

---

# Amazon Redshift

## Getting Started Guide



## **Amazon Redshift: Getting Started Guide**

Copyright © 2021 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

## Table of Contents

Getting started .....	1
Step 1: Set up prerequisites .....	1
Sign up for AWS .....	2
Determine firewall rules .....	2
Step 2: Create an IAM role .....	2
.....	2
Step 3: Create a cluster .....	3
.....	3
Step 4: Authorize cluster access .....	4
.....	4
Step 5: Grant access to the query editor and run queries .....	4
Grant access to the query editor .....	5
Use the query editor .....	6
Step 6: Load sample data .....	7
Step 7: Try example queries .....	10
Step 8: Reset your environment .....	11
Additional resources .....	11
Document history .....	13

# Getting started with Amazon Redshift

Welcome to the *Amazon Redshift Getting Started Guide*. Amazon Redshift is a fully managed, petabyte-scale data warehouse service in the cloud. An Amazon Redshift data warehouse is a collection of computing resources called *nodes*, which are organized into a group called a *cluster*. Each cluster runs an Amazon Redshift engine and contains one or more databases.

If you are a first-time user of Amazon Redshift, we recommend that you begin by reading the following sections:

- [Amazon Redshift management overview](#) – This topic provides an overview of Amazon Redshift.
- [Service highlights and pricing](#) – This product detail page provides details about Amazon Redshift service highlights and pricing.
- [Amazon Redshift Getting Started \(this guide\)](#) – This guide provides a tutorial of using Amazon Redshift to create a sample cluster and work with sample data.

This guide is a tutorial that walks you through the process of creating a sample Amazon Redshift cluster. You can use this sample cluster to evaluate the Amazon Redshift service. In this tutorial, you perform the following steps:

- [Step 1: Set up prerequisites \(p. 1\)](#)
- [Step 2: Create an IAM role \(p. 2\)](#)
- [Step 3: Create a sample Amazon Redshift cluster \(p. 3\)](#)
- [Step 4: Authorize access to the cluster \(p. 4\)](#)
- [Step 5: Grant access to the query editor and run queries \(p. 4\)](#)
- [Step 6: Load sample data from Amazon S3 \(p. 7\)](#)
- [Step 7: Try example queries \(p. 10\)](#)
- [Step 8: Reset your environment \(p. 11\)](#)

After you complete this tutorial, you can find more information about Amazon Redshift and next steps in [Additional resources \(p. 11\)](#).

## **Important**

The sample cluster that you create runs in a live environment. The on-demand rate is \$0.25 per hour for using the sample cluster that is designed in this tutorial until you delete it. For more pricing information, go to [the Amazon Redshift pricing page](#). If you have questions or get stuck, you can reach out to the Amazon Redshift team by posting on our [Discussion forum](#).

This tutorial isn't meant for production environments and doesn't discuss options in depth. After you complete the steps in this tutorial, you can use [Additional resources \(p. 11\)](#) to find more in-depth information. This information can help you plan, deploy, and maintain your clusters, and work with the data in your data warehouse.

## Step 1: Set up prerequisites

Before you begin setting up an Amazon Redshift cluster, make sure that you complete the following prerequisites:

- [Sign up for AWS \(p. 2\)](#)
- [Determine firewall rules \(p. 2\)](#)

## Sign 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.

Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

## Determine firewall rules

As part of this tutorial, you specify a port when you launch your Amazon Redshift cluster. You also create an inbound ingress rule in a security group to allow access through the port to your cluster.

If your client computer is behind a firewall, make sure that you know an open port that you can use. Using this open port, you can connect to the cluster from a SQL client tool and run queries. If you don't know this, work with someone who understands your network firewall rules to determine an open port in your firewall. Though Amazon Redshift uses port 5439 by default, the connection doesn't work if that port isn't open in your firewall. You can't change the port number for your Amazon Redshift cluster after it is created. Thus, make sure that you specify an open port that works in your environment during the launch process.

## Step 2: Create an IAM role

For any operation that accesses data on another AWS resource, your cluster needs permission to access the resource and the data on the resource on your behalf. An example is using a `COPY` command to load data from Amazon S3. You provide those permissions by using AWS Identity and Access Management (IAM). You can do this through an IAM role that is attached to your cluster. Or you can provide the AWS access key for an IAM user that has the necessary permissions. For more information about credentials and access permissions, see [Credentials and access permissions](#).

