Redshift Clusters

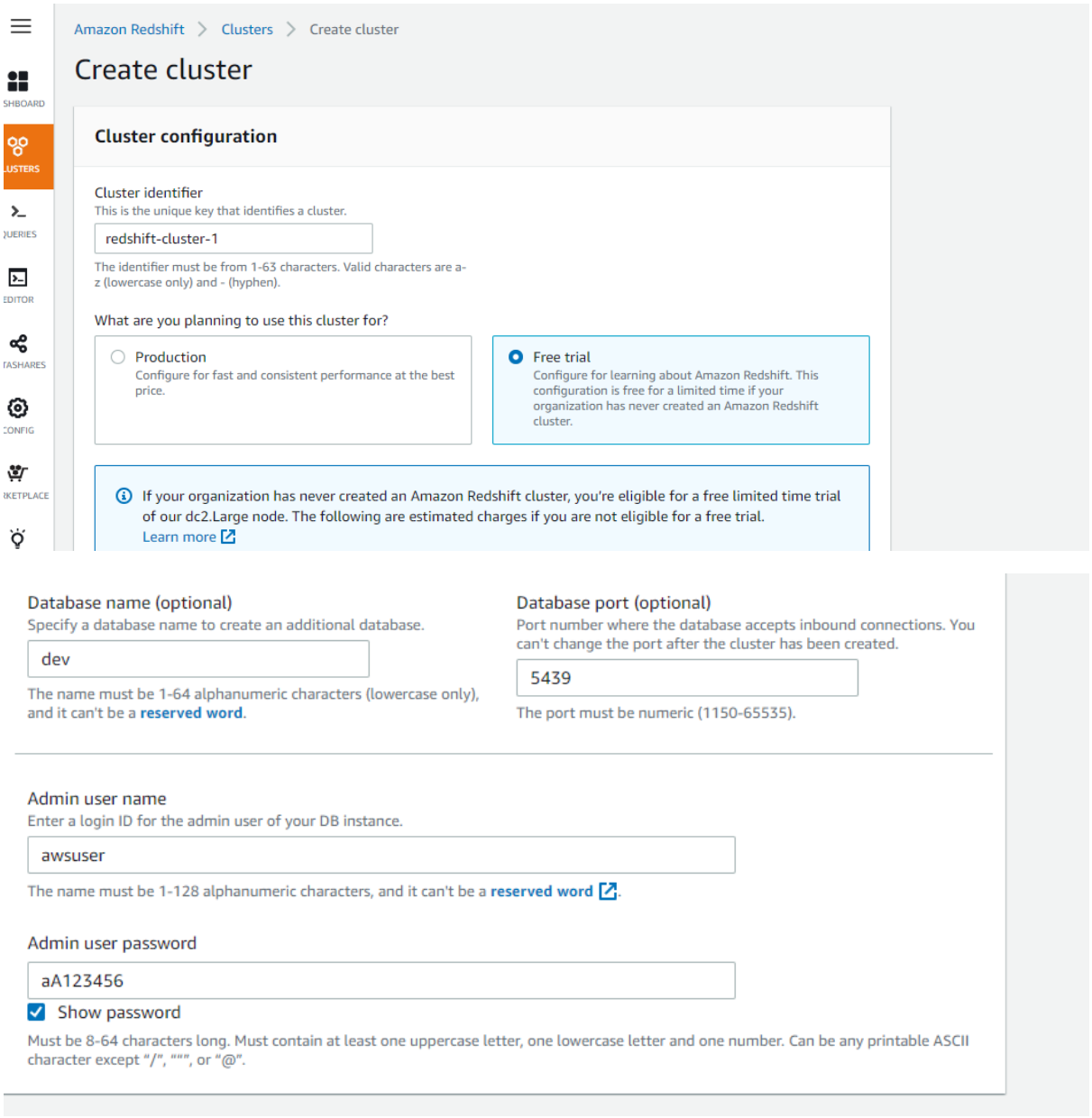
Step 1: Create an IAM Role

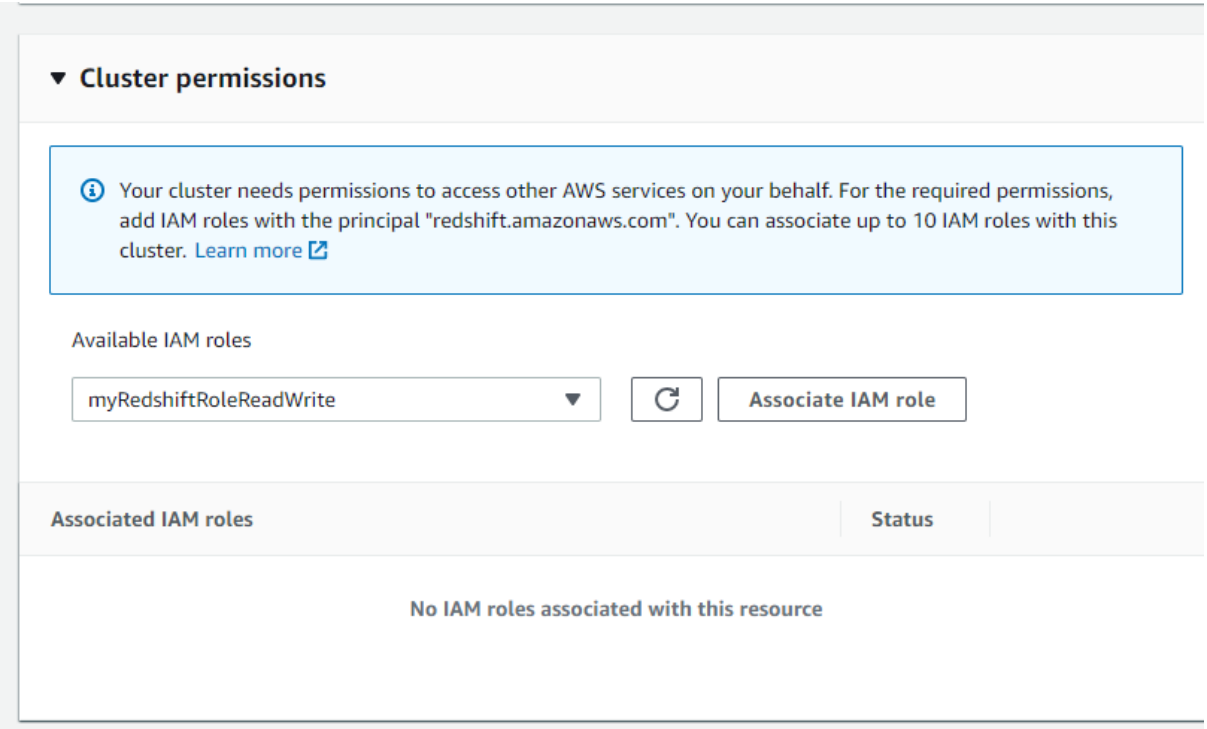
1. In the navigation pane, choose **Roles**.
2. Choose **Create role**.
3. In the **AWS Service** group, choose **Redshift.**
4. Under **Select your use case**, choose **Redshift - Customizable**, then choose **Next: Permissions**.
5. On the **Attach permissions policies** page, choose **AmazonS3ReadOnlyAccess**. You can leave the default setting for **Set permissions boundary**. Then choose **Next: Tags**.
