

Hands-on tutorials

Create and Connect to a Microsoft SQL Server Database with Amazon RDS



Create and Connect to a Microsoft SQL Server Database with Amazon RDS: Hands-on tutorials


Copyright © 2026 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

Create and Connect to a Microsoft SQL Server Database with Amazon RDS	i
Overview	1
What you will accomplish	1
Prerequisites	2
Implementation	2
Conclusion	21

Create and Connect to a Microsoft SQL Server Database with Amazon RDS

AWS experience	Beginner
Time to complete	25 minutes
Cost to complete	\$0.005 per hour* *You will only incur charges if you select In-use Public IPv4 Address.
Requires	<ul style="list-style-type: none">• AWS account <div data-bbox="862 840 1507 1150" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"><p> Note Accounts created within the past 24 hours might not yet have access to the services required for this tutorial.</p></div> <ul style="list-style-type: none">• Recommended browser: The latest version of Chrome or Firefox
Last updated	November 17, 2022

Overview

In this tutorial, you will learn how to create a Microsoft SQL Server database instance, connect to the database, and delete the DB instance. We will do this using [Amazon Relational Database Service \(Amazon RDS\)](#).

What you will accomplish

In this tutorial, you will:

- Create a Microsoft SQL Server database instance
- Connect to the database
- Delete the database instance

Prerequisites

Before starting this tutorial, you will need:

- **An AWS account:** If you don't already have an account, follow the [Setting Up Your AWS Environment](#) tutorial for a quick overview.

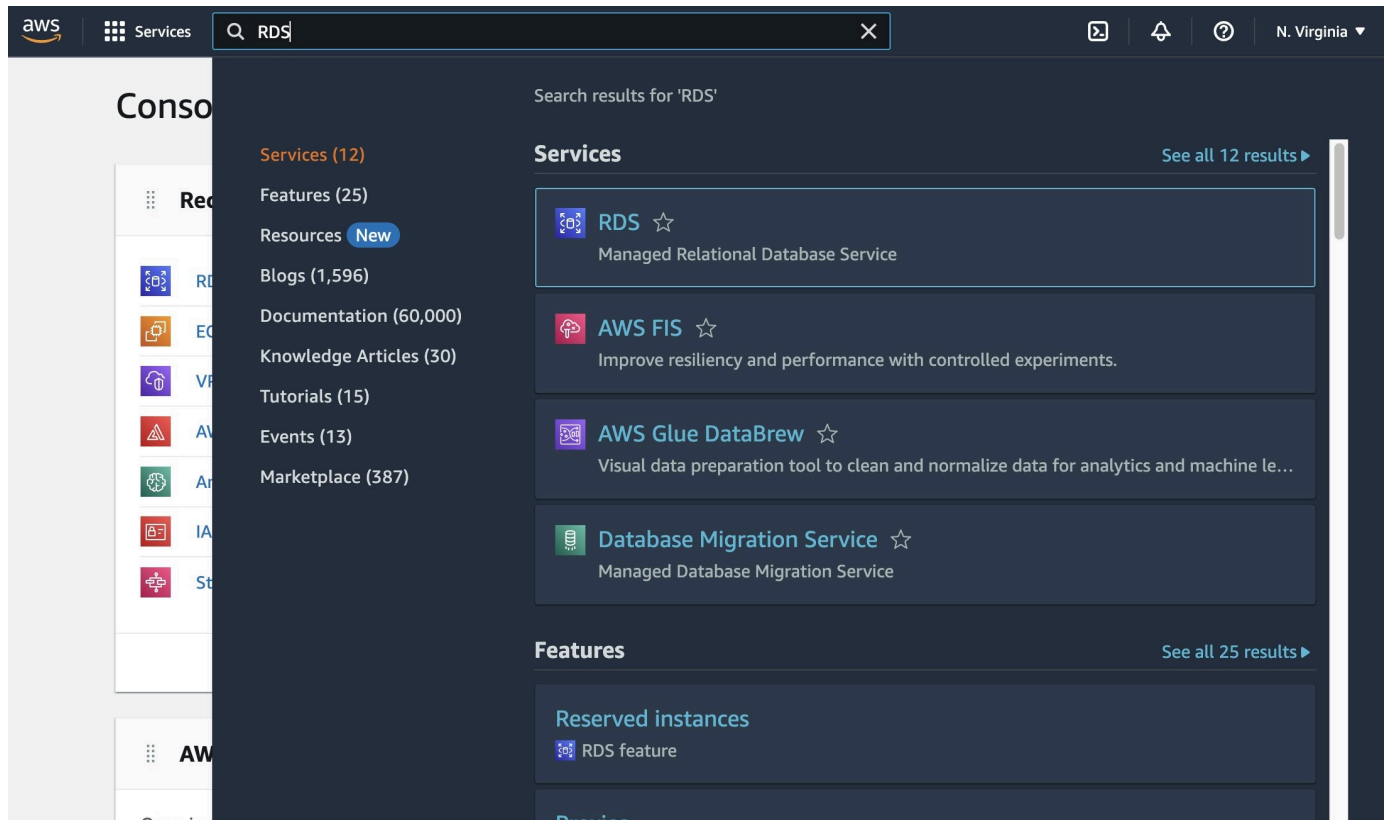
Implementation

Step 1: Create a Microsoft SQL Server DB instance

In this step, we will use Amazon RDS to create a Microsoft SQL Server DB instance with db.t3.small DB instance class, 20 GB of storage, and automated backups enabled with a retention period of one day.

1. Open the Amazon RDS console

Open the [AWS Management Console](#) in a new browser window, so you can keep this step-by-step guide open. In the console, enter **RDS** in the search bar and select **RDS** from the search results.