To best protect your sensitive data and safeguard your AWS access credentials, we recommend creating an IAM role and attaching it to your cluster. For more information about providing access permissions, see [Permissions to access other AWS resources](#).

In this step, you create a new IAM role that enables Amazon Redshift to load data from Amazon S3 buckets. An IAM role is an IAM identity that you can create in your account that has specific permissions. In the next step, you attach the role to your cluster.

### To create an IAM role for Amazon Redshift

1. Sign in to the AWS Management Console and open the IAM console at <https://console.aws.amazon.com/iam/>.
2. In the navigation pane, choose **Roles**.
3. Choose **Create role**.
4. In the **AWS Service** group, choose **Redshift**.

5. Under **Select your use case**, choose **Redshift - Customizable**, then choose **Next: Permissions**.
6. On the **Attach permissions policies** page, choose **AmazonS3ReadOnlyAccess**. You can leave the default setting for **Set permissions boundary**. Then choose **Next: Tags**.
7. The **Add tags** page appears. You can optionally add tags. Choose **Next: Review**.
8. For **Role name**, enter a name for your role. For this tutorial, enter **myRedshiftRole**.
9. Review the information, and then choose **Create Role**.
10. Choose the role name of the role that you just created.
11. Copy the **Role ARN** value to your clipboard—this value is the Amazon Resource Name (ARN) for the role that you just created. You use that value when you use the COPY command to load data in [Step 6: Load sample data from Amazon S3 \(p. 7\)](#).

Now that you have created the new role, your next step is to attach it to your cluster. You can attach the role when you launch a new cluster or you can attach it to an existing cluster. In the next step, you attach the role to a new cluster.

## Step 3: Create a sample Amazon Redshift cluster

Now that you have the prerequisites completed, you can launch your Amazon Redshift cluster.

The cluster that you are about to create is live (and not running in a sandbox). You incur the standard Amazon Redshift usage fees for the cluster until you delete it. If you complete the tutorial described here in one sitting and delete the cluster when you are finished, the total charges are minimal.

### To create an Amazon Redshift cluster

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

#### Important

If you use IAM user credentials, ensure that you have the necessary permissions to perform the cluster operations. For more information, see [Controlling access to IAM users](#) in the *Amazon Redshift Cluster Management Guide*.

2. At upper right, choose the AWS Region in which you want to create the cluster.
3. On the navigation menu, choose **CLUSTERS**, then choose **Create cluster**. The **Create cluster** page appears.
4. In the **Cluster configuration** section, specify values for **Cluster identifier**, **Node type**, and **Nodes**:
  - **Cluster identifier**: Enter **examplecluster** for this tutorial. This identifier must be unique. The identifier must be from 1–63 characters using as valid characters a–z (lowercase only) and - (hyphen).
  - Choose one of the following methods to size your cluster:

#### Note

The following step describes an Amazon Redshift console that is running in an AWS Region that supports RA3 node types. For a list of AWS Regions that support RA3 node types, see [Overview of RA3 node types](#) in the *Amazon Redshift Cluster Management Guide*.

- If your AWS Region supports RA3 node types, choose either **Production** or **Free trial** to answer the question **What are you planning to use this cluster for?**

If your organization is eligible, you might be able to create a cluster under the Amazon Redshift free trial program. To do this, choose **Free trial** to create a configuration with the dc2.large node type. For more information about choosing a free trial, see [Amazon Redshift free trial](#).

- If you don't know how large to size your cluster, choose **Help me choose**. Doing this starts a sizing calculator that asks you questions about the size and query characteristics of the data that you plan to store in your data warehouse.

If you know the required size of your cluster (that is, the node type and number of nodes), choose **I'll choose**. Then choose the **Node type** and number of **Nodes** to size your cluster for the proof of concept.

- Choose **Node type: dc2.1large** with **Nodes: 2** for this tutorial.
5. In the **Database configurations** section, specify values for **Database name (optional)**, **Database port (optional)**, **Master user name**, and **Master user password**.

