.
SELECT sum(qtysold)
FROM   sales, date
WHERE  sales.dateid = date.dateid
AND    caldate = '2008-01-05';
```

```
-- Find top 10 buyers by quantity.
SELECT firstname, lastname, total_quantity
FROM   (SELECT buyerid, sum(qtysold) total_quantity
        FROM  sales
        GROUP BY buyerid
        ORDER BY total_quantity desc limit 10) Q, users
```

```
WHERE Q.buyerid = userid
ORDER BY Q.total_quantity desc;
```

```
-- Find events in the 99.9 percentile in terms of all time gross sales.
SELECT eventname, total_price
FROM  (SELECT eventid, total_price, ntile(1000) over(order by total_price desc) as
 percentile
       FROM (SELECT eventid, sum(pricepaid) total_price
             FROM   sales
             GROUP BY eventid)) Q, event E
       WHERE Q.eventid = E.eventid
       AND percentile = 1
ORDER BY total_price desc;
```

ステップ 6: 環境をリセットする

以上のステップでは、Amazon Redshift クラスターの作成、テーブルへのデータのロード、Amazon Redshift クエリエディター v2 などの SQL クライアントを使用したデータのクエリを正常に完了しました。

このチュートリアルを完了したら、サンプルクラスターを削除して、環境を前の状態にリセットすることをお勧めします。クラスターを削除するまで、そのクラスターについて Amazon Redshift サービスの使用料が継続して発生します。

ただし、他の Amazon Redshift ガイドのタスクや「コマンドを実行して、データウェアハウス内のデータベースを定義して使用する」で説明しているタスクを試す場合は、サンプルクラスターの実行を継続することもできます。

クラスターを削除するには

1. AWS マネジメントコンソール にサインインして、https://console.aws.amazon.com/redshiftv2/ で Amazon Redshift コンソールを開きます。
2. ナビゲーションメニューから [Clusters] (クラスター) を選択してクラスターのリストを表示します。
3. `examplecluster` クラスターを選択します。[アクション] で、[削除] を選択します。[examplecluster を削除しますか?] ページが表示されます。
4. 削除するクラスターを確認し、[最終スナップショットを作成] 設定のボックスをオフにして、**`delete`** を入力して削除を確認します。[クラスターを削除] を選択します。

クラスターリストページで、クラスターのステータスはクラスターが削除される際に更新されます。

本チュートリアルを完了した後は、Amazon Redshift について学習するためのその他のリソースで、Amazon Redshift に関する詳細と次に行うステップを説明しています。

コマンドを実行して、データウェアハウス内のデータベースを定義して使用する

Redshift Serverless データウェアハウスと Amazon Redshift でプロビジョニングされたデータウェアハウスの両方にデータベースが含まれています。データウェアハウスを起動したら、SQL コマンドを使用してほとんどのデータベースアクションを管理できます。いくつかの例外を除いて、SQL の機能と構文は、すべての Amazon Redshift データベースに共通です。Amazon Redshift で使用できる SQL コマンドの詳細については、「Amazon Redshift データベース開発者ガイド」の「SQL コマンド」を参照してください。

データウェアハウスを作成すると、ほとんどのシナリオにおいて、Amazon Redshift はデフォルトの dev データベースも作成します。dev データベースへの接続後に、別のデータベースを作成できます。

以下のセクションでは、Amazon Redshift データベースを使用する際の一般的なデータベースタスクについて説明します。タスクはデータベースの作成から始まり、最後のタスクまで続けた場合は、データベースを削除することで作成したすべてのリソースを削除できます。

このセクションの例では、以下を前提とします。

- Amazon Redshift データウェアハウスを作成済みである。
- Amazon Redshift クエリエディタ v2 などの SQL クライアントツールからデータウェアハウスへの接続を確立済みである。クエリエディタ v2 の詳細については、「Amazon Redshift 管理ガイド」の「Amazon Redshift クエリエディタ v2 を使用したデータベースのクエリの実行」を参照してください。

トピック

Amazon Redshift データウェアハウスに接続する

Amazon Redshift クラスターに接続するには、[クラスター] ページで [Amazon Redshift クラスターに接続する] を展開し、次のいずれかの操作を実行します。

- [データをクエリ] を選択し、クエリエディタ v2 を使用して Amazon Redshift クラスターがホストするデータベースにクエリを実行します。クラスターの作成後は、クエリエディタ v2 を使用して、すぐにクエリを実行できます。

  詳細については、「Amazon Redshift 管理ガイド」の「クエリエディタ v2 を使用したデータベースのクエリの実行」を参照してください。
- [クライアントツールを使用] で、クラスターを選択します。次に、JDBC や ODBC ドライバーの URL をコピーすることで JDBC や ODBC ドライバーを使用し、クライアントツールから Amazon Redshiftに接続します。この URL は、クライアントコンピュータまたはインスタンスから使用します。JDBC または ODBC のデータアクセス API オペレーションを使用する、もしくは JDBC または ODBC をサポートする SQL クライアントツールを使用するように、アプリケーションを記述します。

  クラスター接続文字列を検索する方法については、「クラスター接続文字列を検索する」を参照してください。
- SQL クライアントにドライバーが必要な場合は、JDBC または ODBC ドライバーを選択してオペレーティングシステム固有のドライバーをダウンロードし、クライアントツールから Amazon Redshift に接続できます。

  SQL クライアントに適したドライバーをインストールする方法の詳細については、「JDBC ドライバーのバージョン 2.x 接続の構成」を参照してください。

  ODBC 接続を設定する方法の詳細については、「ODBC 接続の設定」を参照してください。

Redshift Serverless データウェアハウスに接続するには、Amazon Redshift コンソールの [サーバーレスダッシュボード] ページから、次のいずれかの操作を実行します。

- Amazon Redshift クエリエディタ v2 を使用して、Redshift Serverless データウェアハウスがホストするデータベースにクエリを実行します。データウェアハウスの作成後は、クエリエディタ v2 を使用して、すぐにクエリを実行できます。

  詳細については、「Amazon Redshift クエリエディタ v2 を使用したデータベースのクエリの実行」を参照してください。

- JDBC または ODBC ドライバーの URL をコピーし、そのドライバーを使用することで、クライアントツールから Amazon Redshiftに接続します。

  データウェアハウス内のデータを使用するには、クライアントコンピュータやインスタンスから接続するための JDBC または ODBC ドライバーが必要です。JDBC または ODBC のデータアクセス API オペレーションを使用する、もしくは JDBC または ODBC をサポートする SQL クライアントツールを使用するように、アプリケーションを記述します。

  接続文字列を見つける詳しい方法については、「Amazon Redshift 管理ガイド」の「Redshift Serverless への接続」を参照してください。

データベースを作成する

データウェアハウスが稼動していることを確認したら、データベースを作成できます。このデータベースを使用して実際にテーブルを作成し、データをロードし、クエリを実行します。データウェアハウスは、複数のデータベースをホストできます。例えば、SALESDB というセールスデータ用のデータベースと ORDERSDB という注文データ用のデータベースを、同じデータウェアハウスでホストできます。

SALESDB という名前のデータベースを作成するには、SQL クライアントツールで次のコマンドを実行します。