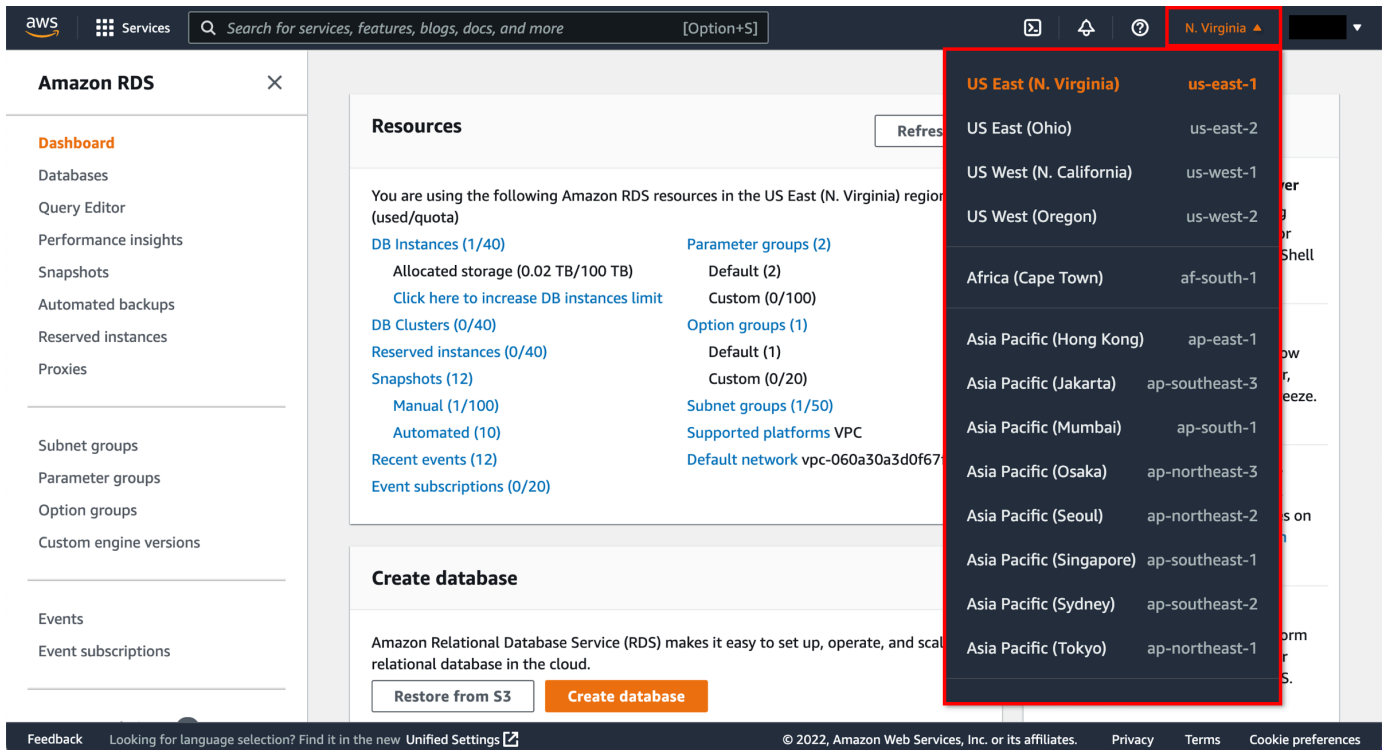


2. Choose a Region

In the top right corner of the Amazon RDS console, select the Region in which you want to create the DB instance.