This tutorial uses these values:

- **Database name (optional)**: Enter `dev`.
  - **Database port (optional)**: Enter `5439`.
  - **Master user name**: Enter `awsuser`.
  - **Master user password**: Enter a value for the password.
6. (Optional) In the **Cluster permissions** section, for **Available IAM roles** choose the IAM role that you previously created, `myRedshiftRole`. Then choose **Associate IAM role**.
  7. (Optional) In the **Additional configurations** section, turn off **Use defaults** to modify **Network and security**, **Database configurations**, **Maintenance**, **Monitoring**, and **Backup** settings.
  8. Choose **Create cluster**.

## Step 4: Authorize access to the cluster

Later in this tutorial, you access your cluster from within a virtual private cloud (VPC) based on the Amazon VPC service. However, if you use an SQL client from outside your firewall to access the cluster, make sure that you grant inbound access.

You can skip this step if you plan to access the cluster with the Amazon Redshift query editor from within your VPC.

### To check your firewall and grant inbound access to your cluster

1. Check your firewall rules if your cluster needs to be accessed from outside a firewall. For example, your client might be an Amazon EC2 instance or an external computer.
2. To access from an Amazon EC2 external client, add an ingress rule to the security group attached to your cluster that allows inbound traffic. You add Amazon EC2 security group rules in the Amazon EC2 console. For example, a CIDR/IP of `192.0.2.0/24` allows clients in that IP address range to connect to your cluster. Find out the correct CIDR/IP for your environment.

## Step 5: Grant access to the query editor and run queries

To query databases hosted by your Amazon Redshift cluster, you have two options:

- Connect to your cluster and run queries on the AWS Management Console with the query editor.

If you use the query editor, you don't have to download and set up an SQL client application.

- Connect to your cluster through an SQL client tool, such as SQL Workbench/J. For more information about using SQL Workbench/J, see [Connect to your cluster by using SQL Workbench/J](#) in the *Amazon Redshift Cluster Management Guide*.

Using the Amazon Redshift query editor is the easiest way to run queries on databases hosted by your Amazon Redshift cluster. After creating your cluster, you can immediately run queries using the Amazon Redshift console.

The following cluster node types support the query editor:

- DC1.8xlarge
- DC2.large
- DC2.8xlarge
- DS2.8xlarge
- RA3.xlplus
- RA3.4xlarge
- RA3.16xlarge

Using the Amazon Redshift console query editor, you can do the following:

- Run single SQL statement queries.
- Download result sets as large as 100 MB to a comma-separated value (CSV) file.
- Save queries for reuse. You can't save queries in the Europe (Paris) Region or the Asia Pacific (Osaka-Local) Region.
- View details of how queries run for user-defined tables.

For details about considerations when using the Amazon Redshift query editor, see [Querying a database using the query editor](#) in the *Amazon Redshift Cluster Management Guide*.

### Topics

- [Grant access to the query editor \(p. 5\)](#)
- [Use the query editor \(p. 6\)](#)

## Grant access to the query editor

To access the Amazon Redshift query editor, you need permission. To enable access, attach the `AmazonRedshiftQueryEditor` and `AmazonRedshiftReadOnlyAccess` IAM policies to the IAM user that you use to access your cluster.

If you have already created an IAM user to access Amazon Redshift, you can attach the `AmazonRedshiftQueryEditor` and `AmazonRedshiftReadOnlyAccess` policies to that user. If you haven't created an IAM user yet, create one and attach the policies to the IAM user.

### To attach the required IAM policies for the query editor

1. Sign in to the AWS Management Console and open the IAM console at <https://console.aws.amazon.com/iam/>.
2. Choose **Users**.
3. Choose the user that needs access to the query editor.
4. Choose **Add permissions**.



5. Choose **Attach existing policies directly**.
6. For **Policy names**, choose **AmazonRedshiftQueryEditor** and **AmazonRedshiftReadOnlyAccess**.
7. Choose **Next: Review**.
8. Choose **Add permissions**.

## Use the query editor

In the following example, you use the query editor to perform the following tasks:

- Run SQL commands.
- View details about how queries run.
- Save a query.
- Download a query result set.

### To use the query editor

1. Sign in to the AWS Management Console and open the Amazon Redshift console at <https://console.aws.amazon.com/redshift/>.
2. On the navigation menu, choose **EDITOR**, then connect to a database in your cluster.

