Building Data Analytics Solutions Using Amazon Redshift

Lab 1 - Load and Query Data in an Amazon Redshift Cluster

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at *AWS Training and Certification*.

Lab overview

Your organization is looking to unlock business value from a wealth of data across different organizational units. It decides to build a data warehouse solution that can perform analytics on the data. Your task is to come up with a fully managed and cost-effective solution at a petabyte scale. You want to explore Amazon Redshift to meet these criteria. In this lab, you experience how to set up a data warehouse in a short time using Amazon Redshift.

OBJECTIVES

By the end of this lab, you will be able to:

- Create an Amazon Redshift cluster.
- Load data into the cluster.
- Use psql to query data in the cluster using the Command Host instance.

TECHNICAL KNOWLEDGE PREREQUISITES

- Experience with Cloud platforms.
- Basic navigation of the AWS Management Console.
- Basic knowledge of Amazon Redshift.

DURATION

This lab requires approximately *40* minutes to complete.

ICON KEY

Various icons are used throughout this lab to call attention to certain aspects of the guide. The following list explains the purpose for each one:

- **Command:** A command that you must run.
- **Expected output:** A sample output that you can use to verify the output of a command or edited file.
- **Note:** A hint, tip, or important guidance.
- **Learn more:** Where to find more information.
- **Caution:** Information of special interest or importance (not so important to cause problems with the equipment or data if you miss it, but it could result in the need to repeat certain steps).
- **Copy edit:** A time when copying a command, script, or other text to a text editor (to edit specific variables within it) might be easier than editing directly in the command line or terminal.
- **Answer:** An answer to a question or challenge.
- **Hint:** A hint to a question or challenge.

Start lab

1. To launch the lab, at the top of the page, choose Start lab.

Caution: You must wait for the provisioned AWS services to be ready before you can continue.

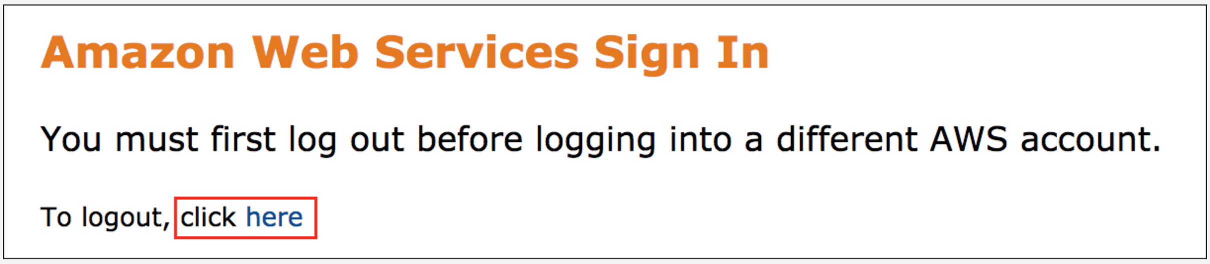
2. To open the lab, choose Open Console.

You are automatically signed in to the AWS Management Console in a new web browser tab.

WARNING: Do not change the Region unless instructed.

COMMON SIGN-IN ERRORS

Error: You must first sign out



Amazon Web Services Sign In

You must first log out before logging into a different AWS account.

To logout, click here

If you see the message, **You must first log out before logging into a different AWS account:**

- Choose the **click here** link.
- Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
- Choose Open Console again.

Error: Choosing Start Lab has no effect

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

- Add the lab domain name to your pop-up or script blocker's allow list or turn it off.
- Refresh the page and try again.

AWS SERVICES NOT USED IN THIS LAB

AWS service capabilities used in this lab are limited to what the lab requires. Expect errors when accessing other services or performing actions beyond those provided in this lab guide.

Task 1: Explore the lab environment

In this task, you review the lab environment to gain a better understanding of the resources you work with through this lab.

LAB ARCHITECTURE

First, examine the lab architecture.

