Signing up for AWS

If you don't already have an AWS account, sign up for one. If you already have an account, you can skip this prerequisite and use your existing account.

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   When you sign up for an AWS account, an AWS account root user is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to an administrative user, and use only the root user to perform tasks that require root user access.

Creating a data warehouse with Amazon Redshift Serverless

The first time you log in to the Amazon Redshift Serverless console, you are prompted to access the getting started experience, which you can use to create and manage serverless resources. In this guide, you'll create serverless resources using Amazon Redshift Serverless's default settings.

For more granular control of your setup, choose **Customize settings**.

To configure with default settings:

1. Sign in to the AWS Management Console and open the Amazon Redshift console at https://console.aws.amazon.com/redshiftv2/.

   Choose **Try Amazon Redshift Serverless**.
2. Under **Configuration**, choose **Use default settings**. Amazon Redshift Serverless creates a default namespace with a default workgroup associated to this namespace. Choose **Save configuration**.

   The following screenshot shows the default settings for Amazon Redshift Serverless.

Get started with Amazon Redshift Serverless

To start using Amazon Redshift Serverless, set up your serverless data warehouse and create a database.
You will receive $0 credit towards your Redshift Serverless usage in this account.

Configuration Info

- ● **Use default settings**
  Default settings have been defined to help you get started. You can change them at any time later.

- ○ **Customize settings**
  Customize your settings for your specific needs.

▼ How it works

Using the default settings

Amazon Redshift Serverless creates a default namespace and workgroup. This configuration uses the default settings and becomes active when you associate the default workgroup to the default namespace.

Customizing the settings

Amazon Redshift Serverless creates a default namespace and workgroup. This configuration becomes active when you associate the default workgroup to the default namespace.

Namespace Info

Namespace is a collection of database objects and users. Data properties include database name and password, permissions, and encryption and security.

⚠ Your data is encrypted by default with an AWS owned key. To choose a different key, choose **Customize settings**.

Namespace name
default

Database name and password

Database name
dev

Admin user credentials
Created based on IAM credentials

Permissions

Default IAM role
arn:aws:iam::694237154968:role/service-role/AmazonRedshift-CommandsAccessRole-20220809T211900 ☒

Encryption and security

AWS KMS encryption
AWS owned KMS key

Audit logging
Off

Workgroup Info

Workgroup is a collection of compute resources from which an endpoint is created. Compute properties include network and security settings.

Workgroup name
default

Network and security

Virtual private cloud (VPC)
example-vpc ☒

VPC security group
example-security-group ☒

Subnet
example-subnet-1
example-subnet-2

Enhanced VPC routing
Off

Cancel   **Save configuration**

3. After setup completes, choose **Continue** to go to your **Serverless dashboard**. You can see that the serverless workgroup and namespace are available.

Loading sample data

Now that you've set up your data warehouse with Amazon Redshift Serverless, you can use the Amazon Redshift query editor v2 to load sample data.

1. To launch query editor v2 from the Amazon Redshift Serverless console, choose **Query data**. When you invoke query editor v2 from the Amazon Redshift Serverless console, a new browser tab opens with the query editor. The query editor v2 connects from your client machine to the Amazon Redshift Serverless environment.

2. If you're launching query editor v2 for the first time, you must configure AWS KMS encryption before you can proceed. Optionally, you can also specify the URI to an S3 bucket for data loading later. After doing so, choose **Configure account**.

Configure account
To start using Redshift query editor v2, configure the account.

AWS KMS encryption
Your data is encrypted by default with a key that AWS owns and manages for you. To choose a different key, customize your encryption settings.