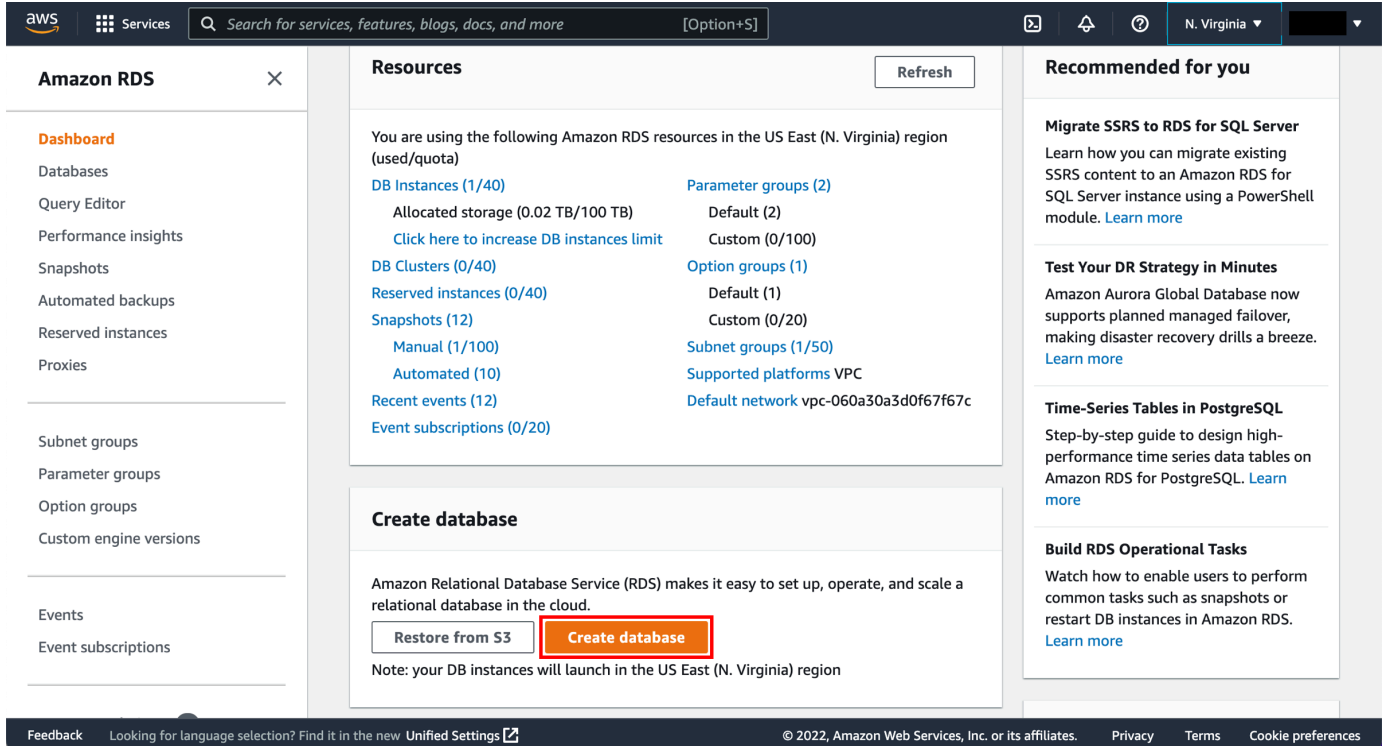
Note

AWS Cloud resources are housed in highly available data center facilities in different areas of the world. Each Region contains multiple distinct locations called Availability Zones. You have the ability to choose which Region to host your Amazon RDS activity in.



3. Create a database

In the **Create database** section, choose **Create database**.



4. Choose engine options

You now have options to select your engine. For this tutorial, choose the Microsoft SQL Server icon. In the Edition section, select SQL Server Express Edition. Leave the default values for License and Version.

Engine options

Engine type [Info](#)

Amazon Aurora



MySQL



MariaDB



PostgreSQL



Oracle

ORACLE®

Microsoft SQL Server



Database management type [Info](#)

Amazon RDS

RDS fully manages your database, including automatic patching. Choose this option if you don't need to customize your environment.

Amazon RDS Custom

RDS manages your database and gives you privileged access to the OS. Use this option if you want to customize the database, OS, and infrastructure.

Edition

SQL Server Express Edition

Affordable database management system that supports database sizes up to 10 GB.

SQL Server Web Edition

In accordance with Microsoft's licensing policies, it can only be used to support public and Internet-accessible webpages, websites, web applications, and web services.

SQL Server Standard Edition

Core data management and business intelligence capabilities for mission-critical applications and mixed workloads.

SQL Server Enterprise Edition

Comprehensive high-end capabilities for mission-critical applications with demanding database workloads and business intelligence requirements.

License

license-included

Version

SQL Server 2019 15.00.4198.2.v1



5. Configure basic settings

You will now configure your DB instance. Enter the configuration settings listed below:

Settings:

- **DB instance identifier:** Enter a name for the DB instance that is unique for your account in the Region that you selected. For this tutorial, enter **myrdstest**.
- **Master username:** Enter a username that you will use to log in to your DB instance. We will use **masterUsername** in this example.
- **Master password:** Enter a password that contains from 8 to 41 printable ASCII characters (excluding /, ", and @) for your master user password.
- **Confirm password:** Re-enter your password.

Settings

DB instance identifier [Info](#)

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username [Info](#)

Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. First character must be a letter.

Auto generate a password

Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password [Info](#)

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)

Burstable classes (includes t classes)

2 vCPUs 2 GiB RAM Network: 2,085 Mbps

Include previous generation classes

Storage

Storage type [Info](#)

Baseline performance determined by volume size

Allocated storage

6. Configure instance specifications

Now configure your instance specifications.

Instance specifications:

- **DB instance class:** Select the default, **db.t3.small — 2 vCPUs, 2 GiB RAM**. This equates to 2 GB memory and 2 vCPUs. To see a list of supported instance classes, see [Amazon RDS Pricing](#).
- **Storage type:** Select **General Purpose SSD (gp2)**. For more information about storage, see [Storage for Amazon RDS](#).
- **Allocated storage:** Select the default of 20 to allocate 20 GB of storage for your database. You can scale up to a maximum of 16 TB with Amazon RDS for SQL Server.
- **Option group:** Leave the default value. Amazon RDS uses option groups to enable and configure additional features. For more information, see [Working with Option Groups](#).
- **Enable storage autoscaling:** If your workload is cyclical or unpredictable, you would enable storage autoscaling to enable Amazon RDS to automatically scale up your storage when needed. This option does not apply to this tutorial.

The screenshot displays the AWS Management Console interface for configuring an Amazon RDS instance. The 'Instance configuration' section is active, showing options for the DB instance class and storage. The 'DB instance class' is set to 'db.t3.small' (2 vCPUs, 2 GiB RAM, Network: 2,085 Mbps). The 'Storage' section shows 'Storage type' as 'General Purpose SSD (gp2)' and 'Allocated storage' as '20 GiB'. A warning message states: 'Provisioning less than 100 GiB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. Learn more'. The 'Storage autoscaling' section has 'Enable storage autoscaling' unchecked.

7. Configure network settings


You are now on the **Connectivity** section, where you can provide information that Amazon RDS needs to launch the Microsoft SQL Server DB instance. See the following list for the example settings for your DB instance.

Connectivity

- **Network type:** Keep the default **IPv4**.
- **Virtual Private Cloud (VPC):** Select **Default VPC**. For more information about VPC, see [Amazon RDS and Amazon Virtual Private Cloud \(VPC\)](#).

Additional connectivity configurations

- **Subnet group:** Choose the **default** subnet group. For more information about subnet groups, see [Working with DB Subnet Groups](#).
- **Public access:** Choose **Yes**. This will allocate an IP address for your database instance so that you can directly connect to the database from your own device.