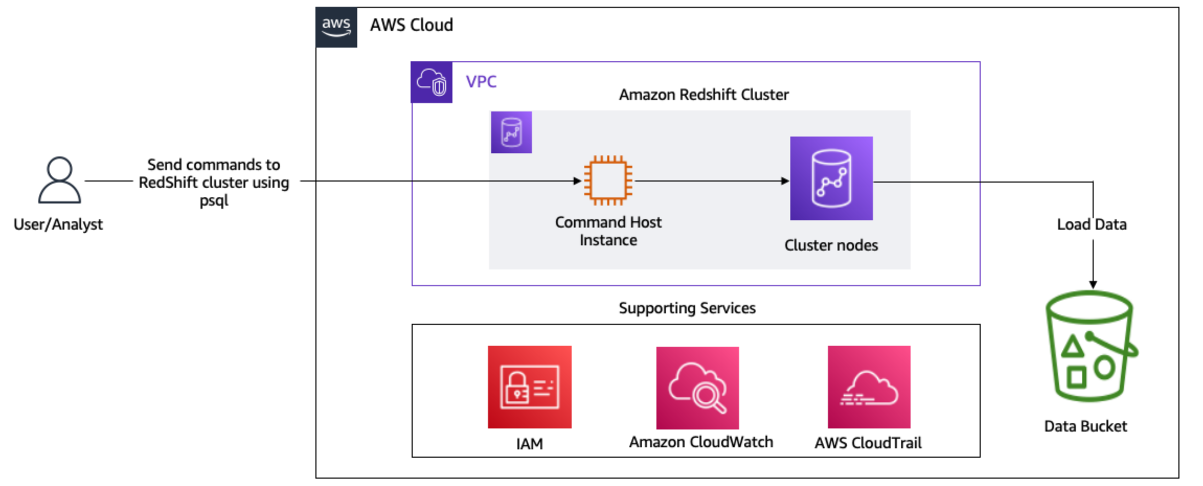


Image description: The preceding diagram depicts the connection between an external user and Amazon Redshift cluster nodes via a CommandHost EC2 instance within an Amazon Virtual Private Cloud. It also depicts other supporting services like IAM, Amazon CloudWatch and AWS CloudTrail.

During the lab deployment process, the following resources are created for you:

- An Amazon Virtual Private Cloud (Amazon VPC)
- An Amazon Simple Storage Service (Amazon S3) bucket that contains data that you load into Amazon Redshift
- An Amazon Elastic Compute Cloud (Amazon EC2) environment for you to use during the lab. Connect to the command host through the **CommandHostSessionUrl** value found in the left pane of these instructions.

Note: During this lab, you need to send commands to a Redshift cluster database. To do this, you connect to the **Command Host** to use **psql**.

REVIEW THE FOLDERS IN THE AMAZON S3 BUCKET

3. If you have not already done so, follow the steps in the Start Lab section to log into the AWS Management Console.
4. At the top-right corner of the page, verify that the **AWS Region** matches the **Region** listed to the left of these instructions.

Choose the **Region** drop-down menu to display the list of AWS Regions and their associated codes. For example, **US West (Oregon)** has a code of **us-west-2**.

5. At the top of the page, in the unified search bar, search for and choose

   S3 .
6. Choose the link for the bucket with **databucket** in the name.
7. Select the **data/** link to open the folder.

There is one CSV file in the folder named **stock_prices.csv**, which contains actual trading data from January 2, 2001 through September 14, 2021.

The file contains data similar to this:

trade_date	ticker	high	low	open_value	close	volume	adj_close
2021-09-09	aal	20.61	19.01	19.10	20.20	60077200	20.20
2021-09-09	aapl	156.11	153.95	155.49	154.07	57305700	154.07
2021-09-09	amzn	3549.99	3480.37	3526.02	3484.16	2719200	3484.16
2021-09-09	ba	216.61	210.72	211.15	213.94	9242800	213.94
2021-09-09	bac	41.35	40.56	40.66	40.93	36266400	40.93
2021-09-09	c	71.10	69.91	69.97	70.46	14212900	70.46
2021-09-09	chwy	76.89	75.54	75.99	76.60	2386900	76.60
2021-09-09	coke	398.45	393.25	397.75	394.14	16200	394.14
2021-09-09	dis	187.58	184.57	185.15	185.91	7190700	185.91
2021-09-09	f	12.95	12.72	12.95	12.76	68806400	12.76

Congratulations! You have successfully reviewed the lab environment. Now you’re ready to begin with the actual tasks!