ⓘ You can't change the AWS KMS encryption settings after configuring your account. ✕

☐ Customize encryption settings (advanced)

S3 bucket - *optional*
Enter an Amazon S3 path for this account. This path can be used for features such as load data from a local file store. Account users and the S3 bucket must be configured with the proper permissions to secure user's data. Learn more ↗
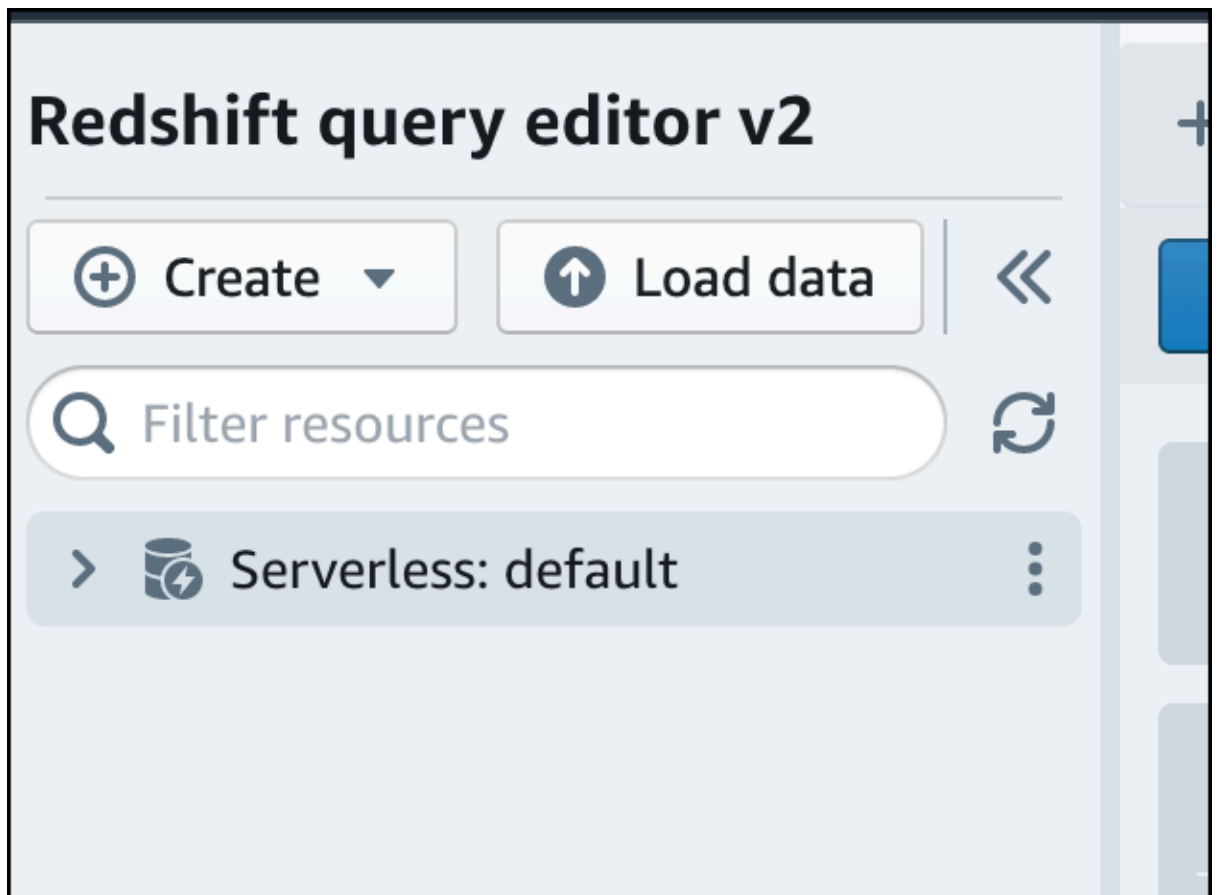
S3 URI

s3://bucket[/prefix]

Cancel   **Configure account**

To learn more about configuring the query editor v2, including which permissions are needed, see Configuring your AWS account in the *Amazon Redshift Management Guide*.