 **Note**

You will incur charges of \$0.005 per hour.

- **VPC security groups:** Select **Create new VPC security group**. This will create a security group that will allow connection from the IP address of the device that you are currently using to the database created.
- **New VPC security group name:** For this tutorial, enter **myrdstest**.
- **Availability zone:** Choose **No preference**. See [Regions and Availability Zones](#) for more details.
- **Port:** Leave the default value of 1433.

Microsoft SQL Server Windows Authentication

- **Directory:** Leave this option disabled.

Connectivity

Network type [Info](#)
To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

IPv4
Your resources can communicate only over the IPv4 addressing protocol.

Dual-stack mode
Your resources can communicate over IPv4, IPv6, or both.

Virtual private cloud (VPC) [Info](#)
VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-060a30a3d0f67f67c)

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

Subnet group [Info](#)
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default-vpc-060a30a3d0f67f67c

Public access [Info](#)

Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

New VPC security group name
myrdstest

Availability Zone [Info](#)
No preference

Additional configuration

Database port [Info](#)
TCP/IP port that the database will use for application connections.

1433

Microsoft SQL Server Windows Authentication

Choose a directory in which you want to allow authorized domain users to authenticate with this SQL Server instance using Windows Authentication.

Enable Microsoft SQL Server Windows authentication
By choosing a directory and continuing with database instance creation you authorize Amazon RDS to create the IAM role necessary for using Windows Authentication

8. Configure additional options

In the **Additional configurations** section:

Database options

- **DB parameter group:** Leave the **default value**. For more information, see [Working with DB Parameter Groups](#).
- **Option group:** Leave the **default value**. Amazon RDS uses option groups to enable and configure additional features. For more information, see [Working with Option Groups](#).

Backup

- **Backup retention period:** You can choose the number of days to retain the backup you take. For this tutorial, set this value to **1 day**.
- **Backup window:** Use the default of **No preference**.

Performance Insights

For this tutorial, do not select **Turn on performance insights**. When this option is enabled, you will receive advanced database performance-monitoring features that make it easy to diagnose and solve performance challenges on Amazon RDS databases.

Monitoring

- **Enhanced monitoring:** Use the default of **Enable Enhanced monitoring**. Enabling Enhanced monitoring will give you metrics in real time for the operating system (OS) that your DB instance runs on. For more information, see [Viewing DB Instance Metrics](#).

Maintenance

- **Auto minor version upgrade:** Select **Enable auto minor version upgrade** to receive automatic updates when they become available.
- **Maintenance window:** Select **No preference**.

Deletion protection

Do not select **Enable deletion protection** for this tutorial. When this option is enabled, you're prevented from accidentally deleting the database.

aws Services Search for services, features, blogs, docs, and more [Option+S] N. Virginia

Additional configuration

Database options, backup turned on, backtrack turned off, Performance Insights turned off, Enhanced Monitoring turned on, maintenance, CloudWatch Logs, delete protection turned off.

Database options

DB parameter group [Info](#)
default.sqlserver-ex-15.0

Option group [Info](#)
default.sqlserver-ex-15-00

Time zone
No preference

Collation [Info](#)

Backup

Enable automated backups
Creates a point-in-time snapshot of your database

Backup retention period [Info](#)
The number of days for which automated backups are retained. You can choose a number from 1 to 35.
1 day

Backup window [Info](#)
The daily time range (in UTC) during which automated backups occur.
 Choose window
 No preference

Copy tags to snapshots

Performance Insights [Info](#)
 Turn on Performance Insights

Monitoring

Enable Enhanced monitoring
Enabling Enhanced monitoring metrics are useful when you want to see how different processes or threads use the CPU.

Granularity
60 seconds

Monitoring Role
default
Clicking "Create database" will authorize RDS to create the IAM role rds-monitoring-role

Log exports

Select the log types to publish to Amazon CloudWatch Logs
 Error log

IAM role

The following service-linked role is used for publishing logs to CloudWatch Logs.
RDS service-linked role

Maintenance

Auto minor version upgrade [Info](#)
 Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the database.

Maintenance window [Info](#)
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.
 Choose window
 No preference