Task 2: Set up a subnet group and parameter group

Typically, a network administrator sets up network-related activities, such as a virtual private cloud (VPC), security groups, subnets, subnet groups, and parameter groups. Typically, also, this setup is a one-time activity. In this task, you create the security and parameter groups yourself, to help you become familiar with some of these concepts. This setup is required to create the Redshift cluster in this task.

CREATE A CLUSTER SUBNET GROUP

Your VPC can have multiple subnets, which group resources based on security and operational needs. A *cluster subnet group* is used to specify a set of subnets in your VPC. When provisioning a Redshift cluster, you select the subnet group. Amazon Redshift creates the cluster in one of the subnets listed in the cluster subnet group.

First, create a cluster subnet group.

8. At the top of the page, in the unified search bar, search for and choose

   ```
   Amazon Redshift
   ```
   .
9. In the navigation pane on the left, expand **Configurations**.

If **Configurations** is not visible, choose the navigation icon and select **Configurations** from the list.

10. Choose **Subnet groups**.
11. Choose **Create cluster subnet group** .
12. On the **Create cluster subnet group** page:

- For **Name**, enter

  ```
  lab-subnet-group
  ```

- For **Description**, enter

  ```
  Lab cluster subnet group
  ```

- For **VPC**, select **Lab VPC**.
- For **Availability Zone**, select the first Availability Zone from the list.
- For **Subnet**, select the first subnet from the list.
- Choose **Add subnet**

Notice that the Availability Zone and Subnet ID of the subnet you selected are now displayed in the **Subnets in this cluster subnet group** section.

13. Choose **Create cluster subnet group** .

Expected output:

A green banner Cluster subnet group lab-subnet-group was create successfully should appear at the top of the page.

The cluster subnet group tells Amazon Redshift which subnet(s) in which Availability Zone(s) can be used when launching the cluster.

CREATE A CLUSTER PARAMETER GROUP

In Amazon Redshift, you associate a *parameter group* with each cluster that you create. The parameter group is a group of settings that apply to all of the databases that you create in the cluster. The parameter group includes settings such as query timeout and date style.

A default parameter group is available, but if you wish to use non-default values, you can create a custom parameter group, which you do in this task.

14. In the left navigation pane, choose **Workload management**.
15. Choose **Create parameter group** or **Create** .
16. In the **Create parameter group** pop-up window:

- For **Parameter group name**, enter

```
redshift-lab-parameters
```

- For **Description**, enter

```
Lab cluster parameter group
```

17. Choose **Create** .

Expected output:

A green banner Parameter group redshift-lab-parameters was created should appear at the top of the page.

For more information about parameter groups, refer to *Amazon Redshift Parameter Groups* in the **Additional Resources** section at the end of this lab.

Next, you want to change the **statement_timeout** parameter of the parameter group you just created. The statement_timeout value is the maximum amount of time a query can run before Amazon Redshift stops it. In this lab, you set it to stop all queries that run for more than 1 minute.

18. On the **Workload management** page, in the **Parameter groups** section, select **redshift-lab-parameters**.

If the parameter group is not listed, you might need to refresh the page.

19. To the right of the **Parameter groups** section, under the **redshift-lab-parameters** heading, choose the **Parameters** tab.
20. On the **Parameters** tab, choose **Edit parameters**
21. On the **Modify parameters: redshift-lab-parameters** page:

- For **statement_timeout**, enter

```
60000
```

  ○ The statement timeout value is measured in milliseconds. 60000 milliseconds is equal to 1 minute.

- Keep the remaining default values.

22. At the bottom of the page, choose **Save** .

Expected output:

A green banner redshift-lab-parameters successfully modified should appear at the top of the page.

Congratulations! You have successfully created Amazon Redshift subnet and parameter groups to use when you create a Redshift cluster.