3. To connect to a workgroup, choose the workgroup name in the tree-view panel.

4. When connecting to a new workgroup for the first time within query editor v2, you must select the type of authentication to use to connect to the workgroup. For this guide, leave **Federated user** selected, and choose **Create connection**.

Connect to default   ✕

Authentication Learn more

◉ Federated user
The query editor v2 generates a temporary password to connect to the database.

◯ Database user name and password
Provide a database user and password for the database that you are connecting to. The query editor v2 stores your credentials in AWS Secrets Manager on your behalf.
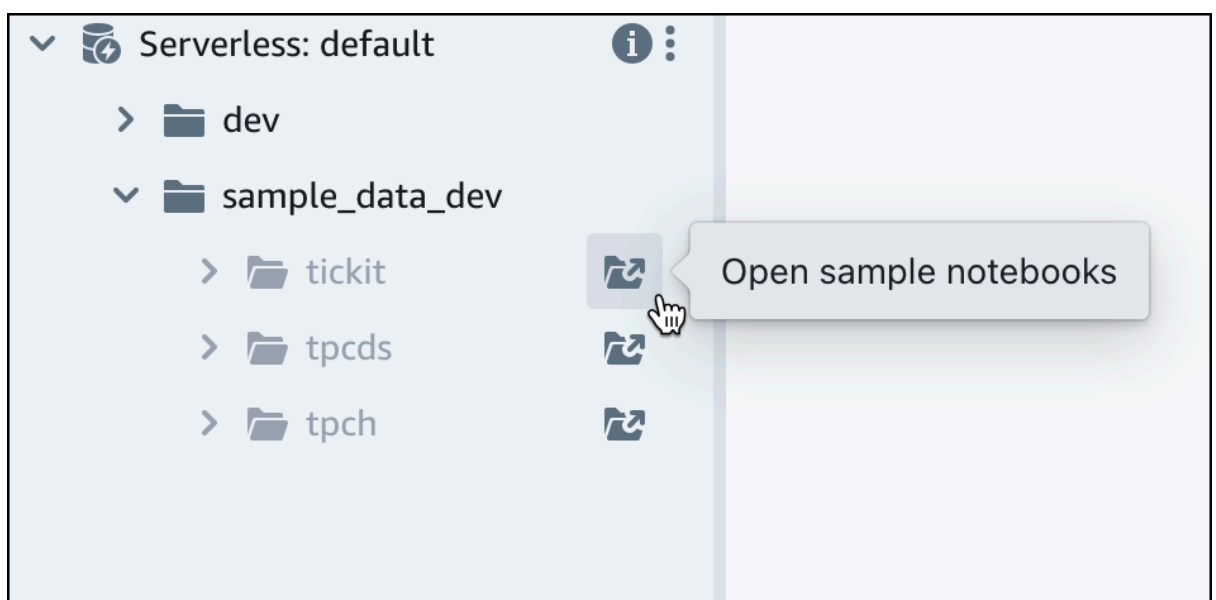
Database

dev

The database name must be 1-64 characters. Valid characters are lowercase alphanumeric characters.

[Cancel] [Create connection]

Once you are connected, you can choose to load sample data from Amazon Redshift Serverless or from an Amazon S3 bucket.

5. Under the Amazon Redshift Serverless default workgroup, expand the **sample_data_dev** database. There are three sample schemas corresponding to three sample datasets that you can load into the Amazon Redshift Serverless database. Choose the sample dataset that you want to load, and choose **Open sample notebooks**.