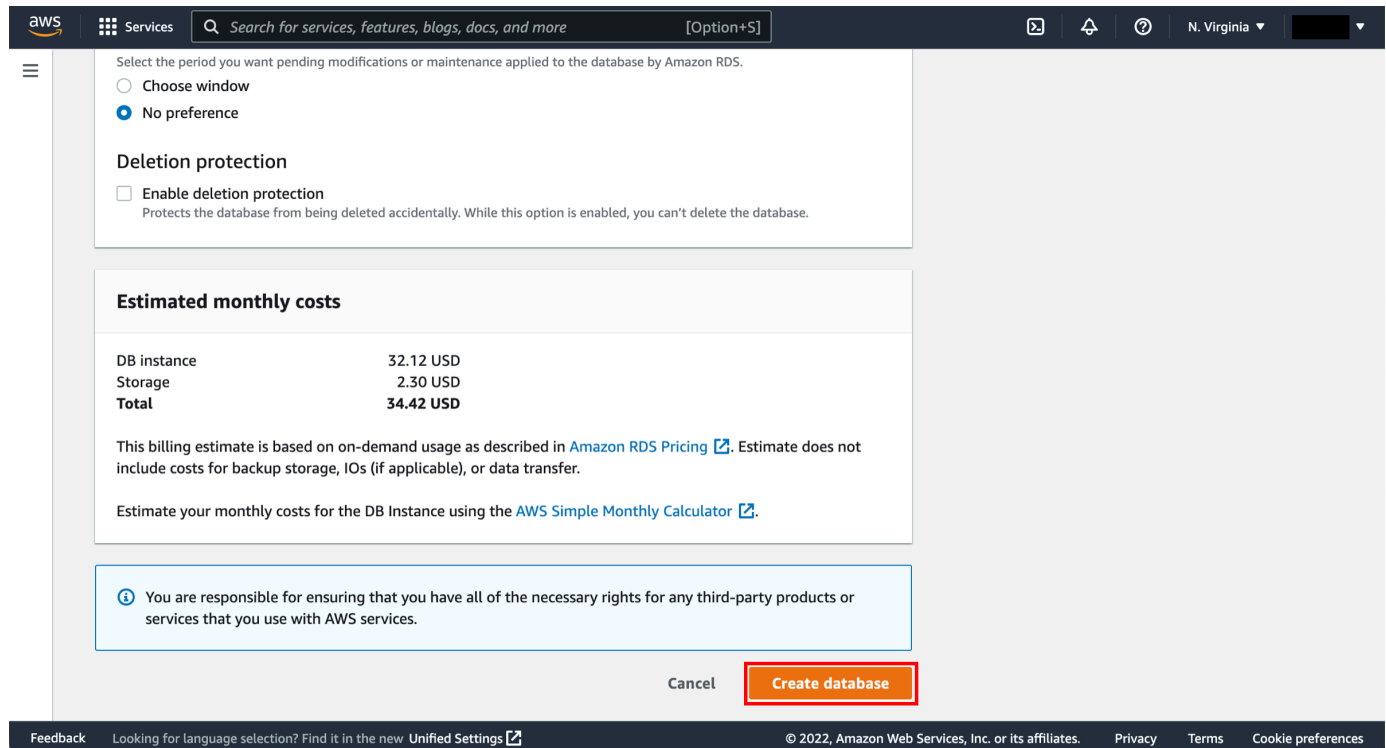
Deletion protection

Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

9. Review costs

At the bottom of the creation wizard, AWS will show you estimated monthly costs for your Amazon RDS database. If you are still eligible for the [Amazon RDS Free Tier](#), you will see a note that the database will be free to you for up to 12 months.

Choose the **Create database** button to create your database.



The screenshot shows the AWS RDS console interface. At the top, there's a search bar and navigation icons. Below that, there are radio buttons for 'Choose window' and 'No preference' (selected). A 'Deletion protection' section has an unchecked checkbox for 'Enable deletion protection'. The 'Estimated monthly costs' section displays a table:

Category	Cost (USD)
DB instance	32.12
Storage	2.30
Total	34.42

Below the table, there's a note: 'This billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.' A link to 'AWS Simple Monthly Calculator' is also present. At the bottom, there's a 'Cancel' button and a prominent orange 'Create database' button.

10. Monitor database creation

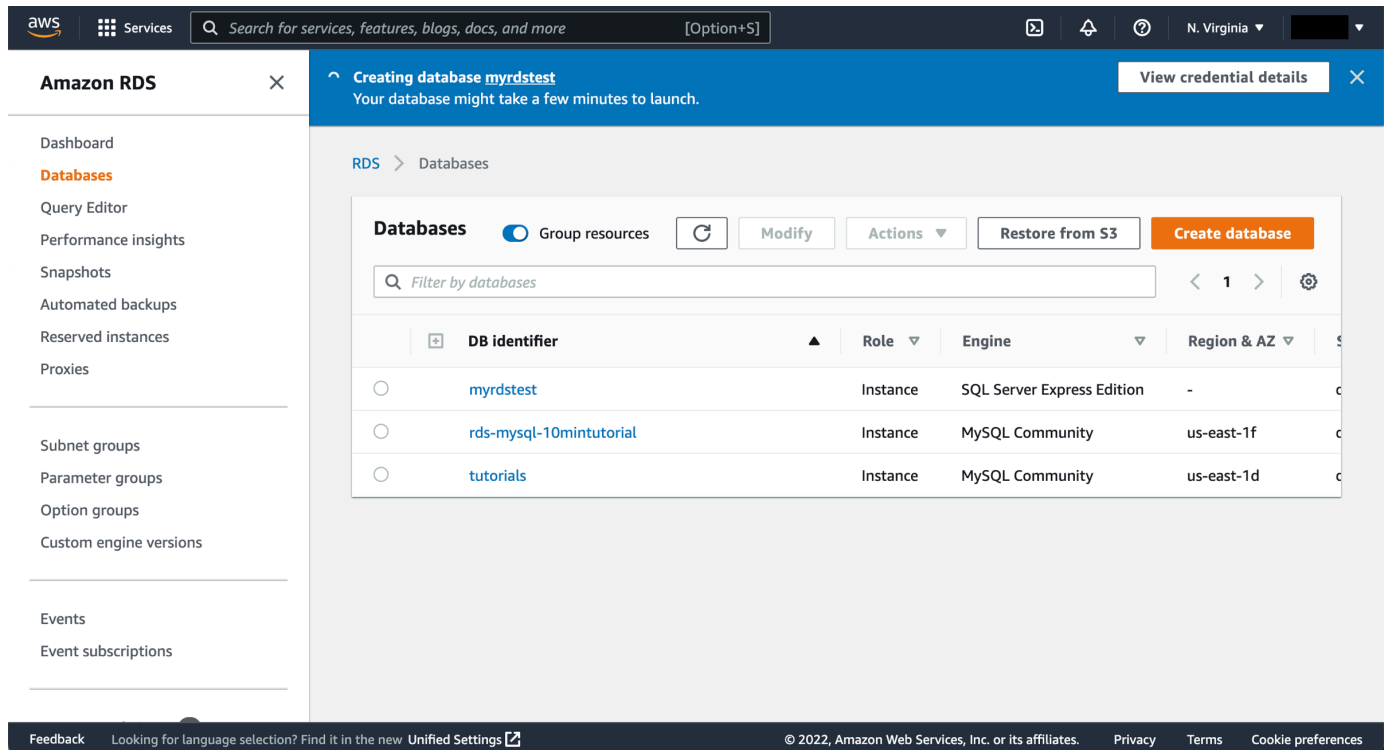
Your DB Instance is now being created. Choose **View Your DB Instances**.