```
CREATE DATABASE salesdb;
```

> **Note**
>
> コマンドを実行したら、データウェアハウスで SQL クライアントツールのオブジェクトリストを更新し、新しい salesdb を確認してください。

この演習では、デフォルトの設定をそのまま使用します。この他のコマンドオプションについては、Amazon Redshift データベース開発者ガイドの「CREATE DATABASE」を参照してください。データベースとその内容を削除するには、「Amazon Redshift データベース開発者ガイド」の「DROP DATABASE」を参照してください。

SALESDB データベースを作成した後は、SQL クライアントからこのデータベースに接続できます。現在の接続に使用したものと同じ接続パラメータを使用しますが、データベース名は SALESDB に変更してください。

ユーザーの作成

デフォルトでは、データウェアハウスの起動時に作成した管理者ユーザーのみが、データウェアハウス内のデフォルトデータベースにアクセスできます。他のユーザーにもアクセス権を与えるには、そのためのアカウントを作成する必要があります。データベースのユーザーアカウントは、データベース別に固有のものではなく、データウェアハウス内のすべてのデータベースに共通です。

新しいユーザーを作成するには、CREATE USER コマンドを使用します。新しいユーザーを作成する際は、ユーザー名とパスワードを指定します。ユーザーには、パスワードを指定することをお勧めします。パスワードは 8 ~ 64 文字で構成し、大文字、小文字、数字をそれぞれ少なくとも 1 つずつ含める必要があります。

例えば、**GUEST** という名前と **ABCd4321** というパスワードでユーザーを作成するには、以下のコマンドを実行します。

```
CREATE USER GUEST PASSWORD 'ABCd4321';
```

GUEST ユーザーとして SALESDB データベースに接続するには、ユーザーの作成時に同じパスワード (例えば ABCd4321) を使用します。

その他のコマンドオプションについては、Amazon Redshift データベース開発者ガイドの「CREATE USER」を参照してください。

スキーマの作成

新しいデータベースを作成した後は、そのデータベースに新しいスキーマを作成できます。スキーマとは、テーブル、ビュー、およびユーザー定義関数 (UDF) など、名前が付けられたデータベースオブジェクトが含まれる名前空間です。データベースには 1 つまたは複数のスキーマを含めることができ、各スキーマは 1 つのデータベースにのみ属します。2 つのスキーマが、同じ名前を共有する異なるオブジェクトを持つことができます。

同じデータベース内に複数のスキーマを作成して、好みの方法でデータを整理したり、データを機能的にグループ化したりできます。例えば、すべてのステージングデータを格納するスキーマや、すべてのレポートテーブルを保存する別のスキーマを作成できます。また、同じデータベースにある異なるビジネスグループに関連するデータを保存するために、別々のスキーマを作成することもできます。スキーマごとに、テーブル、ビュー、ユーザー定義関数 (UDF) など、異なるデータベースオブジェクトを格納できます。さらに、AUTHORIZATION 句を使用してスキーマを作成することもで

きます。この句では、特定のユーザーに所有権を与えることや、指定したスキーマが使用できる最大ディスク領域を指定するクオータを設定することなどが可能です。

Amazon Redshift は、新しいデータベースごとに、public という名前のスキーマを自動的に作成します。データベースオブジェクトの作成中にスキーマ名を指定しない場合、そのオブジェクトは public スキーマに入ります。

スキーマ内のオブジェクトにアクセスするには、schema_name.table_name 表記を使用してオブジェクト修飾します。スキーマの修飾名は、ドットで区切られたスキーマ名とテーブル名で構成されます。例えば、price テーブルを持つ sales スキーマや、price テーブルを持つ inventory スキーマのようになります。price テーブルを参照する際には、その名前を sales.price または inventory.price のように修飾する必要があります。

次の例では、GUEST ユーザーのために、**SALES** という名前のスキーマを作成します。

```
CREATE SCHEMA SALES AUTHORIZATION GUEST;
```

その他のコマンドオプションについては、Amazon Redshift データベース開発者ガイドの「CREATE SCHEMA」を参照してください。

データベース内のスキーマのリストを表示するには、次のコマンドを実行します。

```
select * from pg_namespace;
```

出力は以下の例のようになります。

```
  nspname             | nspowner |         nspacl
----------------------+----------+--------------------------
  sales               |  100     |
  pg_toast            |   1      |
  pg_internal         |   1      |
  catalog_history     |   1      |
  pg_temp_1           |   1      |
  pg_catalog          |   1      | {rdsdb=UC/rdsdb,=U/rdsdb}
  public              |   1      | {rdsdb=UC/rdsdb,=U/rdsdb}
  information_schema  |   1      | {rdsdb=UC/rdsdb,=U/rdsdb}
```

カタログテーブルに対するクエリの方法については、Amazon Redshift データベース開発者ガイドの「カタログテーブルへのクエリの実行」を参照してください。

スキーマへのアクセス許可をユーザーに付与するには、GRANT ステートメントを使用します。

次の例では、SELECT ステートメントを使用して SALES スキーマ内のすべてのテーブルやビューからデータを選択する権限を GUEST ユーザーに付与します。