6. When loading data for the first time, query editor v2 will prompt you to create a sample database. Choose **Create**.
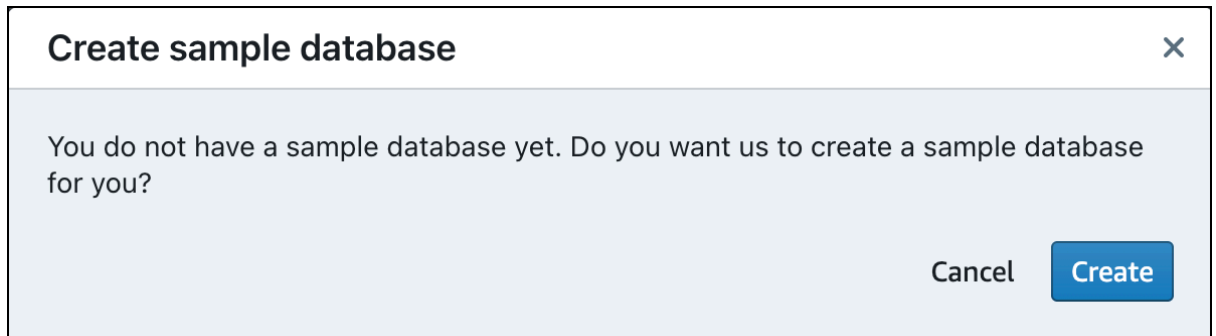

Running sample queries

After setting up Amazon Redshift Serverless, you can start using a sample dataset in Amazon Redshift Serverless. Amazon Redshift Serverless automatically loads the sample dataset, such as the tickit dataset, and you can immediately query the data.

