



14. RELATIONAL DATABASE SERVICE (RDS)

RELATIONAL DATABASE SERVICE (RDS)

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

DB Instances: The basic building block of Amazon RDS is the DB instance. A DB instance is an isolated database environment in the cloud. A DB instance can contain multiple user-created databases, and you can access it by using the same tools and applications that you use with a stand-alone database instance.

Each DB instance runs a DB engine. Amazon RDS currently supports the MySQL, Maria DB, PostgreSQL, Oracle, and Microsoft SQL Server DB engines.

For each DB instance, you can select from 20 GB to 16 TB of associated storage capacity. Each DB instance class has minimum and maximum storage requirements for the DB instances that are created from it. It's important to have enough storage so that your databases have room to grow and that features for the DB engine have room to write content or log entries.

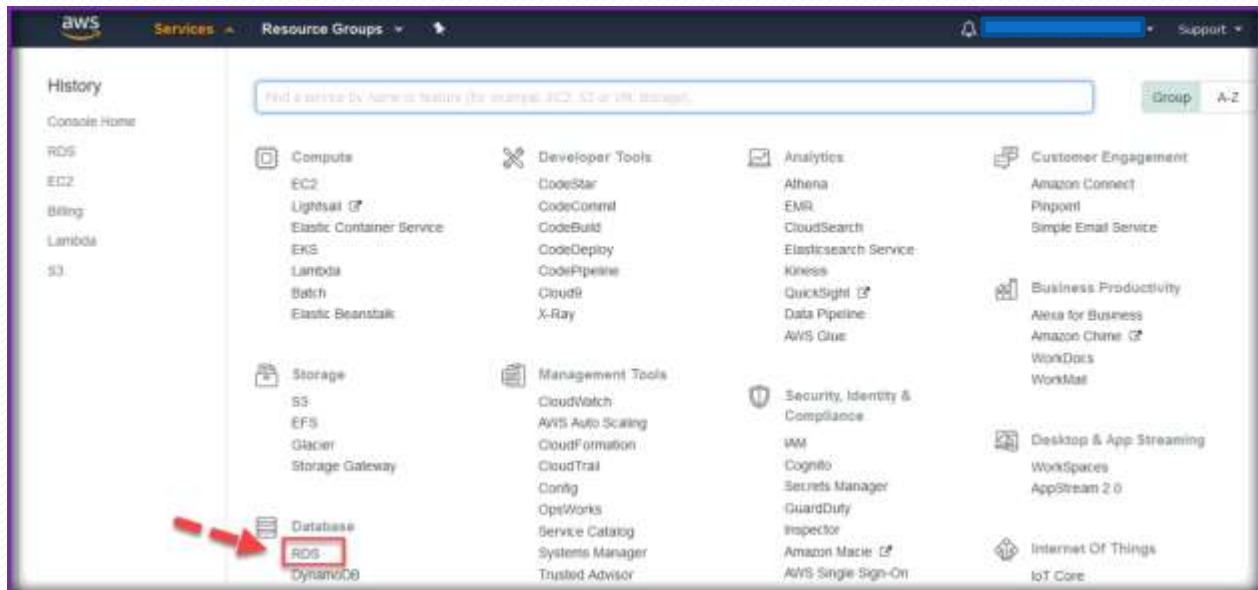
DB instance storage comes in three types: Magnetic, General Purpose (SSD), and Provisioned IOPS (SSD). They differ in performance characteristics and price, allowing you to tailor your storage performance and cost to the needs of your database.

Considerations for running on EC2:

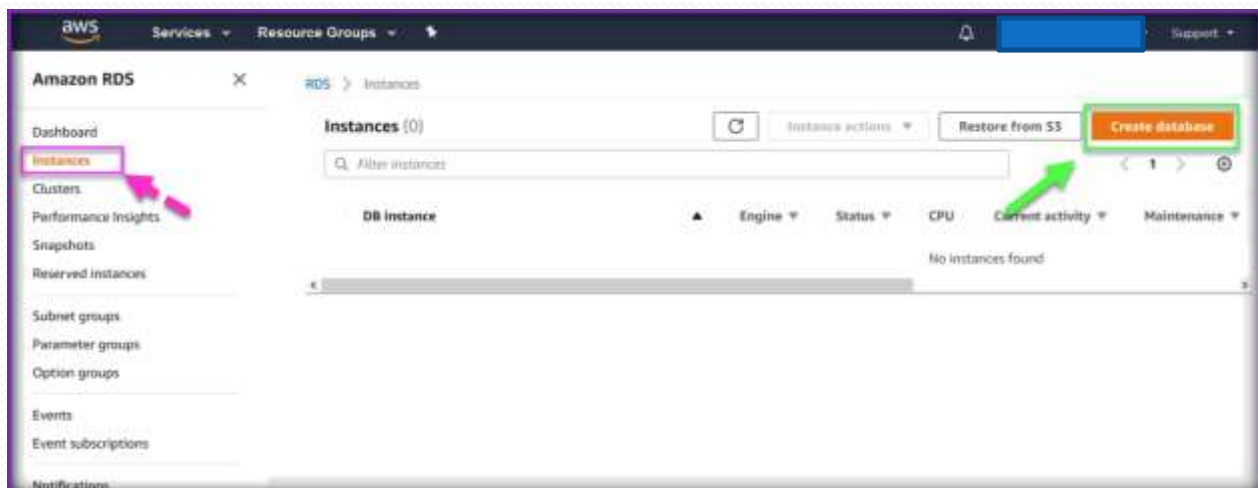
- Have to build and design for high availability.
- Must implement own backup solutions.
- Must manage replication manually between master and slave databases.
- Must manage tighter security controls.
- Upgrading resources or any type of failover requires manual interaction or custom scripting.

CREATING RDS DATABASE INSTANCE ON AWS

Once you logged in to AWS management console, Choose RDS from Database section under AWS services.



In the next page, select Instances from the left pane and click on Create database to start spinning up a database instance.



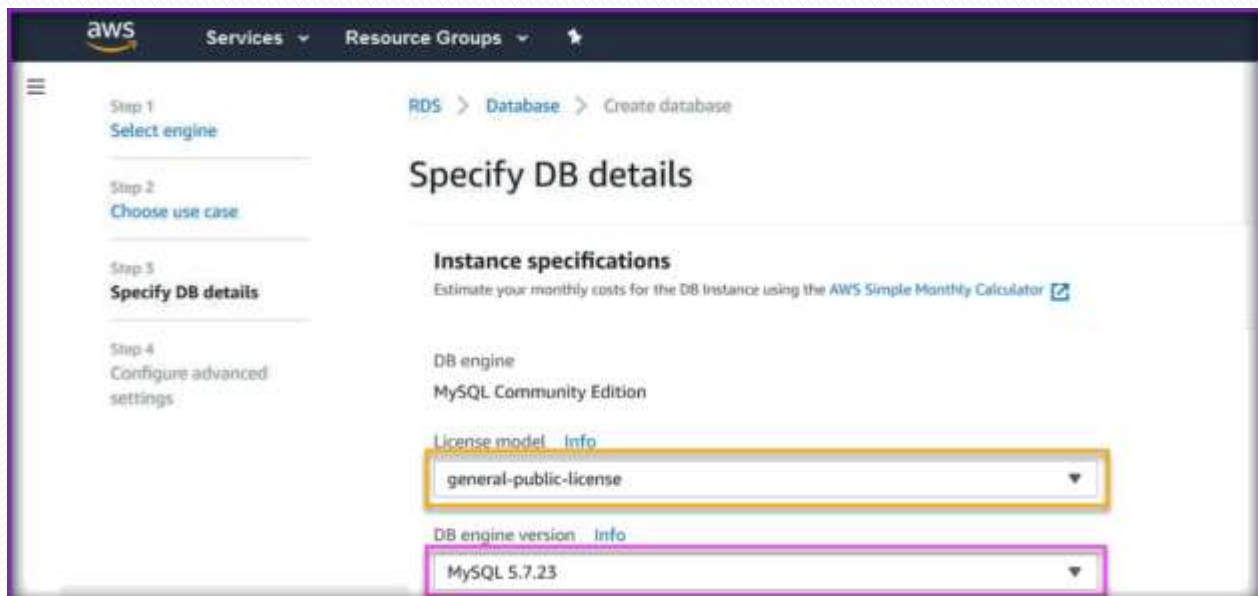
Select the database flavor and click on next to continue.