On the **Connect to database** window, enter the values that you used when you created the cluster, as follows:

- **Cluster:** Choose **examplecluster**.
- **Database name:** Enter **dev**.
- **Database user:** Enter **awsuser**.
- **Database password:** Enter **password that you specified when you created the cluster**.

Then choose **Connect to database**.

3. For **Schema**, choose **public** to create a new table based on that schema.
4. 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));
```

5. Choose **Clear**.
6. 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');
```

7. Choose **Clear**.
8. 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

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

## Step 6: Load sample data from Amazon S3

At this point, you have a database called `dev` and you are connected to it. Next, you 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.

### Note

If you're using a SQL client tool, ensure that your SQL client is connected to the cluster.

After you complete this step, you can do the following:

- Try example queries at [Step 7: Try example queries \(p. 10\)](#).
- Reset your environment at [Step 8: Reset your environment \(p. 11\)](#).
- Find more information about Amazon Redshift at [Additional resources \(p. 11\)](#).

### To load sample data from Amazon S3

1. Create tables.

If you are using the Amazon Redshift query editor, individually 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 venue(  
  venueid smallint not null distkey sortkey,
```

```
venueid integer not null distkey,  
venuevenue varchar(100),  
venuecity varchar(30),  
venuestate char(2),  
venuevenue 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);
```

2. Load sample data from Amazon S3 by using the COPY command.

**Note**

We recommend using the COPY command to load large datasets into Amazon Redshift from Amazon S3 or DynamoDB. For more information about COPY syntax, see [COPY](#) in the *Amazon Redshift Database Developer Guide*.

- a. Download the file [ticketdb.zip](#), which contains individual sample data files.

- b. Unzip and load the individual files to a `ticket` folder in your Amazon S3 bucket in your AWS Region.
- c. Edit the COPY commands in this tutorial to point to the files in your Amazon S3 bucket. For information about how to manage files with Amazon S3, see [Creating and configuring an S3 Bucket](#) in the *Amazon Simple Storage Service Console User Guide*.
- d. Provide authentication for your cluster to access Amazon S3 on your behalf to load the sample data. You can provide either role-based authentication or key-based authentication. We recommend using role-based authentication. You can use the `myRedshiftRole` IAM role that you created at [Step 2: Create an IAM role \(p. 2\)](#) to enable Amazon Redshift to load data from Amazon S3 buckets. For more information about both types of authentication, see [CREDENTIALS](#) in the *Amazon Redshift Database Developer Guide*.

For this step, you provide authentication by referencing the IAM role that you created and then attached to your cluster in previous steps.

#### Note

If you don't have proper permissions to access Amazon S3, you receive the following error message when running the COPY command: `S3ServiceException: Access Denied`. For information about IAM permissions for the COPY command, see [COPY](#) in the *Amazon Redshift Database Developer Guide*.

The COPY commands include a placeholder for the Amazon Resource Name (ARN) for the IAM role, your bucket name, and an AWS Region, as shown in the following example.

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

To authorize access using an IAM role, replace `<iam-role-arn>` in the CREDENTIALS parameter string with the role ARN for the IAM role that you created in [Step 2: Create an IAM role \(p. 2\)](#).

Your COPY command looks similar to the following example.

```
copy users from 's3://<myBucket>/ticket/allusers_pipe.txt'  
credentials 'aws_iam_role=arn:aws:iam::123456789012:role/myRedshiftRole'  
delimiter '|' region '<aws-region>;
```

To load the sample data, replace `<myBucket>`, `<iam-role-arn>`, and `<aws-region>` in the following COPY commands with your values. If you are using the Amazon Redshift query editor, individually run the following commands.

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

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

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

```
copy date from 's3://<myBucket>/ticket/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

Now, try some example queries, as shown following. For more information on working with the SELECT command, see [SELECT](#) in the *Amazon Redshift Developer Guide*.

```
-- 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;
```

## Step 8: Reset your environment

When you have completed this tutorial, we suggest that you reset your environment to the previous state by deleting your sample cluster. You continue to incur charges for the Amazon Redshift service until you delete the cluster.

However, you might want to keep the sample cluster running if you intend to try tasks in other Amazon Redshift guides.