6. The **Add tags** page appears. You can optionally add tags. Choose **Next: Review**.
7. For **Role name**, enter a name for your role. For this tutorial, enter `myRedshiftRole`ReadWrite (This role helps to read and write to S3 bucket)
8. Review the information, and then choose **Create Role**.
9. Choose the role name of the role that you just created.

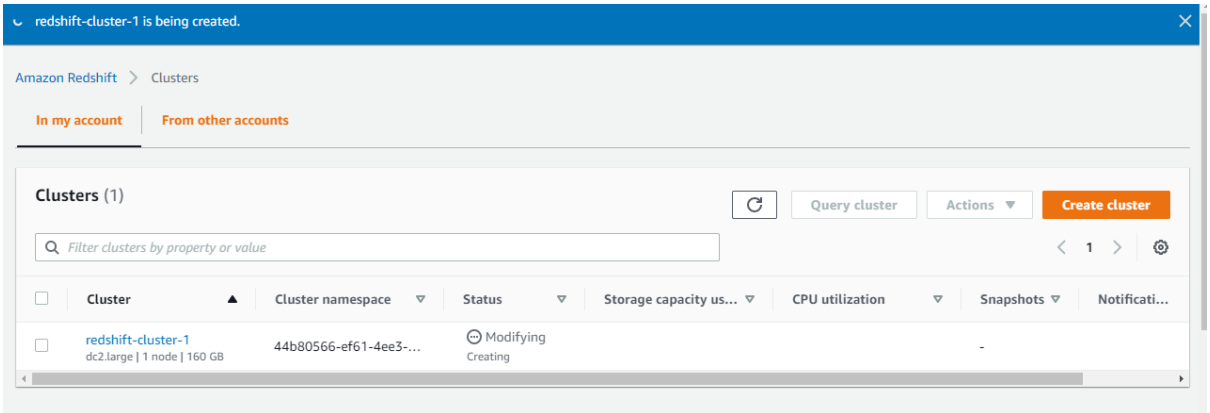
Step 2:

1. On the navigation menu, choose **CLUSTERS**, then choose **Create cluster**. The **Create cluster** page appears.
2. In the **Cluster configuration** section, specify values for **Cluster identifier**, **Node type**, and **Nodes**:

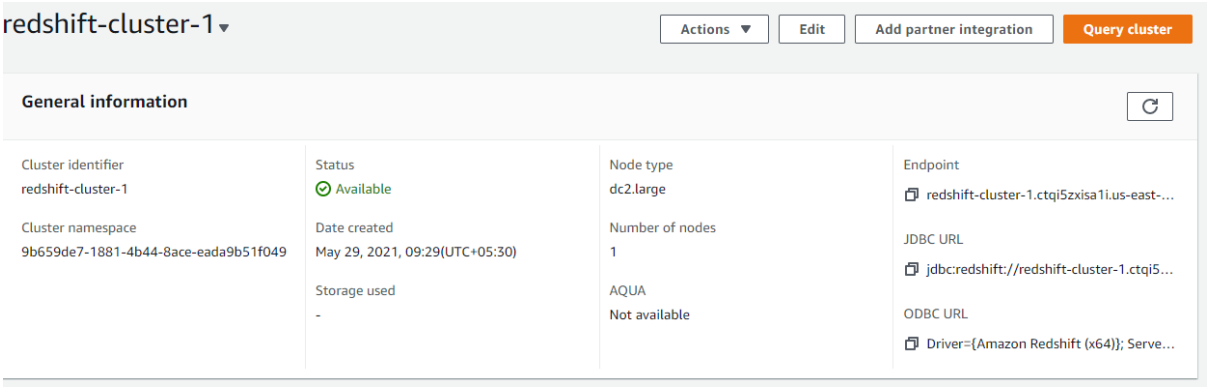
- **Cluster identifier**: Enter `redshift-cluster-1`
- What are you planning to use this cluster for? Select Free trial
- **Node Type dc2.large| 1 node will be selected**
- **Database name (optional): Enter** dev**.**
- **Database port (optional): Enter** 5439**.**
- **Master user name: Enter** awsuser**.**
- **Master user password: Enter a value for the password.**
- **In Cluster Permissions select Available IAM Role myRedshiftRoleReadWrite**


Amazon Redshift
Clusters
Create cluster
Create cluster
Cluster configuration
Cluster identifier
This is the unique key that identifies a cluster.
redshift-cluster-1
The identifier must be from 1-63 characters. Valid characters are a-z (lowercase only) and - (hyphen).
What are you planning to use this cluster for?
Production
Configure for fast and consistent performance at the best price.
Free trial
Configure for learning about Amazon Redshift. This configuration is free for a limited time if your organization has never created an Amazon Redshift cluster.
If your organization has never created an Amazon Redshift cluster, you're eligible for a free limited time trial of our dc2.Large node. The following are estimated charges if you are not eligible for a free trial.
Learn more
Database name (optional)
Specify a database name to create an additional database.
dev
The name must be 1-64 alphanumeric characters (lowercase only), and it can't be a reserved word.
Database port (optional)
Port number where the database accepts inbound connections. You can't change the port after the cluster has been created.
5439
The port must be numeric (1150-65535).
Admin user name
Enter a login ID for the admin user of your DB instance.
awsuser
The name must be 1-128 alphanumeric characters, and it can't be a reserved word.
Admin user password
aA123456
Show password
Must be 8-64 characters long. Must contain at least one uppercase letter, one lowercase letter and one number. Can be any printable ASCII character except "/", """, or "@".