```
GRANT SELECT ON ALL TABLES IN SCHEMA SALES TO GUEST;
```

次の例では、すべての使用可能な権限を一度に GUEST ユーザーに付与します。

```
GRANT ALL ON SCHEMA SALES TO GUEST;
```

テーブルを作成する

新しいデータベースの作成後には、データを格納するためのテーブルを作成します。テーブルの作成時に列情報を指定します。

例えば、**DEMO** という名前のテーブルを作成するには、次のコマンドを実行します。

```
CREATE TABLE Demo (
  PersonID int,
  City varchar (255)
);
```

デフォルトでは、テーブルなどの新しいデータベースオブジェクトは、データウェアハウスの作成時に作成される public という名前のデフォルトスキーマ内に作成されます。また別のスキーマを使用しても、データベースオブジェクトを作成できます。スキーマの詳細については、Amazon Redshift データベース開発者ガイドの「データベースセキュリティの管理」を参照してください。

また、schema_name.object_name 表記を使用して SALES スキーマ内にテーブルを作成することもできます。

```
CREATE TABLE SALES.DEMO (
  PersonID int,
  City varchar (255)
);
```

スキーマとそのテーブルを表示および検査するには、Amazon Redshift クエリエディタ v2 を使用できます。または、システムビューを使用して、スキーマ内のテーブルのリストを表示することもできます。詳細については、「システムテーブルとビューをクエリする」を参照してください。

encoding 列、distkey 列、および sortkey 列は、Amazon Redshift が並列処理のために使用します。これらの要素を含むテーブルの設計については、「Amazon Redshift テーブル設計のベストプラクティス」を参照してください。

テーブルにデータ行を挿入する

テーブルの作成後は、そのテーブル内にデータ行を挿入します。

> **Note**
>
> INSERT コマンドは、テーブルに行を挿入します。標準的なバルク負荷の場合は、COPY コマンドを使用します。詳細については、「COPY コマンドを使ってデータをロードする」を参照してください。

例えば、DEMO テーブルに値を挿入するには、以下のコマンドを実行します。

```
INSERT INTO DEMO VALUES (781, 'San Jose'), (990, 'Palo Alto');
```

特定のスキーマ内にあるテーブルにデータを挿入するには、次のコマンドを実行します。

```
INSERT INTO SALES.DEMO VALUES (781, 'San Jose'), (990, 'Palo Alto');
```

テーブルからデータを選択する

テーブルを作成してデータを挿入した後、テーブル内にあるデータを表示するには、SELECT ステートメントを使用します。SELECT * ステートメントは、テーブル内のすべてのデータについて、すべての列名と行の値を返します。SELECT を使用すると、新しく追加したデータがテーブルに正しく入力されたかどうかを、便利に確認できます。

DEMO テーブルに入力したデータを表示するには、次のコマンドを実行します。

```
SELECT * from DEMO;
```

結果は次のようになります。

```
 personid |   city
----------+-----------
      781 | San Jose
```

```
      990 | Palo Alto
(2 rows)
```

クエリテーブルでの SELECT ステートメントの使用方法については、「SELECT」を参照してください。

データをロードする

このガイドの例の多くでは、TICKIT サンプルデータセットを使用しています。個別のサンプルデータファイルを含むファイル tickitdb.zip をダウンロードします。サンプルデータは、各自の Amazon S3 バケットにアップロードできます。

データベースのサンプルデータをロードするには、まずテーブルを作成します。その後、COPY コマンドを使って、Amazon S3 バケットに保存されている、サンプルデータを格納したテーブルをロードします。テーブルを作成し、サンプルデータをロードする手順については、「ステップ 4: Amazon S3 から Amazon Redshift にデータをロードする」を参照してください。

システムテーブルとビューをクエリする

データウェアハウスには、作成したテーブルに加え、いくつかのシステムテーブルとビューが含まれています。これらのテーブルとビューには、インストールに関する情報と、システムで実行されている各種クエリや処理に関する情報が格納されます。これらのシステムテーブルとビューに対してクエリを実行して、データベースに関する情報を収集することができます。詳細については、「Amazon Redshift データベース管理者ガイド」の「システムテーブルとビューのリファレンス」を参照してください。各テーブルやビューについての説明では、テーブルをすべてのユーザーが表示できるか、スーパーユーザーのみが表示できるかを示しています。スーパーユーザーのみが表示可能なテーブルに対してクエリを実行するには、スーパーユーザーとしてログインします。

テーブル名のリストを表示する

スキーマ内のすべてのテーブルのリストを表示するには、PG_TABLE_DEF システムカタログテーブルをクエリします。最初に search_path に対する設定を確認します。

```
SHOW search_path;
```

その結果は以下のようになります。

```
  search_path
```

```
---------------
 $user, public
```

次の例では、SALES スキーマを検索パスに追加し、SALES スキーマ内のすべてのテーブルを表示します。

```
set search_path to '$user', 'public', 'sales';

SHOW search_path;

      search_path
------------------------
 "$user", public, sales


select * from pg_table_def where schemaname = 'sales';

 schemaname | tablename |  column  |          type          | encoding | distkey |
 sortkey | notnull
------------+-----------+----------+------------------------+----------+---------
+---------+---------
 sales      | demo      | personid | integer                | az64     | f       |
  0 | f
 sales      | demo      | city     | character varying(255) | lzo      | f       |
  0 | f
```

次の例では、現在のデータベース上のすべてのスキーマに含まれる、DEMO という名前のすべてのテーブルのリストを表示します。

```
set search_path to '$user', 'public', 'sales';
select * from pg_table_def where tablename = 'demo';

 schemaname | tablename |  column  |          type          | encoding | distkey |
 sortkey | notnull
------------+-----------+----------+------------------------+----------+---------
+---------+---------
 public     | demo      | personid | integer                | az64     | f       |
  0 | f
 public     | demo      | city     | character varying(255) | lzo      | f       |
  0 | f
 sales      | demo      | personid | integer                | az64     | f       |
  0 | f
```