### To delete a cluster

1. Sign in to the AWS Management Console and open the Amazon Redshift console at <https://console.aws.amazon.com/redshift/>.
2. On the navigation menu, choose **CLUSTERS** to display your list of clusters.
3. Choose the **examplecluster** cluster. For **Actions**, choose **Delete**. The **Delete cluster** page appears.
4. Confirm the cluster to be deleted, then choose **Delete cluster**.

On the cluster list page, the cluster status is updated as the cluster is deleted.

## Additional resources

When you have completed this tutorial, we recommend that you continue to learn more about the concepts introduced in this guide by using the following Amazon Redshift resources:

- [Amazon Redshift Cluster Management Guide](#): This guide builds upon this *Amazon Redshift Getting Started*. It provides in-depth information about the concepts and tasks for creating, managing, and monitoring clusters.
- [Amazon Redshift management overview](#): This topic provides an overview of Amazon Redshift clusters.
- [Managing clusters using the console](#): This topic shows how to create, modify, resize, delete, reboot, and back up clusters using the Amazon Redshift console.
- [Querying a database](#): This topic shows how to query databases hosted by your Amazon Redshift cluster with the Amazon Redshift query editor or a SQL client tool.
- [Monitoring Amazon Redshift cluster performance](#): This topic shows how to monitor, isolate, and optimize your queries using the query monitoring features on the Amazon Redshift console.
- [Amazon Redshift Database Developer Guide](#): This guide also builds upon this *Amazon Redshift Getting Started*. It provides in-depth information for database developers about designing, building, querying, and maintaining the databases that make up your data warehouse.
- [Amazon Redshift system and architecture overview](#): This topic provides an overview of Amazon Redshift system and architecture.
- [Getting started using databases](#): This topic describes the basic steps to begin using the Amazon Redshift database.
- [SQL reference](#): This topic describes SQL commands and function references for Amazon Redshift.
- [System tables and views](#): This topic describes system tables and views for Amazon Redshift.
- [Tutorials for Amazon Redshift](#): This topic shows tutorials to learn about Amazon Redshift features.
- [Tuning table design](#): This tutorial describes how to optimize the design of your tables.
- [Loading data from Amazon Simple Storage Service](#): This tutorial describes how to load data into your Amazon Redshift database tables from data files in an Amazon Simple Storage Service bucket.
- [Querying nested data with Amazon Redshift Spectrum](#): This tutorial describes how to use Redshift Spectrum to query nested data in Parquet, ORC, JSON, and Ion file formats using external tables.

- [Configuring manual workload management \(WLM\) queues](#): This tutorial describes how to configure manual workload management (WLM) in Amazon Redshift.
- Feature videos:
  - Watch the following video to learn how to federate Amazon Redshift access with Microsoft Azure AD single sign-on: [Federating Amazon Redshift access with Microsoft Azure AD single sign-on](#).
  - Watch the following video to learn how to create and monitor usage limits using the Amazon Redshift console: [Cost Controls for Amazon Redshift Spectrum and Concurrency Scaling](#).
  - Watch the following video to learn how to monitor, isolate, and optimize your queries using the query monitoring features on the Amazon Redshift console: [Query Monitoring with Amazon Redshift](#).

# Document history

The following table describes the important changes since the last release of the *Amazon Redshift Getting Started Guide*.

**Latest documentation update: November 11, 2019**

Change	Description	Release date
Documentation update	Updated the guide to remove the original Amazon Redshift console and improve step flow.	August 14, 2020
New console	Updated the guide to describe the new Amazon Redshift console.	November 11, 2019
New feature	Updated the guide to describe the quick-launch cluster procedure.	August 10, 2018
New feature	Updated the guide to launch clusters from the Amazon Redshift dashboard.	July 28, 2015
New feature	Updated the guide to use new node type names.	June 9, 2015
Documentation update	Updated screenshots and procedure for configuring VPC security groups.	April 30, 2015
Documentation update	Updated screenshots and procedures to match the current console.	November 12, 2014
Documentation update	Moved loading data from Amazon S3 information into its own section and moved next steps section into the final step for better discoverability.	May 13, 2014
Documentation update	Removed the Welcome page and incorporated the content into the main Getting Started page.	March 14, 2014
Documentation update	This is a new release of the <i>Amazon Redshift Getting Started Guide</i> that addresses customer feedback and service updates.	March 14, 2014
New guide	This is the first release of the <i>Amazon Redshift Getting Started Guide</i> .	February 14, 2013