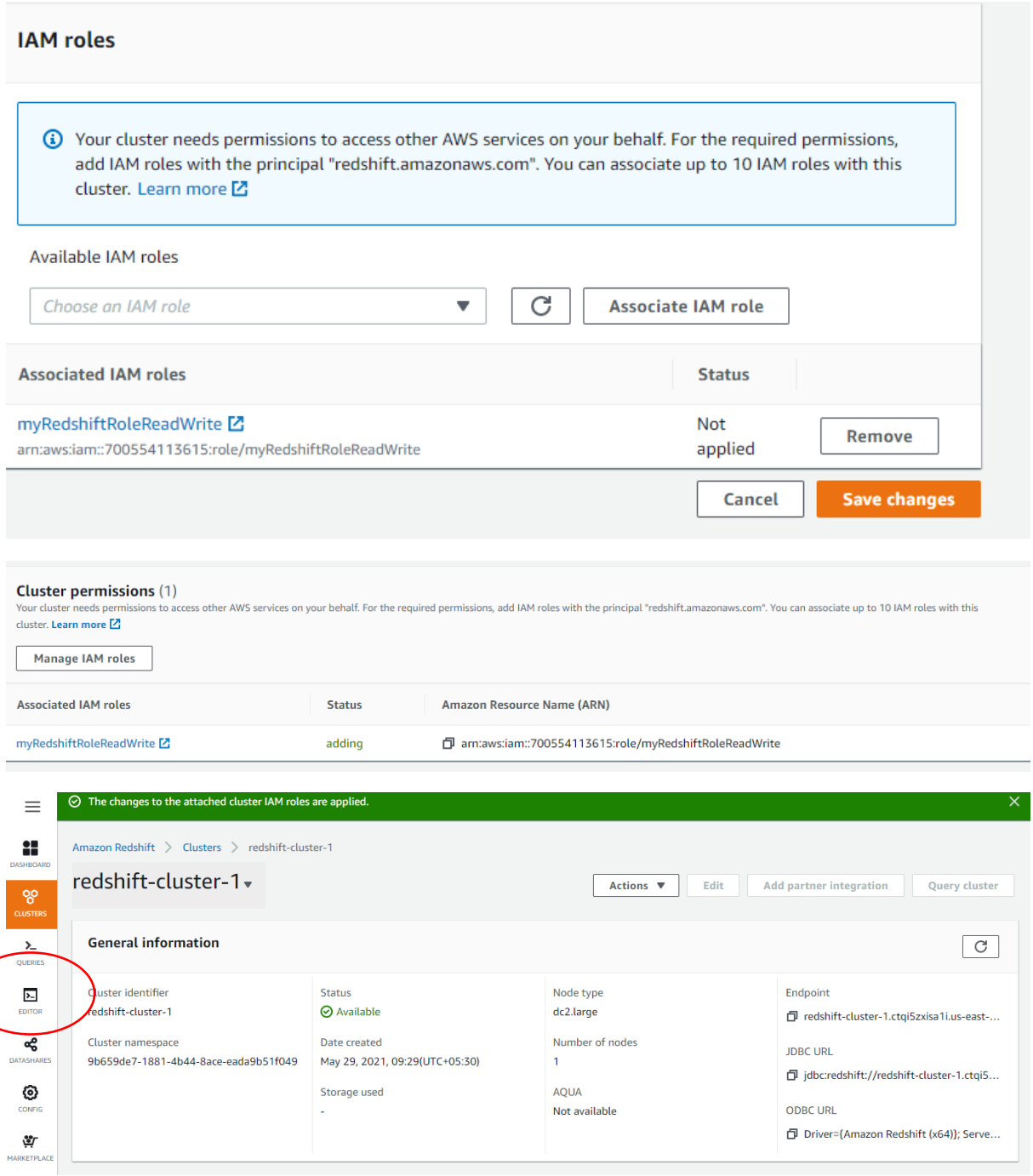




Please check the EndPoint details



Check for Cluster Permissions, if the IAM role was not attached properly attach it again.