```
 sales      | demo      | city     | character varying(255) | lzo      | f       |
  0 | f
```

詳細については、「PG_TABLE_DEF」を参照してください。

Amazon Redshift クエリエディタ v2 を使用して接続先のデータベースを最初に選択することで、指定したスキーマ内のすべてのテーブルを表示することもできます。

ユーザーを表示する

ユーザー ID (USESYSID) およびユーザー権限とともに、すべてのユーザーのリストを表示するには、PG_USER カタログをクエリします。

```
SELECT * FROM pg_user;

  usename   | usesysid | usecreatedb | usesuper | usecatupd |  passwd  | valuntil |
 useconfig
------------+----------+-------------+----------+-----------+----------+----------
+-----------
 rdsdb      |        1 | true        | true     | true      | ******** | infinity |
 awsuser    |      100 | true        | true     | false     | ******** |          |
 guest      |      104 | true        | false    | false     | ******** |          |
```

定期的な管理およびメンテナンスタスクを実行するために、Amazon Redshift の内部でユーザー名 rdsdb が使用されます。SELECT ステートメントに `where usesysid > 1` を追加すると、クエリをフィルタリングしてユーザー定義のユーザー名のみを表示することができます。

```
SELECT * FROM pg_user WHERE usesysid > 1;

  usename   | usesysid | usecreatedb | usesuper | usecatupd |  passwd  | valuntil |
 useconfig
------------+----------+-------------+----------+-----------+----------+----------
+-----------
 awsuser    |      100 | true        | true     | false     | ******** |          |
 guest      |      104 | true        | false    | false     | ******** |          |
```

最近のクエリを表示する

先の例では、adminuser のユーザー ID (user_id) は 100 です。adminuser が実行した最近の 4 つのクエリを一覧表示するには、SYS_QUERY_HISTORY ビューをクエリできます。

このビューを使用して、最近実行したクエリのクエリ ID (query_id) やプロセス ID (session_id) を確認できます。また、このビューを使用してクエリが完了するまでにかかった時間を確認できます。SYS_QUERY_HISTORY には、特定のクエリを見つけやすくするために、クエリ文字列 (query_text) の最初の 4,000 文字が含まれています。SELECT ステートメントで LIMIT 句を使用すると、結果を制限できます。

```
SELECT query_id, session_id, elapsed_time, query_text
FROM sys_query_history
WHERE user_id = 100
ORDER BY start_time desc
LIMIT 4;
```

結果は以下のようになります。

```
 query_id |  session_id  |  elapsed_time |   query_text
----------+--------------+---------------
+-----------------------------------------------------------------
 892      |    21046     |       55868   | SELECT query, pid, elapsed, substring
 from ...
 620      |    17635     |     1296265   | SELECT query, pid, elapsed, substring
 from ...
 610      |    17607     |       82555   | SELECT * from DEMO;
 596      |    16762     |      226372   | INSERT INTO DEMO VALUES (100);
```

実行中のクエリのセッション ID を確認する

クエリに関するシステムテーブル情報を取得するには、クエリに関連するセッション ID (プロセス ID) の指定が必要になる場合があります。または、実行中のクエリのセッション ID の確認が必要になる場合もあります。例えば、プロビジョニングされたクラスターで実行時間が長すぎるクエリをキャンセルする場合は、セッション ID が必要になります。STV_RECENTS システムテーブルにクエリを実行すると、実行中のクエリのプロセス ID および対応するクエリ文字列のリストを取得できます。クエリから複数のセッションが返された場合は、クエリテキストを参照して、どのセッション ID が必要かを判断できます。

実行中のクエリのセッション ID を確認するには、次の SELECT ステートメントを実行します。

```
SELECT session_id, user_id, start_time, query_text
FROM sys_query_history
WHERE status='running';
```

クエリをキャンセルする

クエリの実行時間が長すぎる場合や、リソースの消費数が多すぎる場合は、クエリをキャンセルします。この例としては、チケット販売者の名前と販売済みのチケット数を記載したリストを、作成しようとしているような場合が挙げられます。以下のクエリでは、SALES テーブルと USERS テーブルからデータを選択し、WHERE 句内で SELLERID と USERID を一致させることで 2 つのテーブルを結合しています。

```
SELECT sellerid, firstname, lastname, sum(qtysold)
FROM sales, users
WHERE sales.sellerid = users.userid
GROUP BY sellerid, firstname, lastname
ORDER BY 4 desc;
```

結果は以下のようになります。

```
 sellerid | firstname | lastname | sum
----------+-----------+----------+------
  48950   |   Nayda   |   Hood   | 184
  19123   |   Scott   | Simmons  | 164
  20029   |    Drew   | Mcguire  | 164
  36791   |  Emerson  | Delacruz | 160
  13567   |   Imani   |   Adams  | 156
  9697    |  Dorian   |    Ray   | 156
  41579   | Harrison  | Durham   | 156
  15591   |  Phyllis  |  Clay    | 152
  3008    |  Lucas    | Stanley  | 148
  44956   |  Rachel   |Villarreal| 148
```

> **Note**
>
> これは複雑なクエリです。このチュートリアルでは、このクエリの構造を理解する必要はありません。

前のクエリは数秒間で実行され、2,102 行を返しました。

ここで、WHERE 句を挿入し忘れたとします。

```
SELECT sellerid, firstname, lastname, sum(qtysold)
```

```
FROM sales, users
GROUP BY sellerid, firstname, lastname
ORDER BY 4 desc;
```

その場合、結果セットには、SALES テーブルのすべての行と、USERS テーブルのすべての行を掛け合わせた数 (49989×3766) が含まれることになります。これはデカルト結合と呼ばれ、推奨されません。この例の場合は結果が 1 億 8800 万行を超えてしまい、実行に時間がかかりすぎます。

実行中のクエリをキャンセルするには、クエリのセッション ID を指定して CANCEL コマンドを実行します。Amazon Redshift クエリエディタ v2 を使用すると、クエリの実行中に [キャンセル] ボタンを選択してクエリをキャンセルできます。

セッション ID を見つけるには、前のステップで示したように、新しいセッションを開始して STV_RECENTS テーブルをクエリします。次の例に、結果をより読みやすくする方法を示します。これを行うには、TRIM 関数を使用してクエリ文字列の末尾部分を削除し、最初の 20 文字だけを表示します。

実行中のクエリのセッション ID を確認するには、次の SELECT ステートメントを実行します。