Note

Depending on the DB instance class and storage allocated, it could take several minutes for the new DB instance to become available.

The new DB instance appears in the list of DB instances on the Amazon RDS console. The DB instance will have a status of creating until the DB instance is created and ready for use. When the state changes to available, you can connect to a database on the DB instance.

Feel free to move on to the next step as you wait for the DB instance to become available.



The screenshot shows the Amazon RDS console interface. At the top, there's a navigation bar with the AWS logo, 'Services' menu, a search bar, and the region 'N. Virginia'. The main content area is titled 'Creating database myrdstest' with a sub-message 'Your database might take a few minutes to launch.' and a 'View credential details' button. Below this, the 'Databases' section is active, showing a table of existing databases. The table has columns for 'DB identifier', 'Role', 'Engine', and 'Region & AZ'. Three databases are listed: 'myrdstest' (SQL Server Express Edition), 'rds-mysql-10mintutorial' (MySQL Community), and 'tutorials' (MySQL Community).

DB identifier	Role	Engine	Region & AZ
myrdstest	Instance	SQL Server Express Edition	-
rds-mysql-10mintutorial	Instance	MySQL Community	us-east-1f
tutorials	Instance	MySQL Community	us-east-1d

Step 2: Download a SQL client

Once the database instance creation is complete and the status changes to available, you can connect to a database on the DB instance using any standard SQL client. In this step, we will download Microsoft SQL Server Management Studio, a popular client for SQL Server.

- Download SQL Server Management Studio

Go to the [SQL Documentation](#), under which you will find SQL tools. Look for [SQL Server Management Studio \(SSMS\)](#) and download the latest version.

Note

Remember to download the SQL client to the same device from which you created the RDS DB Instance. The security group your database is placed in is configured to allow connection only from the device from which you created the DB instance.

Version

SQL Server 2022

Filter by title

SQL Server Management Studio (SSMS)

Download SSMS

Release notes

> Preview

> Overview

> Quickstarts

> Tutorials

> Concepts

> How-to

> References

> Resources

> SqlPackage.exe

> SQL Server Profiler

> Visual Studio native helpers

> Extended Features

> Visual Studio Code

> Tutorials

> SQL Server on Linux

> SQL on Azure

> Azure Arc-enabled SQL Server

> Resources

> Reference

Learn / SQL / Tools /

+

✉

✎

⋮

Download SQL Server Management Studio (SSMS)

Article • 11/18/2022 • 7 minutes to read • 46 contributors

Feedback

Applies to: SQL Server (all supported versions) Azure SQL Database Azure SQL Managed Instance Azure Synapse Analytics

SQL Server Management Studio (SSMS) is an integrated environment for managing any SQL infrastructure, from SQL Server to Azure SQL Database. SSMS provides tools to configure, monitor, and administer instances of SQL Server and databases. Use SSMS to deploy, monitor, and upgrade the data-tier components used by your applications, and build queries and scripts.

Use SSMS to query, design, and manage your databases and data warehouses, wherever they are - on your local computer, or in the cloud.

Download SSMS

To download SSMS 19 Preview 3, visit [Download SSMS 19](#).

[↓ Free Download for SQL Server Management Studio \(SSMS\) 18.12.1](#)

SSMS 18.12.1 is the latest general availability (GA) version. If you have a previous GA version of SSMS 18 installed, installing SSMS 18.12.1 upgrades it to 18.12.1.

- Release number: 18.12.1
- Build number: 15.0.18424.0
- Release date: June 21, 2022

Step 3: Connect to the Microsoft SQL Server database

In this step, you will connect to the database you created using SQL Server Management Studio.