Task 3: Create an Amazon Redshift cluster

In this task, you create a new Redshift cluster under the subnet group and parameter group you created in the previous task.

23. In the navigation pane at the left of the page, choose **Clusters**.
24. On the **Clusters** page, in the **Clusters** section, choose **Create cluster** .

The cluster configuration page sets the cluster identifiers and login details for the *Admin* user.

25. On the **Create cluster** page, in the **Cluster configuration** section:

- For **Cluster identifier**, enter

lab-cluster

- For **Choose the size of the cluster**, select **I'll choose**.
- For **Node type**, select **dc2.large**.
- For **Nodes**, enter

1

You set the node type and number of nodes to these values because you only need a minimal environment to test in for this lab. In a production environment you want to ensure you size the cluster to best meet your needs. For more information about clusters, refer to *Amazon Redshift Clusters* in the **Additional Resources** section at the end of this lab.

26. In the **Database configurations** section:

- For **Admin user name**, enter

dbadmin

- For **Admin user password**, enter

Redshift123

27. In the **Cluster permissions** section, select **Associate IAM role** .
28. Choose **RedshiftAccessRole**.
29. Choose **Associate IAM roles** .
30. To the right of the **Additional configurations** heading, turn off **Use defaults**.

In this section, instead of using the default VPC settings, you have the option to choose a different VPC and subnets. You might want to use a specific VPC for many reasons. For example, your organization might want to keep all Redshift clusters in a separate VPC CIDR range. (Classless Inter-Domain Routing is a method for allocating IP addresses and for IP routing.)

31. Expand the **Network and security** section.
32. In the **Network and security** section:

- For **Virtual private cloud (VPC)**, select **Lab VPC**.
- For **VPC security groups**, deselect **default**, and then select **Redshift Security Group**.
  - Verify there is only one blue box that says **Redshift Security Group** below the **VPC security groups** field. If the **default** box is still there, choose the X to remove it.
- For **Cluster subnet group**, select **lab-subnet-group**.
- For **Availability Zone**, select the first Availability Zone from the list.

33. Expand the **Database configurations** section.
34. In the **Database configurations** section:

- For **Database name**, keep the

  `dev` default value.

- For **Database port**, enter

  `5439`

- For **Parameter groups**, select **redshift-lab-parameters**.

35. At the bottom of the page, choose **Create cluster** .

Expected output:

A blue banner Amazon Redshift is creating lab-cluster should appear at the top of the **Clusters** page.

In the **Clusters** section, notice the **Status** of **lab-cluster** is **Creating**. When the cluster creation is complete, the status changes to Available.

The cluster can take approximately 5 minutes to launch.

CHALLENGE TASK

While the cluster is creating, choose the **lab-cluster** link to open the cluster's **General information** page. Your task is to investigate the content of the page to discover the value of the following three items:

- Node public IP address
- Next maintenance schedule
- Associated IAM roles

If you encounter an error message stating the **lab-cluster** cluster is was not found, wait 2-3 minutes and the refresh the page.

Congratulations! You have successfully created and configured a new Amazon Redshift cluster!

Task 4: Load data to the Amazon Redshift cluster

In this task, you use the **psql** interface to create a table in the Redshift cluster. You then use a **COPY** operation to import the data from your S3 bucket to your table.

36. Choose the link for new cluster you just created.

On the **lab-cluster** page, in the **General information** section, you find the cluster endpoint. The endpoint should be similar to: **lab-cluster.cvrd3r46okph.us-west-2.redshift.amazonaws.com:5439/dev**

37. **Copy edit:** Copy the endpoint value and paste it on a notepad. Remove the **:5439/dev** portion from the end and save the remaining endpoint URL for use in the next task.