```
SELECT user_id, session_id, start_time, query_text
FROM sys_query_history
WHERE status='running';
```

結果は以下のようになります。

```
 user_id |   session_id  |   start_time               |   query_text
---------+---------------+----------------------------
+-------------------------------------------------------------
 100     |    1073791534 | 2024-03-19 22:26:21.205739 | SELECT user_id, session_id,
 start_time, query_text FROM  ...
```

セッション ID が 1073791534 であるクエリをキャンセルするには、次のコマンドを実行します。

```
CANCEL 1073791534;
```

> **Note**
>
> CANCEL コマンドはトランザクションを停止しません。トランザクションを停止またはロールバックするには、ABORT や ROLLBACK コマンドを使用します。トランザクションに関

連付けられたクエリをキャンセルするには、まずクエリをキャンセルした後にトランザクションを停止します。

キャンセルしたクエリがトランザクションに関連付けられている場合は、ABORT または ROLLBACK コマンドを使用してトランザクションをキャンセルし、データに加えられた変更をすべて破棄します。

```
ABORT;
```

スーパーユーザー以外でサインしている場合は、自分のクエリのみをキャンセルできます。スーパーユーザーはすべてのクエリをキャンセルできます。

お使いのクエリツールがクエリの同時実行をサポートしていない場合は、別のセッションを開始してクエリをキャンセルします。

クエリのキャンセルの詳細については、「Amazon Redshift データベース開発者ガイド」の「CANCEL」を参照してください。

Superuser キューを使ってクエリをキャンセルする

現在のセッションで同時に実行されているキューの数が非常に多い場合、他のクエリが完了するまで、CANCEL コマンドを実行できないことがあります。そのような場合、別のワークロード管理クエリキューを使用して CANCEL コマンドを実行することができます。

ワークロード管理を使用すると、クエリを別のクエリキューで実行できるため、他のキューが完了するのを待つ必要がなくなります。ワークロード管理は、Superuser キューと呼ばれる個別のキューを作成します。このキューはトラブルシューティングに使用できます。Superuser キューを使用するには、スーパーユーザーとしてログインし、SET コマンドを使用してクエリグループを「superuser」に設定します。コマンドを実行したあと、RESET コマンドを使用してクエリグループをリセットします。

superuser キューを使用してクエリをキャンセルするには、以下のコマンドを実行します。