The screenshot shows the 'Select engine' step of the AWS RDS 'Create database' wizard. The breadcrumb navigation at the top reads 'RDS > Database > Create database'. On the left, a sidebar lists four steps: 'Step 1: Select engine' (active), 'Step 2: Choose use case', 'Step 3: Specify DB details', and 'Step 4: Configure advanced settings'. The main area is titled 'Select engine' and contains a section 'Engine options' with six database engine cards: Amazon Aurora, MySQL, MariaDB, PostgreSQL, Oracle, and Microsoft SQL Server. The MySQL card is selected, indicated by a blue radio button and a green rectangular highlight. Below the cards, the 'MySQL' section provides a description: 'MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.' It lists four bullet points: 'Supports database size up to 16 TiB.', 'Instances offer up to 32 vCPUs and 244 GiB Memory.', 'Supports automated backup and point-in-time recovery.', and 'Supports cross-region read replicas.' At the bottom, there is a checkbox 'Only enable options eligible for RDS Free Usage Tier' with an 'Info' link, and two buttons: 'Cancel' and 'Next' (highlighted with a blue border).

Choose use case as Dev/Test -Mysql and then click on Next.

The screenshot shows the 'Choose use case' step of the AWS RDS 'Create database' wizard. The breadcrumb navigation at the top reads 'RDS > Database > Create database'. On the left, the same sidebar as the previous screenshot is visible. The main area is titled 'Choose use case' and contains a section 'Use case' with the question 'Do you plan to use this database for production purposes?'. Below this, there are three use case options: 'Production - Amazon Aurora' (marked 'Recommended'), 'Production - MySQL', and 'Dev/Test - MySQL' (selected with a blue radio button and a green rectangular highlight). A green arrow points to the 'Dev/Test - MySQL' option. The 'Dev/Test - MySQL' option includes the text: 'This instance is intended for use outside of production or under the RDS Free Usage Tier.' Below the options, it states 'Billing is based on RDS pricing' with a link icon. At the bottom, there are three buttons: 'Cancel', 'Previous', and 'Next' (highlighted with a red border and a red arrow pointing to it).

Select license model and engine version as per your requirement.



aws Services Resource Groups

RDS > Database > Create database

Step 1 Select engine
Step 2 Choose use case
Step 3 Specify DB details
Step 4 Configure advanced settings

Specify DB details

Instance specifications
Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#)

DB engine
MySQL Community Edition

License model [Info](#)
general-public-license

DB engine version [Info](#)
MySQL 5.7.23

Select Instance type and specify Multi AZ information, choose Storage type, and specify the storage for your database in gb's.



DB instance class [Info](#)
db.t2.micro — 1 vCPU, 1 GiB RAM

Multi-AZ deployment [Info](#)
☐ Create replica in different zone
Creates a replica in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.
☒ No **Choose No**

Storage type [Info](#)
General Purpose (SSD)

Allocated storage
20 GiB
(Minimum: 20 GiB, Maximum: 16384 GiB) Higher allocated storage [may improve](#) IOPS performance.

Next, specify a tag name, provide user name and password which you want to use to connect to database server.

Settings

DB instance identifier [Info](#)
Specify a name that is unique for all DB instances owned by your AWS account in the current region.

mysql-db-server

DB instance identifier is case insensitive, but stored as all lower-case, as in "mydbinstance". Must contain from 1 to 63 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Master username [Info](#)
Specify an alphanumeric string that defines the login ID for the master user.

admin

Master Username must start with a letter. Must contain 1 to 16 alphanumeric characters.

Master password [Info](#) **Confirm password** [Info](#)

.....

Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "", or "@".

Cancel Previous **Next**

Under Network & Security choose VPC configurations.

- Select VPC from VPC drop down list.
- Select Default for Subnet Group.
- Choose either Yes or No for Public accessibility.
- Specify an Availability Zone from the drop-down list.
- Choose VPC security groups from the list.

Configure advanced settings

Network & Security

Virtual Private Cloud (VPC) [Info](#)

VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-d21a42ba)



Only VPCs with a corresponding DB subnet group are listed.