The final endpoint should be similar to: **lab-cluster.cvrd3r46okph.us-west-2.redshift.amazonaws.com**

DIRECTIONS FOR CONNECTING TO THE COMMAND HOST TO USE PSQL

38. **Copy edit:** Copy the **CommandHostSessionUrl** value found in the left pane of these instructions and paste it into a new browser tab to access the command host terminal.
39. **Command:** Run the following commands on the command host:

Note: Replace the string **<INSERT_REDSHIFT_CLUSTER_ENDPOINT>** with the value you recorded in the previous step. Make sure that you have removed the **:5439/dev** portion from the end before running the command.

```
cd ~
export PGPASSWORD='Redshift123'
psql -U dbadmin -h '<INSERT_REDSHIFT_CLUSTER_ENDPOINT>' -d dev -p 5439
```

Expected output: Your values differ from what is seen below.

```
******************************
**** This is OUTPUT ONLY. ****
******************************
```

```
sh-4.2$ cd ~
sh-4.2$ export PGPASSWORD='Redshift123'
sh-4.2$ psql -U dbadmin -h lab-cluster.cvrd3r46okph.us-west-
2.redshift.amazonaws.com -d dev -p 5439
psql (13.7, server 8.0.2)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits:
256, compression: off)
Type "help" for help.

dev=#
```

This should log you into the database and give you a prompt where you can enter **SQL** commands used in this lab.

CREATE A TABLE

40. Using the psql prompt, enter the following query to create a new table named **stocksummary**:

```
CREATE TABLE IF NOT EXISTS stocksummary (
        Trade_Date VARCHAR(15),
        Ticker VARCHAR(5),
        High DECIMAL(8,2),
        Low DECIMAL(8,2),
        Open_value DECIMAL(8,2),
        Close DECIMAL(8,2),
        Volume DECIMAL(15),
        Adj_Close DECIMAL(8,2)
        );
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

CREATE TABLE
dev=#
```

SHOW TABLE

41. Using the psql prompt, enter the following query to show tables in the database:

```
\dt
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

            List of relations
 schema |     name     | type  |  owner
--------+--------------+-------+---------
 public | stocksummary | table | dbadmin
(1 row)

dev-#
```

IMPORT DATA FROM AMAZON S3 TO THE AMAZON REDSHIFT TABLE

Next, use the **COPY** command to load stock data into the *stocksummary* table.

The COPY command uses the Amazon Redshift massively parallel processing (MPP) architecture to read and load data in parallel from files in Amazon S3, from an Amazon DynamoDB table, or from text output from one or more remote hosts.

The COPY command appends the new input data to any existing rows in the table. The maximum size of a single input row from any source is 4 MB.

42. Using the psql prompt, enter the following query to load data into the **stocksummary** table from the Amazon S3 bucket:

- Replace the **INSERT_DATA_BUCKET_NAME** placeholder value with the **dataBucket** value listed to the left of these instructions.

- Replace the **INSERT_REDSHIFT_ROLE** placeholder value with the **RedshiftAccessRole** value listed to the left of these instructions. (Be sure to keep the single quote marks.)

```
COPY stocksummary
FROM 's3://INSERT_DATA_BUCKET_NAME/data/stock_prices.csv'
iam_role 'INSERT_REDSHIFT_ROLE'
CSV IGNOREHEADER 1;
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

INFO:  Load into table 'stocksummary' completed, 108230 record(s) loaded
successfully.
COPY
dev=#
```

VALIDATE THE DATA LOADING

43. Using the psql prompt, enter the following query to query the **stocksummary** table for the stocks that were traded on January 3, 2020:

```
SELECT * FROM stocksummary WHERE Trade_Date LIKE '2020-01-03' ORDER BY Ticker;
```

Expected output:

The query results should display the details for stocks that were traded on January 3, 2020, similar to this:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

 trade_date | ticker |  high   |   low   | open_value |  close  |  volume   |
adj_close
------------+--------+---------+---------+------------+---------+-----------+-
----------
 2020-01-03 | aal    |   28.29 |   27.34 |      28.27 |   27.64 |  14008900 |
27.54
 2020-01-03 | aapl   |   75.14 |   74.12 |      74.28 |   74.35 | 146322800 |
73.37
 2020-01-03 | amzn   | 1886.19 | 1864.50 |    1864.50 | 1874.96 |   3764400 |
1874.96
 2020-01-03 | ba     |  334.89 |  330.29 |     330.63 |  332.76 |   3875900 |
330.79
 2020-01-03 | bac    |   35.15 |   34.75 |      34.97 |   34.90 |  50357900 |
33.50
 2020-01-03 | c      |   80.51 |   79.44 |      79.80 |   79.69 |  12437400 |
74.87
 2020-01-03 | chwy   |   29.39 |   28.53 |      29.00 |   29.34 |   2205300 |
29.34
```