```
SET query_group TO 'superuser';
CANCEL 1073791534;
RESET query_group;
```

Amazon Redshift データベースでデータをクエリする

以下に、Amazon S3 データ、リモートデータベースマネージャー、リモート Amazon Redshift データベース、Amazon Redshift を使用した機械学習 (ML) モデルのトレーニングなど、リモートソースでデータのクエリを開始する方法について説明します。

トピック

ご使用のデータレイクのクエリの実行

Amazon Redshift を使用したクエリにより、データを Amazon Redshift テーブルにロードすることなく、Amazon S3 のデータを取得できます。Amazon Redshift は、Amazon Redshift クラスターと Amazon S3 データレイクの両方に保存されている非常に大きなデータセットの高速オンライン分析処理(OLAP)用に設計された SQL 機能を提供します。Iceberg、Parquet、ORC、RCFile、TextFile、SequenceFile、RegexSerde、OpenCSV、AVRO など、さまざまな形式でデータをクエリできます。Amazon S3 でファイルの構造を定義するには、外部スキーマとテーブルを作成します。その後、AWS Glue または独自のApache Hive メタストアなど、外部のデータカタログを使用します。いずれの外部データカタログへの変更も、ただちにすべての Amazon Redshift クラスターに反映されます。

AWS Glue データカタログにデータを登録し AWS Lake Formation で有効化した後、データレイクのクエリを開始できます。

外部テーブルを1 つ以上の列でパーティション分割し、パーティション消去でクエリのパフォーマンスを最適化することができます。Amazon Redshift テーブルを使用し、外部テーブルのクエリと結合ができます。複数の Amazon Redshift クラスターから外部テーブルにアクセスすることが可能で、同じ AWS リージョン内のあらゆるクラスターから Amazon S3 のデータにクエリを実行できます。Amazon S3 データファイルを更新すると、即時に、あらゆる Amazon Redshift クラスターから、そのデータをクエリすることが可能になります。

RG と Redshift Serverless の統合データレイククエリエンジンの使用

Amazon Redshift RG クラスターおよび Amazon Redshift Serverless には、クラスター独自のコンピューティングリソースで実行される統合データレイククエリエンジンが含まれており、データレイクとデータウェアハウスの両方のユースケースに対して統一されたエクスペリエンスを提供します。

統合されたデータレイククエリエンジンにより、Redshift Spectrum を使用する必要がなくなり、関連する Redshift Spectrum の料金も発生しなくなります。統合データレイククエリエンジンは、別の AWS リージョンにある Amazon S3 バケット内のデータのクエリもサポートしています。クロスリージョンクエリでは、追加のデータ転送料金が発生します。デフォルトでは有効になっているため、統合データレイククエリエンジンを有効にするために追加の設定は必要ありません。

> **Note**
>
> 場合によっては、専用のコンピューティングリソースを使用して個別にスケーリングする Redshift Spectrum を実行している RA3 クラスターと比較して、RG のパフォーマンスが低下しているように見えることがあります。クエリのパフォーマンスが遅い場合は、ノードを追加するか、より大きな RG インスタンスサイズにアップグレードすることを検討してください。

DC2 および RA3 に Redshift Spectrum を使用する

DC2 および RA3 でプロビジョニングされたクラスターでは、Redshift Spectrum は、クラスターに依存しない専用の Amazon Redshift サーバー上にあります。Redshift Spectrum は、述語フィルタリングや集計など、大量の演算を行う多くのタスクを Redshift Spectrum レイヤーにプッシュします。また、Redshift Spectrum では、インテリジェントなスケーリングにより、超並列処理を活用することもできます。

Redshift スペクトラムとデータレイクの操作方法など、Redshift スペクトラムの詳細については、Amazon Redshift データベース開発者ガイドの「Amazon Redshift Spectrum の開始方法」を参照してください。

リモートデータベースマネージャーでのデータのクエリの実行

フェデレーティッドクエリを使用して、Amazon RDS データベースおよび Amazon Aurora データベースのデータを Amazon Redshift データベースのデータに結合できます。Amazon Redshift を使用すると、処理対象のデータを直接 (移動せずに) クエリしたり、変換を適用したり、そのデータを Redshift テーブルに挿入したりできます。フェデレーティッドクエリの計算の一部は、リモートデータソースに分散されます。

フェデレーティッドクエリを実行するために、Amazon Redshift はまずリモートデータソースに接続します。Amazon Redshift は次に、リモートデータソース内のテーブルに関するメタデータを取得し、クエリを発行し、その結果の行を取得します。その後、Amazon Redshift は、結果の行を Amazon Redshift のコンピューティングノード間で分散してさらに処理を行います。

フェデレーティッドクエリのための環境のセットアップ方法については、Amazon Redshift データベース開発者ガイドで、以下のトピックを参照してください。

- PostgreSQL への横串検索を使用した開始方法
- MySQL への横串検索の使用開始方法

他の Amazon Redshiftデータベースのデータへのアクセス

Amazon Redshift のデータ共有を使用すると、Amazon Redshift クラスターもしくは AWS アカウント間で、読み取り目的でのライブデータの共有を、より安全かつ簡単に行えます。手動によるコピーや移動なしで、Amazon Redshift クラスター全体にわたるデータに対する詳細で高性能のアクセスを、即座に行えるようになります。ユーザーは、Amazon Redshift クラスターに更新された一貫性のある最新情報を確認できます。データベース、スキーマ、テーブル、ビュー (通常ビュー、遅延バインディングビュー、マテリアライズドビューを含む)、および SQL ユーザー定義関数 (UDF)など、さまざまなレベルでデータを共有することが可能です。

Amazon Redshift のデータ共有は、次のユースケースで特に便利です。

- 複数のビジネスインテリジェンス (BI) クラスターまたは分析クラスターとデータを共有する、一元的な抽出、変換、ロード (ETL) クラスターを使用します。このアプローチは、個々のワークロードに対して読み込みワークロードの分離とチャージバックを提供します。
- 環境間でのデータの共有 – 開発、テスト、本番稼働環境間でデータを共有します。さまざまなレベルの詳細なデータを共有することで、チームの俊敏性を向上させることができます。

データ共有の詳細については、「Amazon Redshift データベース開発者ガイド」の「データ共有タスクの管理」を参照してください。

Amazon Redshift データを使用した機械学習モデルのトレーニング

Amazon Redshift 機械学習 (Amazon Redshift ML) を使用すると、モデルのトレーニングのためのデータを Amazon Redshift に提供することができます。次に、Amazon Redshift ML は、入力データのパターンをキャプチャするモデルを作成します。その後、これらのモデルを使用して、追加のコス