Subnet group [Info](#)

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default

Public accessibility [Info](#)

☒ Yes

EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

☐ No

DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

Availability zone [Info](#)

ap-south-1a

VPC security groups

Security groups have rules authorizing connections from all the EC2 instances and devices that need to access the DB instance.

☐ Create new VPC security group

☒ Choose existing VPC security groups

Choose VPC security groups

Linux-Firewall X

Database options

Database name [Info](#)

mydatabase

Note: if no database name is specified then no initial MySQL database will be created on the DB instance.

Port [Info](#)

To port the DB instance will use for application connections

3306

DB parameter group [Info](#)

default:mysql5.7

Option group [Info](#)

default:mysql-5-7

IAM DB authentication [Info](#)

☐ Enable IAM DB authentication

Manage your database user credentials through AWS IAM users and roles.

☒ Disable

Under database options, specify a database name to be created with in the rds instance.

- Specify Database port to run RDS instance.
- Choose DB Parameter and Option groups.
- Choose Disable IAM DB authentication.

Database options

Database name [Info](#)

Note: If no database name is specified then no initial MySQL database will be created on the DB instance.

Port [Info](#)
TCP/IP port the DB instance will use for application connections

DB parameter group [Info](#)

Option group [Info](#)

IAM DB authentication [Info](#)

☐ Enable IAM DB authentication
Manage your database user credentials through AWS IAM users and roles.

☒ Disable

We can specify the encryption, if we need for the database instance.

Encryption

Encryption

☐ Enable encryption [Learn more](#) 
Select to encrypt the given instance. Master key ids and aliases appear in the list after they have been created using the Key Management Service(KMS) console.

☐ Disable encryption

 The selected engine or DB instance class does not support storage encryption.

Under the Backup section, specify backup, monitoring, and maintenance details.

- Like how many days of backups you need for this RDS instance.
- Backup window period time and duration.

Backup

Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail [here](#).

Backup retention period [Info](#)
Select the number of days that Amazon RDS should retain automatic backups of this DB instance.

7 days ▼

Backup window [Info](#)

☒ Select window
☐ No preference

Start Time

10 ▼ : 00 ▼ UTC

Duration

2 ▼ hours

☒ Copy tags to snapshots

Choose the monitoring mode as enable Enhanced monitoring or disable.


Monitoring

Enhanced monitoring

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

☒ Disable enhanced monitoring

Select which logs from your db server to export to cloud watch service.



Log exports

Select the log types to publish to Amazon CloudWatch Logs

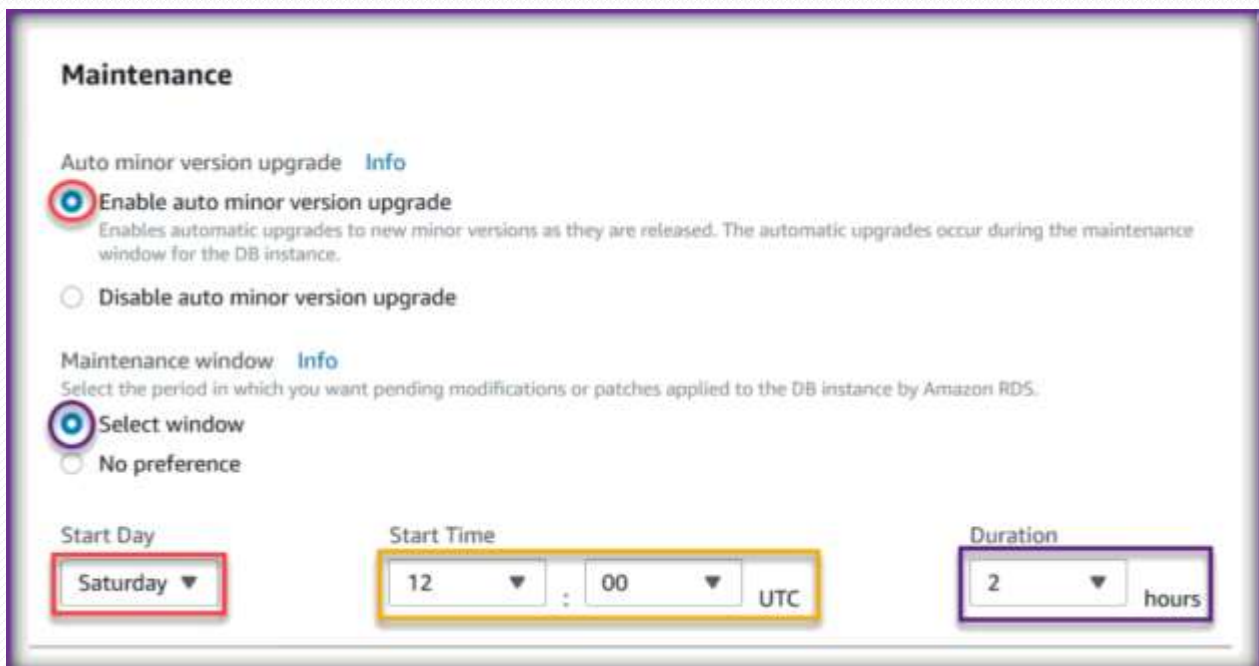
- ☐ Audit log
- ☒ Error log
- ☐ General log
- ☒ Slow query log