```
 2020-01-03 | coke     |  287.35 |  277.48 |    279.76 |  285.76 |     37500 |
283.90
 2020-01-03 | dis      |  147.89 |  146.05 |    146.39 |  146.50 |   7320200 |
146.50
 2020-01-03 | f        |    9.36 |    9.14 |      9.31 |    9.21 |  45040800 |
9.05
 2020-01-03 | ge       |   96.00 |   92.23 |     92.55 |   95.76 |  10735725 |
95.13
 2020-01-03 | gs       |  232.61 |  230.30 |    231.60 |  231.58 |   2274500 |
223.40
 2020-01-03 | hsy      |  145.88 |  143.75 |    143.97 |  145.25 |    770900 |
140.04
 2020-01-03 | intc     |   60.70 |   59.81 |     59.81 |   60.09 |  15293900 |
57.55
 2020-01-03 | kodk     |    4.19 |    3.92 |      4.00 |    4.03 |    242900 |
4.03
 2020-01-03 | m        |   16.61 |   16.20 |     16.31 |   16.53 |  12026100 |
15.87
 2020-01-03 | ma       |  302.42 |  298.60 |    299.45 |  300.42 |   2501300 |
297.73
 2020-01-03 | msft     |  159.94 |  158.05 |    158.32 |  158.61 |  21116200 |
155.93
 2020-01-03 | nke      |  102.00 |  100.30 |    100.58 |  101.91 |   4541800 |
100.38
 2020-01-03 | pg       |  123.52 |  121.86 |    122.16 |  122.58 |   7970500 |
117.41
 2020-01-03 | pypl     |  110.41 |  108.76 |    109.48 |  108.76 |   7098300 |
108.76
 2020-01-03 | sq       |   63.27 |   62.33 |     62.59 |   63.00 |   5087100 |
63.00
 2020-01-03 | tsla     |   90.80 |   87.38 |     88.09 |   88.60 |  88892500 |
88.60
 2020-01-03 | v        |  190.96 |  187.91 |    188.41 |  189.60 |   4899700 |
187.61
 2020-01-03 | wmt      |  118.79 |  117.58 |    118.26 |  117.88 |   5399200 |
114.59
(25 rows)

dev=#
```

44. Using the psql prompt, enter the following query to find the all time high stock price for each company:

```
select a.ticker, a.trade_date, '$'||a.adj_close as highest_stock_price
from stocksummary a,
  (select ticker, max(adj_close) adj_close
  from stocksummary x
  group by ticker) b
where a.ticker = b.ticker
  and a.adj_close = b.adj_close
order by a.ticker;
```

Expected output:

The query results should display the all time high stock price for each company, similar to this:

```
******************************
**** This is OUTPUT ONLY. ****
******************************
```

```
 ticker | trade_date | highest_stock_price
--------+------------+---------------------
 aal    | 2006-11-24 | $59.34
 aal    | 2006-11-22 | $59.34
 aapl   | 2021-09-07 | $156.69
 amzn   | 2021-07-08 | $3731.40
 ba     | 2019-03-01 | $430.29
 bac    | 2021-06-04 | $43.04
 c      | 2006-12-27 | $442.23
 chwy   | 2021-02-12 | $118.69
 coke   | 2021-06-08 | $450.68
 dis    | 2021-03-08 | $201.91
 f      | 2001-04-18 | $17.01
 ge     | 2016-07-19 | $232.21
 gs     | 2021-08-27 | $417.66
 hsy    | 2021-08-17 | $181.21
 intc   | 2021-04-09 | $67.40
 kodk   | 2014-01-08 | $37.20
 kodk   | 2014-01-09 | $37.20
 m      | 2015-07-16 | $54.98
 ma     | 2021-04-28 | $395.18
 msft   | 2021-08-23 | $304.64
 nke    | 2021-08-05 | $173.56
 pg     | 2021-09-13 | $145.67
 pypl   | 2021-07-23 | $308.52
 sq     | 2021-08-05 | $281.80
 tsla   | 2021-01-26 | $883.09
 v      | 2021-07-27 | $250.58
 wmt    | 2021-08-20 | $151.44
(27 rows)