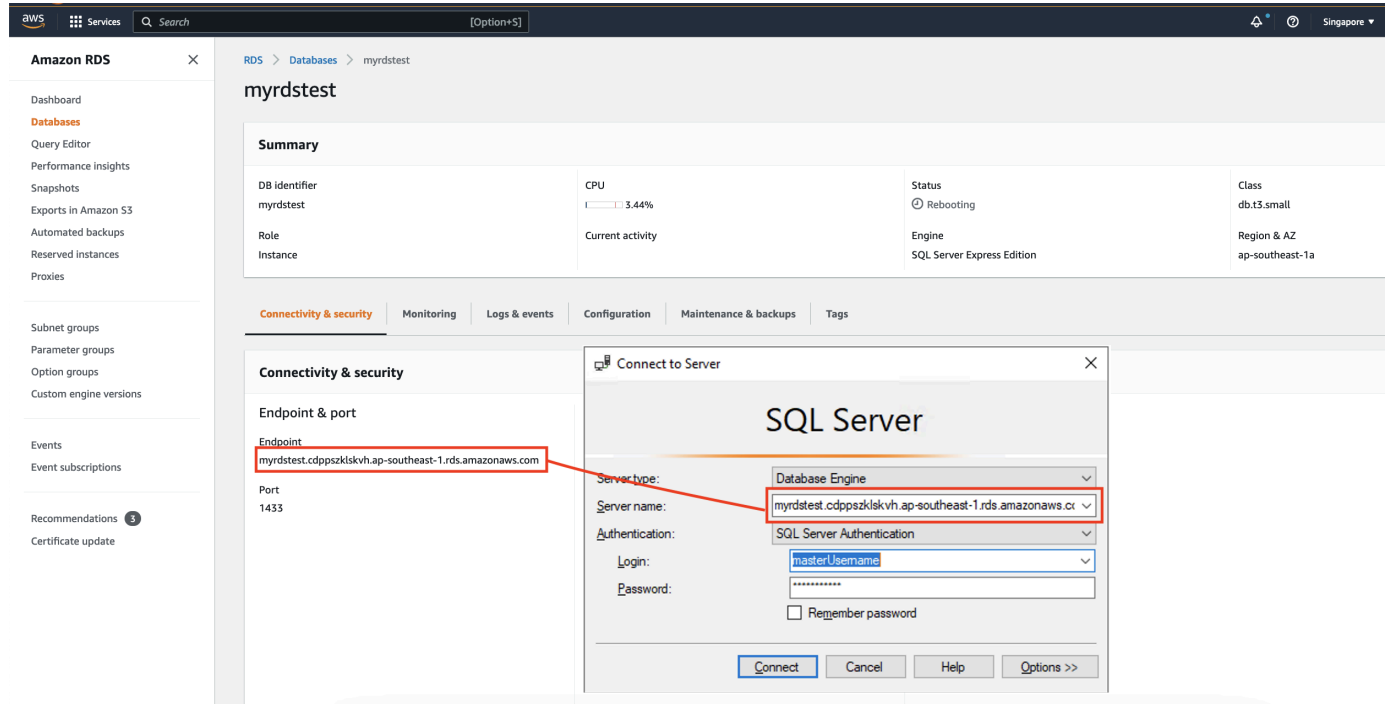
1. Configure SSMS connection settings

Once you have completed your download, install and open the program. A dialog box appears. Enter the following:

- **Server type:** Select Database Engine
- **Hostname:** Copy and paste the hostname from the Amazon RDS console as shown in the screenshot to the right. Afterwards, change the colon between the DNS and port number to a comma. For example, your server name should look like **sample-instance.cg034hpkmmjt.us-east-1.rds.amazonaws.com,1433**

- **Username:** Type in the username you created for the Amazon RDS database. Our example is **masterUsername**.
- **Password:** Enter the password you used while creating the Amazon RDS database.

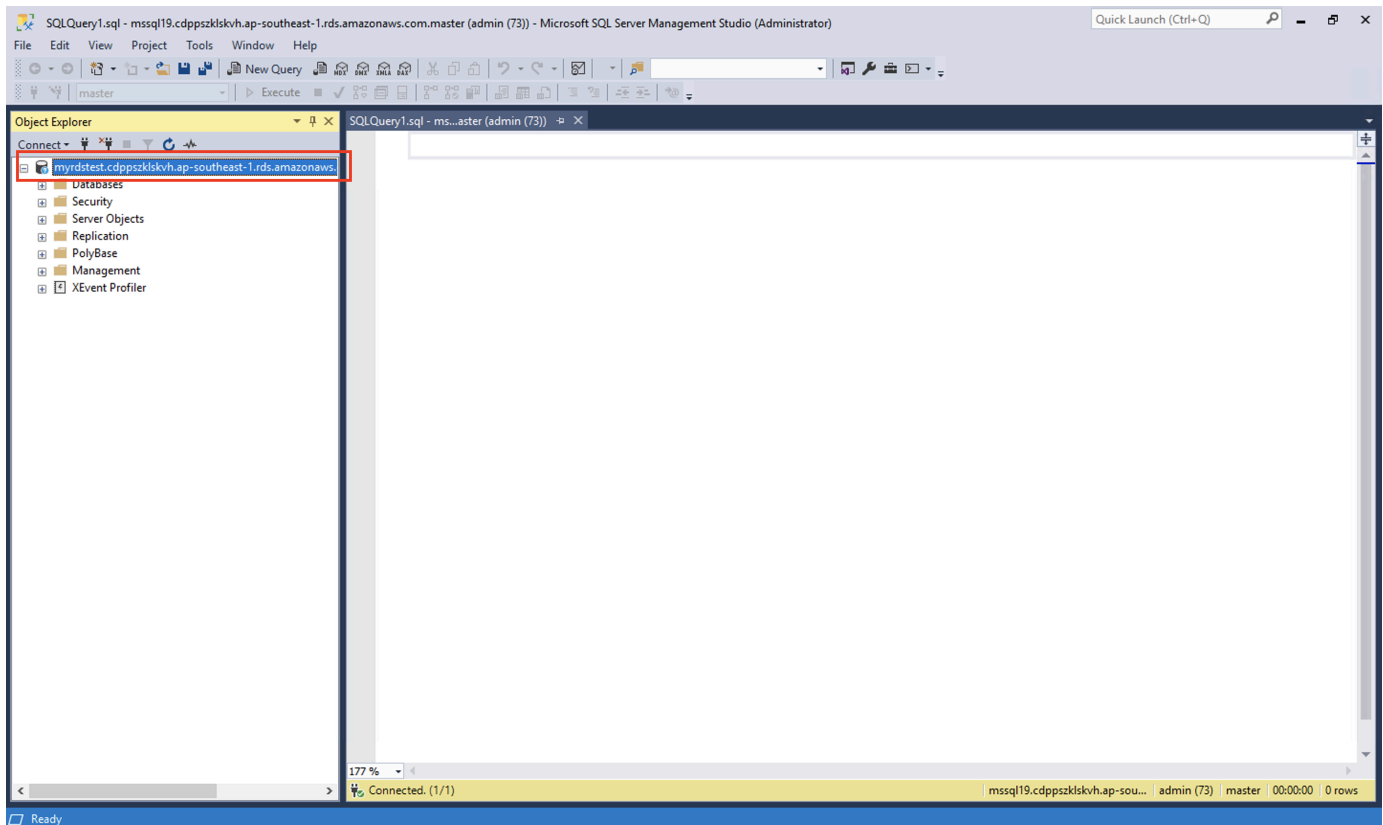
Choose Connect.