トを発生させることなく、新しい入力データの予測を生成できます。Amazon Redshift ML を使用すると、SQL ステートメントを使用して機械学習モデルをトレーニングし、予測のために SQL クエリでそれらを呼び出すことができます。パラメータを繰り返し変更し、トレーニングデータを改善することで、予測の精度を継続的に向上させることができます。

Amazon Redshift ML を使用すると、SQL ユーザーは、使い慣れた SQL コマンドを使用して、機械学習モデルを簡単に作成、トレーニング、デプロイできます。Amazon Redshift ML では、Amazon Redshift クラスター内のデータを使用して、Amazon SageMaker AI Autopilot でモデルをトレーニングし、自動的に最適なモデルを取得できます。その後、モデルはローカライズされ、Amazon Redshift データベース内から予測が行えるようになります。

Amazon Redshift ML の詳細については、Amazon Redshift データベース開発者ガイドの「Amazon Redshift 機械学習の開始方法」を参照してください。

Amazon Redshift の概念の説明

Amazon Redshift Serverless を使用すると、プロビジョニングされたデータウェアハウスをすべて設定しなくても、データにアクセスして分析することができます。リソースは自動的にプロビジョニングされて、データウェアハウス容量はインテリジェントにスケーリングされ、要求が厳しく、予測不可能なワークロードであっても高速なパフォーマンスを実現します。データウェアハウスがアイドル状態のときには課金されず、使用した分のみ支払います。Amazon Redshift クエリエディタ v2 またはお好みのビジネスインテリジェンス (BI) ツールで、データをロードしてクエリを直ちに開始することができます。使いやすい管理不要の環境で、最高のコストパフォーマンスと使い慣れた SQL 機能をお楽しみください。

Amazon Redshift を初めて使用する方には、以下のセクションを初めに読むことをお勧めします。

- Amazon Redshift Serverless 機能の概要 - このトピックでは、Amazon Redshift Serverless の概要と、その主要な機能について説明します。
- 主なサービスと料金設定 — この製品詳細ページでは、Amazon Redshift Serverless の主なサービスと料金設定を確認できます。
- Amazon Redshift Serverless データウェアハウスの使用を開始。 — このトピックでは、Amazon Redshift Serverless データウェアハウスを作成する方法と、クエリエディタ v2 を使用してデータのクエリを開始する方法について説明します。

Amazon Redshift リソースを手動で管理したい場合は、データクエリのニーズに合わせてプロビジョニングされたクラスターを作成することができます。詳細については、「Amazon Redshift クラスター」を参照してください。

組織が適格であり、Amazon Redshift Serverless が利用できない AWS リージョン でクラスターが作成されている場合、Amazon Redshift 無料トライアルプログラムでクラスターを作成できる場合があります。[このクラスターを何に使用する予定ですか?] という質問に対して、[本番稼働用] または [無料トライアル] のいずれかを選択します。[無料トライアル] を選択したときには、dc2.large ノードタイプの設定を作成します。無料トライアルの選択に関する詳細については、「Amazon Redshift 無料トライアル」を参照してください。Amazon Redshift Serverless が利用可能な AWS リージョン の一覧については、Amazon Web Services 全般のリファレンスの「Redshift Serverless API」に記載されている Amazon Redshift エンドポイントを参照してください。

以下に、Amazon Redshift Serverless の主要な概念をいくつか示します。

- 名前空間 - データベースオブジェクトとユーザーのコレクションです。名前空間は、スキーマ、テーブル、ユーザー、データ共有、スナップショットなど、Amazon Redshift Serverless で使用するすべてのリソースをグループ化します。
- ワークグループ - コンピューティングリソースの集合です。ワークグループには、Amazon Redshift Serverless が計算タスクを実行するために使用するコンピューティングリソースが含まれています。このようなリソースの例としては、Redshift 処理ユニット (RPU)、セキュリティグループ、使用制限などがあります。ワークグループには、Amazon Redshift Serverless コンソール、AWS Command Line Interface、または Amazon Redshift Serverless API を使用して設定できるネットワークとセキュリティ設定があります。

名前空間とワークグループリソースの設定の詳細については、「名前空間の使用」と「ワークグループの使用」を参照してください。

以下に、Amazon Redshift でプロビジョニングされたクラスターの主要な概念をいくつか示します:

- クラスター – Amazon Redshift データウェアハウスの中核となるインフラストラクチャコンポーネントは、クラスターです。

  クラスターは、1 つまたは複数のコンピューティングノードで構成されます。コンピューティングノードは、コンパイルされたコードを実行します。

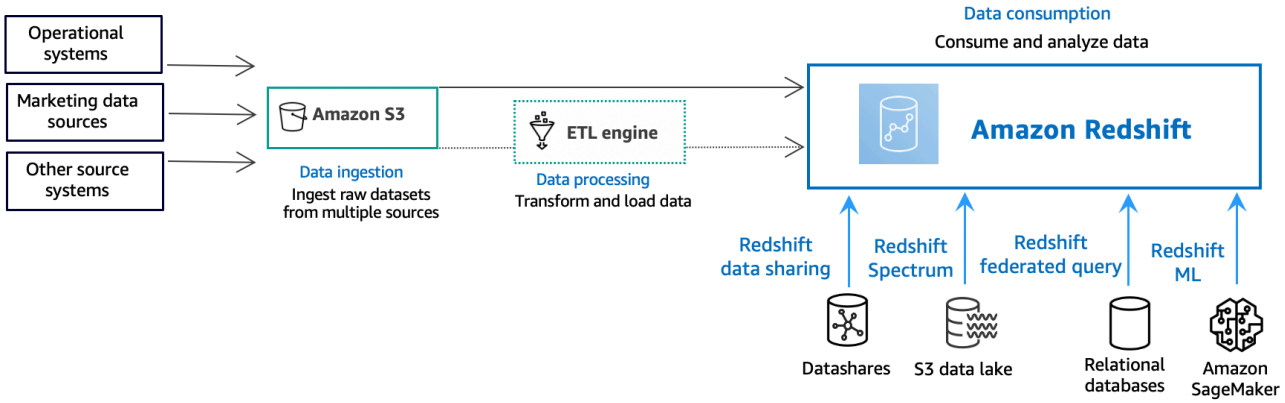
  クラスターが 2 つ以上のコンピューティングノードでプロビジョニングされている場合、追加のリーダーノードがコンピューティングノードを調整します。リーダーノードは、ビジネスインテリジェンスツールやクエリエディタなどのアプリケーションとの外部通信を処理します。クライアントアプリケーションはリーダーノードとのみ直接通信します。コンピューティングノードは外部アプリケーションに対して透過的です。
- データベース – クラスターには、1 つ以上のデータベースが含まれています。

  ユーザーデータは、コンピューティングノード上の 1 つ以上のデータベースに保存されます。SQL クライアントはリーダーノードと通信し、リーダーノードは実行中のクエリをコンピューティングノードと調整します。コンピューティングノードとリーダーノードの詳細については、データウェアハウスシステムのアーキテクチャを参照してください。データベース内では、ユーザーデータは 1 つ以上のスキーマに編成されます。

  Amazon Redshift はリレーショナルデータベース管理システム (RDBMS) であり、他の RDBMS アプリケーションと互換性があります。それは、標準的な RDBMS と同じ機能、データの挿入や削除といったオンライントランザクション処理 (OLTP) 機能を提供します。Amazon Redshift は、データセットのハイパフォーマンスなバッチ分析とレポート作成にも最適化されています。

以下に、Amazon Redshift の一般的なデータ処理フローの説明とともに、フローのさまざまな部分の説明を示します。Amazon Redshift システムアーキテクチャーの詳細については、データウェアハウスシステムのアーキテクチャを参照してください。

次の図表は、Amazon Redshift の一般的なデータ処理フローを示しています。



Amazon Redshift データウェアハウスは、エンタープライズクラスのリレーショナルデータベースのクエリおよび管理を行うためのシステムです。Amazon Redshift では、ビジネスインテリジェンス (BI) や、レポート作成、データ処理、分析用のツールなど、多種類のアプリケーションとのクライアント接続をサポートしています。分析クエリを実行するときは、大量のデータを複数のステージからなる操作で取得、比較、および評価して、最終的な結果を生成します。