dev=#
```

Congratulations! You have successfully created a table, loaded the data, and validated the data using psql queries. In the challenge task, repeat the process with a different dataset.

(OPTIONAL) CHALLENGE TASK - INGEST AND QUERY MOVIE DATA

The **challengeBucket** S3 bucket contains data related to various movies. Your task is to create a movies table and find the number of movies that Mark Wahlbe is associated with as an actor.

Hint: The column names in the movies data file are:

year,
title,

directors_0,

rating,

genres_,

genres_,

rank,

running_time_sec,

actors_0,

actors_1,

actors_2,

directors_1, and

directors_2

```
SELECT * FROM STL_LOAD_ERRORS
```

If you get stuck, refer to *Task 4 challenge solution* in the **Appendix** section.

Conclusion

Congratulations! You have successfully:

- Created an Amazon Redshift cluster.
- Loaded data into the cluster.
- Used psql to query data in the cluster using the Command Host instance.

End lab

Follow these steps to close the console and end your lab.

45. Return to the **AWS Management Console**.
46. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
47. Choose End lab and then confirm that you want to end your lab.

Additional Resources

- Amazon Redshift parameter groups
- Amazon Redshift clusters

Appendix

TASK 4 CHALLENGE SOLUTION

Code to create the table:

```
CREATE TABLE IF NOT EXISTS movies  (
        year VARCHAR(4) DEFAULT NULL,
        title VARCHAR(200) DEFAULT NULL,
        directors VARCHAR(35) DEFAULT NULL,
        rating VARCHAR(10) DEFAULT NULL,
        genres_0 VARCHAR(35) DEFAULT NULL,
        genres_1 VARCHAR(35) DEFAULT NULL,
        rank VARCHAR(10) DEFAULT NULL,
        running_time_secs VARCHAR(35) DEFAULT NULL,
        actors_0 VARCHAR(35) DEFAULT NULL,
        actors_1 VARCHAR(35) DEFAULT NULL,
        actors_2 VARCHAR(35) DEFAULT NULL,
        directors_1 VARCHAR(35) DEFAULT NULL,
        directors_2 VARCHAR(35) DEFAULT NULL
);
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

CREATE TABLE
```

Code to import the data:

- Replace the **INSERT_CHALLENGE_BUCKET_NAME** placeholder value with the **challengeBucket** value listed to the left of these instructions.
- Replace the **INSERT_REDSHIFT_ROLE** placeholder value with the **RedshiftAccessRole** value listed to the left of these instructions. (Be sure to keep the single quote marks.)

```
COPY movies
FROM 's3://INSERT_CHALLENGE_BUCKET_NAME/data/movies.csv'
iam_role 'INSERT_REDSHIFT_ROLE'
CSV IGNOREHEADER 1;
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
******************************

INFO:  Load into table 'movies' completed, 4609 record(s) loaded successfully.
COPY
```

Code to query the movies table for any movie with **Mark Wahlberg** as an actor:

```
SELECT title FROM movies WHERE actors_0='Mark Wahlberg' OR actors_1='Mark
Wahlberg' OR actors_2='Mark Wahlberg';
```

Expected output:

```
******************************
**** This is OUTPUT ONLY. ****
```

```
******************************

              title
---------------------------------
 Transformers: Age of Extinction
 Pain & Gain
 2 Guns
 Ted
 Lone Survivor
 The Lovely Bones
 Shooter
 Broken City
 The Other Guys
 The Fighter
 The Italian Job
 Contraband
 Invincible
 The Happening
 Date Night
 Planet of the Apes
 Four Brothers
 The Perfect Storm
 Three Kings
 Fear
 Max Payne
 Rock Star
 We Own the Night
 The Yards
(24 rows)

dev=#
```