The screenshot displays the AWS Management Console for an Amazon RDS instance named 'myrdstest'. The 'Connect to Server' dialog box is open, showing the 'SQL Server' connection configuration. The 'Server name' field is populated with 'myrdstest.cdppszklskvh.ap-southeast-1.rds.amazonaws.com', which is highlighted with a red box. The 'Login' field is populated with 'masterUsername', also highlighted with a red box. The 'Server type' is set to 'Database Engine' and 'Authentication' is set to 'SQL Server Authentication'. The 'Connect' button is visible at the bottom of the dialog.

2. Verify database connection

You are now connected to the database. In the SQL Server Management Studio, you will see various schema objects available in the database. Now you can create tables, insert data, and run queries.

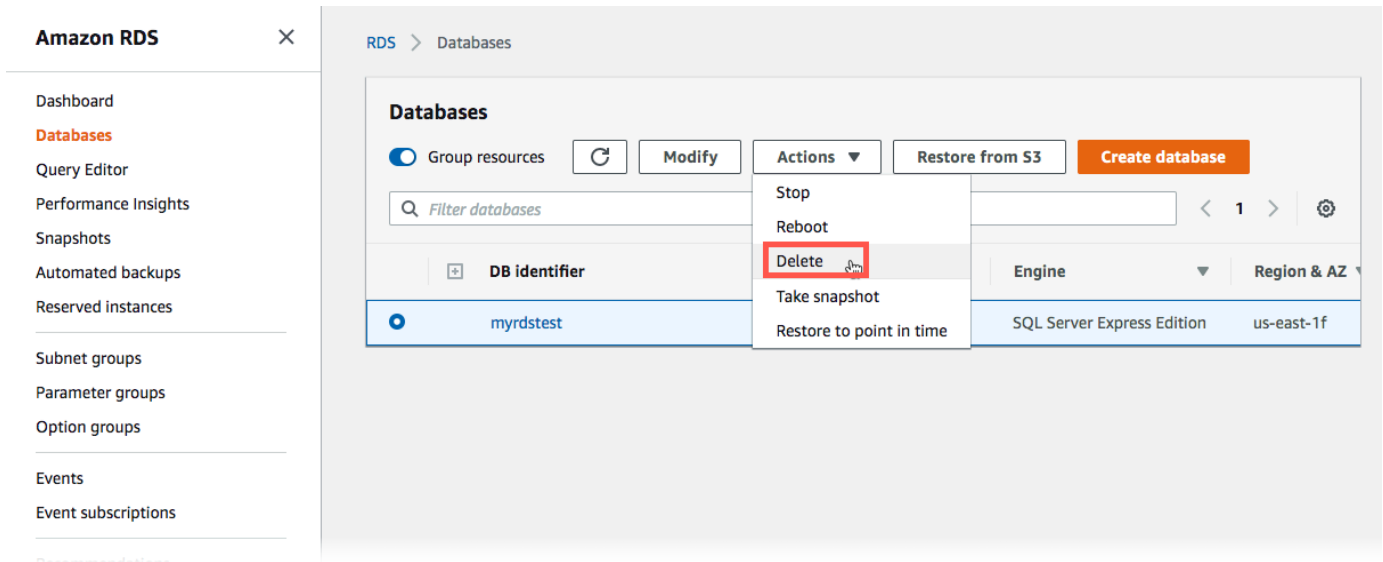


(Optional) Delete the DB instance

You can easily delete the Microsoft SQL Server DB instance from the Amazon RDS console. It is a best practice to delete instances that you are no longer using so that you don't keep getting charged for them.

1. Delete the instance

Go back to the Amazon RDS console. Select **Databases**, choose the instance that you want to delete, and then select **Delete** from the **Actions** dropdown menu.



The screenshot shows the Amazon RDS console interface. On the left is a navigation menu with options like Dashboard, Databases, Query Editor, Performance Insights, Snapshots, Automated backups, Reserved instances, Subnet groups, Parameter groups, Option groups, Events, and Event subscriptions. The main content area is titled 'Databases' and shows a table of database instances. One instance, 'myrdstest', is selected. An 'Actions' dropdown menu is open, and the 'Delete' option is highlighted with a red box. Other options in the menu include Stop, Reboot, Take snapshot, and Restore to point in time. The table also shows columns for 'Engine' (SQL Server Express Edition) and 'Region & AZ' (us-east-1f).

2. Confirm deletion

You are asked to create a final snapshot and to confirm the deletion. For our example, do not create a final snapshot, acknowledge that you want to delete the instance, and then choose **Delete**.

Note

Deleting your DB Instance may take a few minutes

Delete myrdstest instance? ✕

Are you sure you want to Delete the **myrdstest** DB Instance?

Create final snapshot?
Determines whether a final DB Snapshot is created before the DB instance is deleted.

Retain automated backups
Determines whether retaining automated backups for 1 days after deletion

I acknowledge that upon instance deletion, automated backups, including system snapshots and point-in-time recovery, will no longer be available.

To confirm deletion, type *delete me* into the field

delete me|

⚠ We strongly recommend taking a final snapshot before instance deletion since after your instance is deleted, automated backups will no longer be available.

Cancel **Delete**

Conclusion

Congratulations! You have created, connected to, and deleted a Microsoft SQL Server database instance with [Amazon RDS](#). Amazon RDS makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to focus on your applications and business.