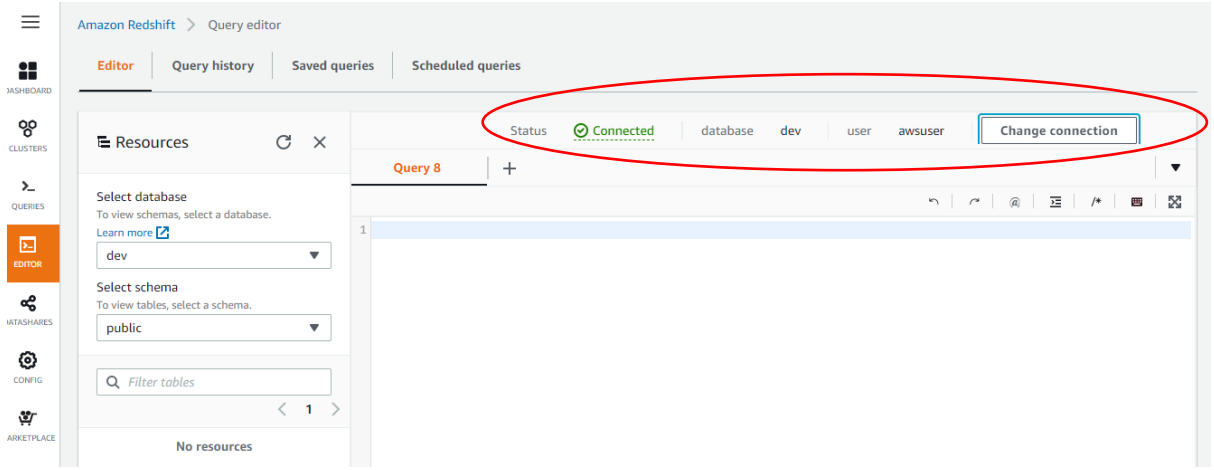
1. On the navigation menu, choose **EDITOR**, then connect to a database in your cluster.

   On the **Connect to database** page, there are two ways to authenticate, namely, **Temporary credentials** and **Secrets Manager**. For this tutorial, choose **Create a new connection** and **Temporary credentials**, then enter the values that you used when you created the cluster, as follows:

- **Cluster**: Choose `redshift-cluster-1`

- **Database name**: Enter `dev`.
- **Database user**: Enter `awsuser`.

  Then choose **Connect**.



2. For **Schema**, choose **public**.
3. Enter the following in the query editor window, and choose **Run** to create a new table.

```
create table shoes(
                shoetype varchar (10),
                color varchar(10));
```

6. Choose **Clear**.
7. Enter the following command in the query editor window, and choose **Run** to add rows to the table.

```
insert into shoes values
('loafers', 'brown'),
('sandals', 'black');
```

10. Choose **Clear**.
11. Enter the following command in the query editor window, and choose **Run** to query the new table.

```
select * from shoes;
```

The **Query results** displays the results.

Shoe type	Color
sandals	black
loafers	brown

12. Choose **Execution** to view the run details.
13. Choose **Export** to download the query results as a file. The supported file formats are CSV, TXT, and HTML.

Step 6 Load Sample Data

Create the below tables in Redshift Editors

```
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 venue(
      venueid smallint not null distkey sortkey,
      venuename varchar(100),
      venuecity varchar(30),
      venuestate char(2),
      venueseats integer);
```

```
create table category(
      catid smallint not null distkey sortkey,
      catgroup varchar(10),
      catname varchar(10),
      catdesc varchar(50));
```

```
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'));

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 listing(
        listid integer not null distkey,
        sellerid integer not null,
        eventid integer not null,
        dateid smallint not null  sortkey,
        numtickets smallint not null,
        priceperticket decimal(8,2),
        totalprice decimal(8,2),
        listtime 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);
```

Create S3 folder and upload the tickit folder from local laptop to S3 folder

To copy from S3 to Redshift perform the below commands

```
copy users from 's3://<myBucket>/tickit/allusers_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' region '<aws-region>';


copy venue from 's3://<myBucket>/tickit/venue_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' region '<aws-region>';


copy category from 's3://<myBucket>/tickit/category_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' region '<aws-region>';


copy date from 's3://<myBucket>/tickit/date2008_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' region '<aws-region>';


copy event from 's3://<myBucket>/tickit/allevents_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' timeformat 'YYYY-MM-DD HH:MI:SS' region '<aws-region>';


copy listing from 's3://<myBucket>/tickit/listings_pipe.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '|' region '<aws-region>';


copy sales from 's3://<myBucket>/tickit/sales_tab.txt'
credentials 'aws_iam_role=<iam-role-arn>'
delimiter '\t' timeformat 'MM/DD/YYYY HH:MI:SS' region '<aws-region>';
```

Step 7: Try example queries

```
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;
```

Delete Cluster

1. On the navigation menu, choose **CLUSTERS** to display your list of clusters.
2. Choose the **redshift-cluster-1** cluster. For **Actions**, choose **Delete**. The **Delete cluster** page appears.
3. Confirm the cluster to be deleted, then choose **Delete cluster**.

Reference: https://docs.aws.amazon.com/redshift/latest/gsg/getting-started.html