- Once Amazon Redshift Serverless finishes loading the sample data, all of the sample queries are loaded in the editor. You can choose **Run all** to run all of the queries from the sample notebooks.
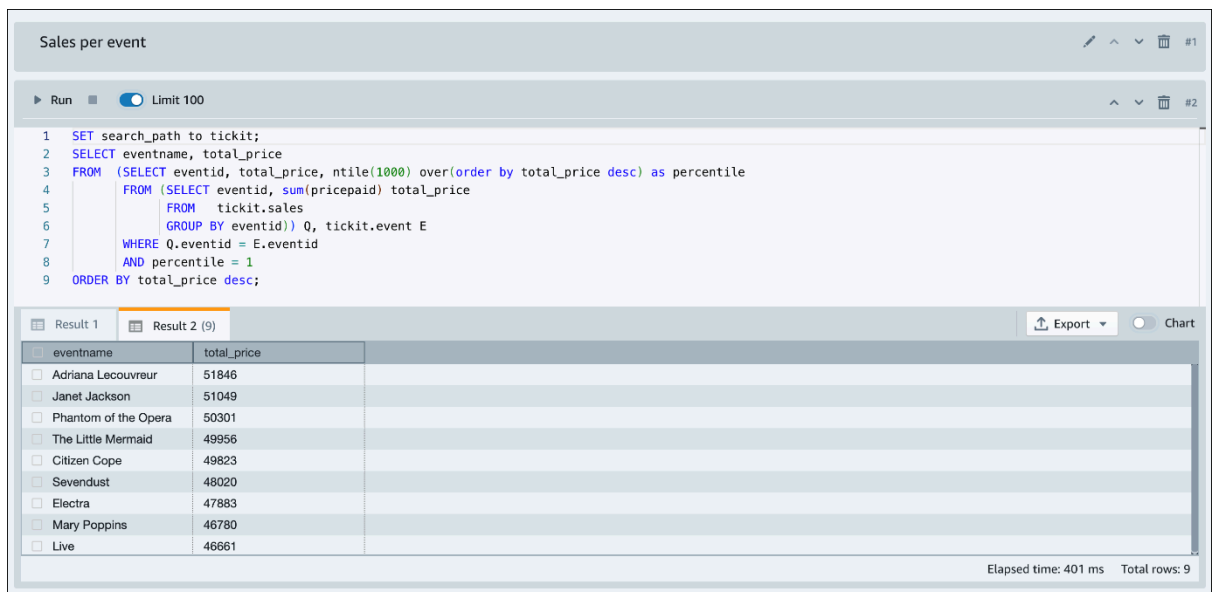
You can also export the results as a JSON or CSV file or view the results in a chart.
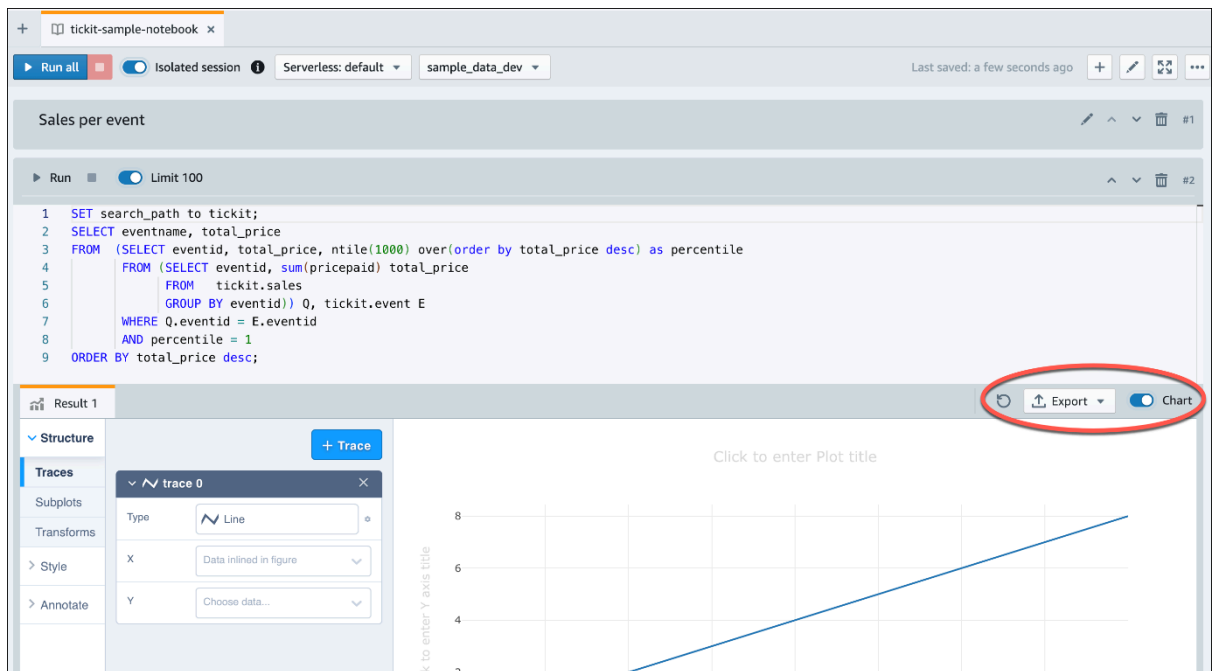
You can also load data from an Amazon S3 bucket. See [Loading in data from Amazon S3](#) to learn more.