データインジェストレイヤーでは、さまざまなタイプのデータソースが、構造化データ、半構造化データ、非構造化データをデータストレージレイヤーに継続的にアップロードします。このデータストレージエリアは、さまざまな消費準備状態のデータを保存するステージングエリアとして機能します。ストレージの例として、Amazon Simple Storage Service (Amazon S3) バケットがあります。

オプションのデータ処理レイヤーでは、ソースデータは、抽出、変換、ロード (ETL)、または抽出、ロード、変換 (ELT) パイプラインを使用して、前処理、検証、変換を行います。これらの生のデータセットは、ETL オペレーションを使用して洗練されます。ETLエンジンの例はAWS Glueです。

データ消費レイヤーでは、データが Amazon Redshift クラスターにロードされ、そこで分析ワークロードを実行できます。

分析ワークロードの例については、「外部データソースへのクエリ」を参照してください。

Amazon Redshift について学習するためのその他のリソース

Amazon Redshift Serverless の詳細については、以下の Amazon Redshift リソースを使用して、このガイドで説明した概念についてさらに詳細な説明をご覧になることをお勧めします。

- 関連動画:これらの動画は、Amazon Redshift の機能について学習するのに役立ちます。
  - Amazon Redshift Serverless をさらに詳細に理解するため、次の動画をご覧ください。Amazon Redshift Serverless について 90 秒で説明。
  - サーバーレスデータウェアハウスを設定してデータのクエリを開始する方法については、次の動画をご覧ください。Amazon Redshift Serverless の開始方法。
- 「Amazon Redshift 管理ガイド」: このガイドは、「Amazon Redshift 入門ガイド」を基に作成されています。Amazon Redshift Serverless および Amazon Redshift でプロビジョニングされたクラスターの作成、管理、および監視に関する概念とタスクの詳細情報を提供します。
- Amazon Redshift データベースデベロッパーガイド: このガイドも、Amazon Redshift 入門ガイドに基づいて作成されたものです。データウェアハウスを構成するデータベースの構築、クエリ、および保守に関する詳細情報をデータベースデベロッパー向けに提供します。
  - SQL リファレンス: このトピックでは、Amazon Redshift の SQL コマンドと関数リファレンスについて説明します。
  - システムテーブルとビューのリファレンス: このトピックでは、Amazon Redshift のシステムテーブルとビューについて説明します。
- Amazon Redshift のチュートリアル: このトピックでは、Amazon Redshift の機能に関するチュートリアルを示します。
  - Amazon S3 からデータをロードする: このチュートリアルでは、Amazon S3 バケット内のデータファイルから、Amazon Redshift データベーステーブルにデータをロードする方法について説明しています。
  - データ共有の開始方法: このセクションでは、他の Amazon Redshift クラスターのデータを共有してアクセスする方法について説明します。
  - Amazon Redshift での空間 SQL 関数の使用: このチュートリアルでは、Amazon Redshift でいくつかの空間 SQL 関数を使用する方法を説明しています。
  - Amazon Redshift Spectrum を使用したネストデータのクエリ: このチュートリアルでは、Redshift Spectrum を使って、外部テーブルを使用して、Parquet、ORC、JSON、Ion のファイル形式でネストされたデータをクエリする方法について説明します。
  - 手動ワークロード管理 (WLM) キューの設定: このチュートリアルでは、Amazon Redshift で手動ワークロード管理 (WLM) を設定する方法について説明します。

  - Amazon Redshift ML の開始方法: このセクションでは、ユーザーが使い慣れた SQL コマンドを使用して、機械学習モデルを作成、トレーニング、デプロイする方法について説明します。
- 最新情報: このウェブページには、Amazon Redshift の新機能と製品アップデートが記載されています。

ドキュメント履歴

> **Note**
>
> Amazon Redshift の新機能の説明については、「最新情報」を参照してください。

次の表で、Amazon Redshift 入門ガイドにおける重要なドキュメントの変更点について説明します。

変更	説明	リリース日
ドキュメントの更新	Query Editor v2 マネージドポリシーの変更と、サーバーレス名前空間とワークグループのアクセス許可の改善を反映するようにガイドを更新しました。	2024 年 2 月 21 日
ドキュメントの更新	最新のコンソールインターフェイスの改善と Query Editor v2 の機能強化を反映するようにスクリーンショットと手順を更新しました。	2023 年 3 月 11 日
新機能	Amazon Redshift Serverless の開始方法の手順とワークフローを含むようにガイドを更新しました。サーバーレスデータウェアハウスの作成と管理に関する包括的なセクションを追加しました。	2022 年 7 月 12 日
ドキュメントの更新	Query Editor v2 をプライマリクエリインターフェイスとして反映するようにガイドを更新し、レガシークエリエディタへの参照を置き換えました。	2022 年 2 月
ドキュメントの更新	ガイドを更新し、一般的なデータベースタスクの開始方法、データレイクのクエリ、リモートソースにあるデータのクエリ、データの共有、Amazon Redshift データを使用した機械学習モデルのトレーニングに関する、新しいセクションを追加しました。	2021 年 6 月 30 日
新機能	新しいサンプルロード手順について説明するようにガイドを更新しました。	2021 年 6 月 4 日

変更	説明	リリース日
ドキュメントの更新	ガイドを更新して、元の Amazon Redshift コンソールを削除し、ステップフローを改善しました。	2020 年 8 月 14 日
新しいコンソール	新しい Amazon Redshift コンソールについて説明するようにガイドを更新しました。	2019 年 11 月 11 日
新機能	クイック起動クラスター手順について説明するようにガイドを更新しました。	2018 年 8 月 10 日
新機能	Amazon Redshift ダッシュボードからクラスターを起動するようにガイドを更新しました。	2015 年 7 月 28 日
新機能	新しいノードタイプ名を使用するようにガイドを更新しました。	2015 年 6 月 9 日
ドキュメントの更新	VPC セキュリティグループの設定のスクリーンショットと手順を更新しました。	2015 年 4 月 30 日
ドキュメントの更新	現在のコンソールに合致するようにスクリーンショットと手順を更新しました。	2014 年 11 月 12 日
ドキュメントの更新	見つけやすくするために、Amazon S3 からのデータのロードに関する説明を独立したセクションに移動し、次のステップセクションを最終ステップに移動しました。	2014 年 5 月 13 日
ドキュメントの更新	ようこそページを削除し、そのコンテンツをメインの使用開始ページに組み込みました。	2014 年 3 月 14 日
ドキュメントの更新	これは、カスタマーフィードバックとサービスの更新情報を反映した Amazon Redshift 入門ガイドの新しいリリースです。	2014 年 3 月 14 日
新規ガイド	これは Amazon Redshift 入門ガイドの初版リリースです。	2013 年 2 月 14 日