Loading in data from Amazon S3

After creating your data warehouse, you can load data from Amazon S3.

At this point, you have a database named `dev`. Next, you will create some tables in the database, upload data to the tables, and try a query. For your convenience, the sample data that you load is available in an Amazon S3 bucket.

1. Before you can load data from Amazon S3, you must first create an IAM role with the necessary permissions and attach it to your serverless namespace. To do so, choose **Namespace configuration** from the navigation menu, and then choose **Security and encryption**. Then, choose **Manage IAM roles**.

2. Expand the **Manage IAM roles** menu, and choose **Create IAM role**.

3. Choose the level of S3 bucket access that you want to grant to this role, and choose **Create IAM role as default**.

Create the default IAM role   ✕

ⓘ Associate an IAM role so that your serverless endpoint can LOAD and UNLOAD data. You can create an IAM role as the default for this configuration that has the **AmazonRedshiftAllCommandsFullAccess** ↗ policy attached. This policy includes permissions to run SQL commands to COPY, UNLOAD, and query data with Amazon Redshift Serverless. This policy also grants permissions to run SELECT statements for related services, such as Amazon S3, Amazon CloudWatch logs, Amazon SageMaker, and AWS Glue. You won't be able to run these SQL commands without an IAM role attached to your namespace.

Specify an S3 bucket for the IAM role to access
To create a new bucket, visit S3 ↗

○ No additional S3 bucket
   Create the IAM role without specifying S3 buckets.

◉ Any S3 bucket
   Allow users that have access to your Redshift Serverless data to also access any S3 bucket and its contents in your AWS account.

○ Specific S3 buckets
   Specify one or more S3 buckets that the IAM role being created has permission to access.

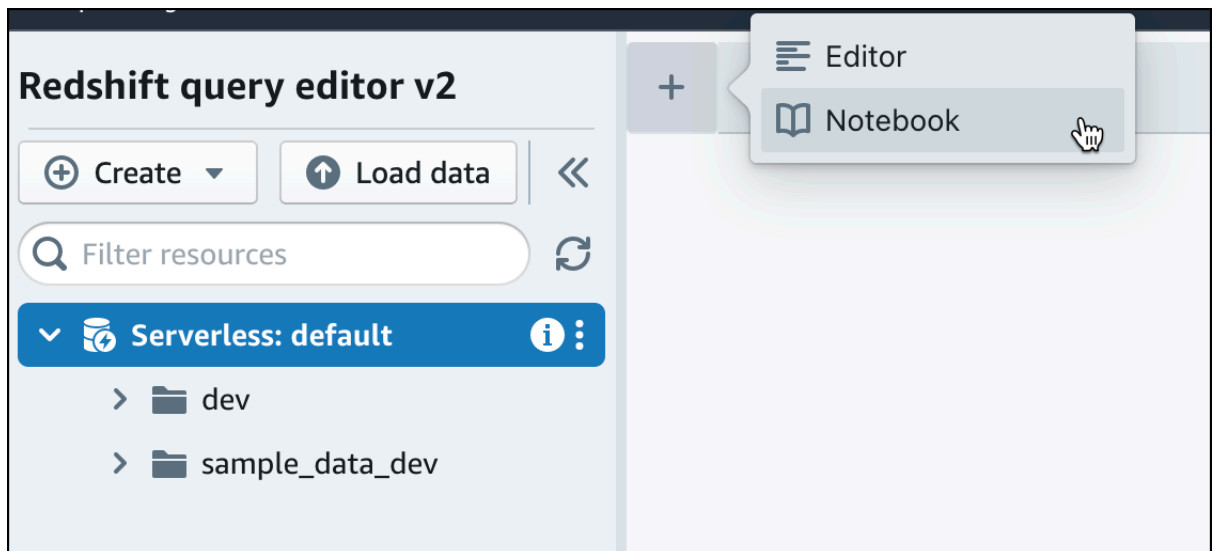   Cancel   Create IAM role as default

4. Choose **Save changes**. You can now load sample data from Amazon S3.

The following steps use data within a public Amazon Redshift S3 bucket, but you can replicate the same steps using your own S3 bucket and SQL commands.

Load sample data from Amazon S3

1. In query editor v2, choose ✛ Add, then choose **Notebook** to create a new SQL notebook.

2. Switch to the `dev` database.
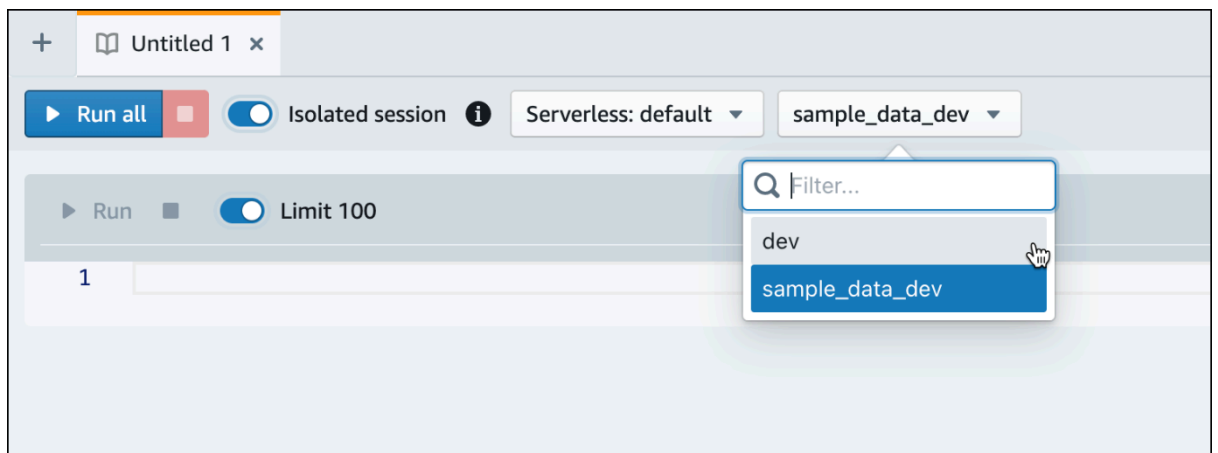
3. Create tables.

   If you are using the query editor v2, copy and run the following create table statements to create tables in the `dev` database. For more information about the syntax, see CREATE TABLE in the *Amazon Redshift Database Developer Guide*.

```
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. In the query editor v2, create a new SQL cell in your notebook.

5. Now use the COPY command in query editor v2 to load large datasets from Amazon S3 or Amazon DynamoDB into Amazon Redshift. For more information about COPY syntax, see COPY in the *Amazon Redshift Database Developer Guide*.

You can run the COPY command with some sample data available in a public S3 bucket. Run the following SQL commands in the query editor v2.

```
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. After loading data, create another SQL cell in your notebook and try some example queries. For more information on working with the SELECT command, see [SELECT](#) in the *Amazon Redshift Developer Guide*. To understand the sample data's structure and schemas, explore using the query editor v2.

```
7.  -- Find top 10 buyers by quantity.

8.  SELECT firstname, lastname, total_quantity

9.  FROM   (SELECT buyerid, sum(qtysold) total_quantity

10.          FROM  sales

11.          GROUP BY buyerid

12.          ORDER BY total_quantity desc limit 10) Q, users

13.    WHERE Q.buyerid = userid
```

```
14.    ORDER BY Q.total_quantity desc;

15.

16.    -- Find events in the 99.9 percentile in terms of all time
       gross sales.

17.    SELECT eventname, total_price

18.    FROM  (SELECT eventid, total_price, ntile(1000) over(order
       by total_price desc) as percentile

19.          FROM (SELECT eventid, sum(pricepaid) total_price

20.                FROM   sales

21.                GROUP BY eventid)) Q, event E

22.          WHERE Q.eventid = E.eventid

23.          AND percentile = 1

    ORDER BY total_price desc;
```

Now that you've loaded in data and ran some sample queries, you can explore other areas of Amazon Redshift Serverless. See the following list to learn more about how you can use Amazon Redshift Serverless.

- You can load data from an Amazon S3 bucket. See Loading data from Amazon S3 for more information.
- You can use the query editor v2 to load in data from a local character-separated file that is smaller than 5 MB. For more information, see Loading data from a local file.
- You can connect to Amazon Redshift Serverless with third-party SQL tools with the JDBC and ODBC driver. See Connecting to Amazon Redshift Serverless for more information.
- You can also use the Amazon Redshift Data API to connect to Amazon Redshift Serverless. See Using the Amazon Redshift Data API for more information.
- You can use your data in Amazon Redshift Serverless with Redshift ML to create machine learning models with the CREATE MODEL command. See Tutorial: Building customer churn models to learn how to build a Redshift ML model.
- You can query data from an Amazon S3 data lake without loading any data into Amazon Redshift Serverless. See Querying a data lake for more information.