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

RDS Service Linked Role

Under maintenance section choose following things.

- Minor upgrades and patches to be installed or not.
- Specify when to install upgrades and patches in UTC format.



Maintenance

Auto minor version upgrade [Info](#)

- ☒ Enable auto minor version upgrade
Enables automatic upgrades to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the DB instance.
- ☐ Disable auto minor version upgrade

Maintenance window [Info](#)
Select the period in which you want pending modifications or patches applied to the DB instance by Amazon RDS.

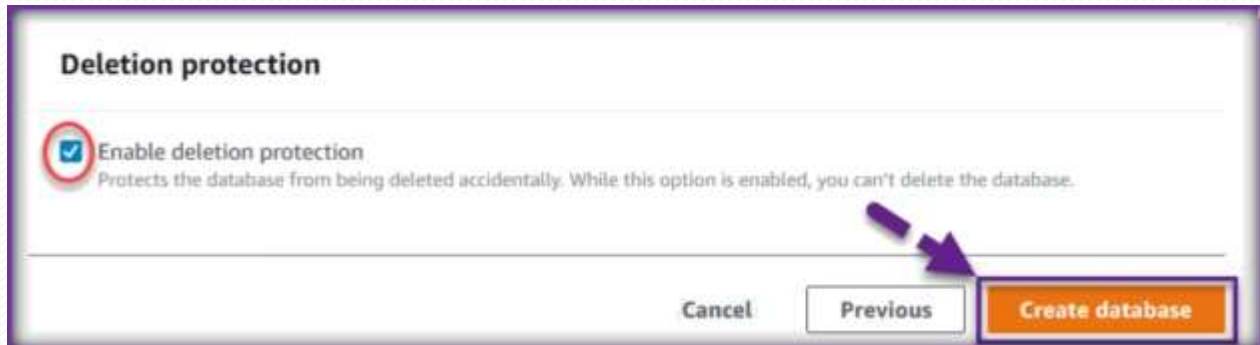
- ☒ Select window
- ☐ No preference

Start Day
Saturday ▼

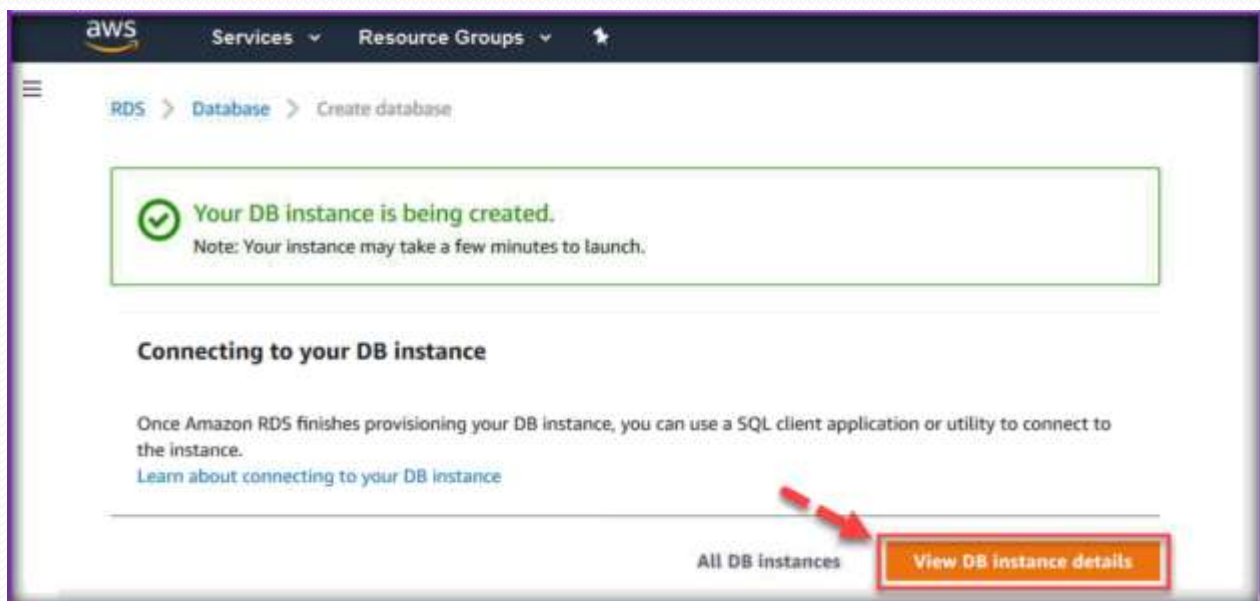
Start Time
12 ▼ : 00 ▼ UTC

Duration
2 ▼ hours

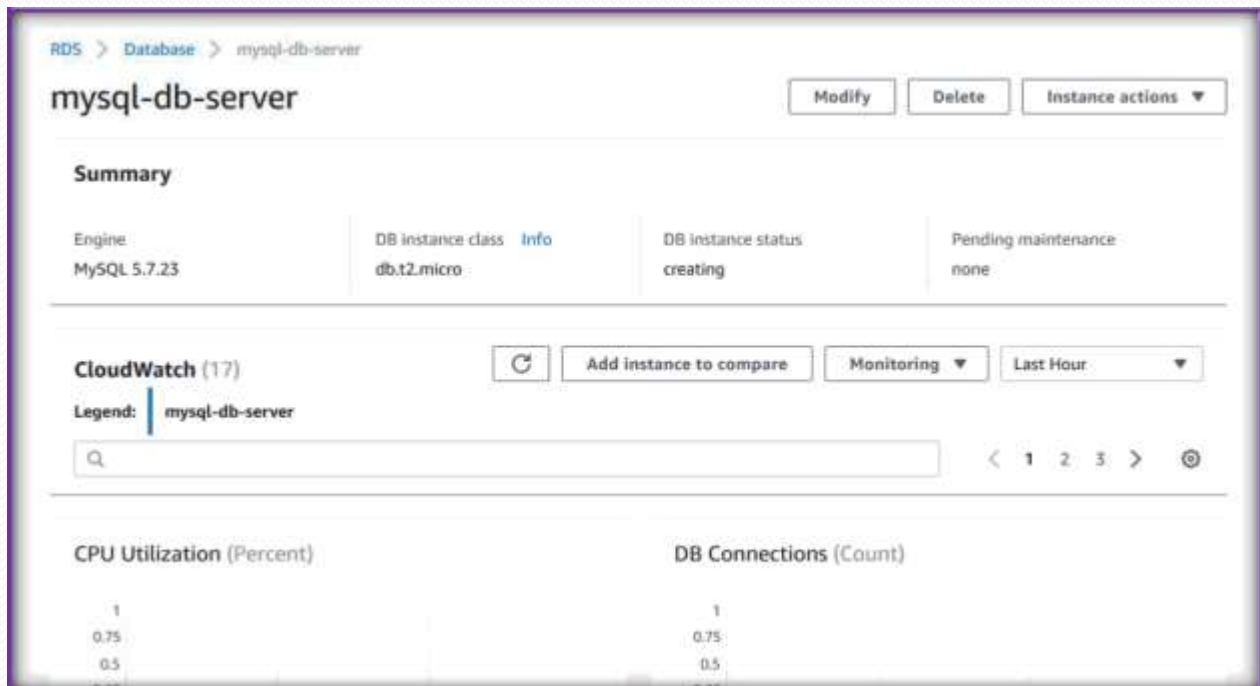
Specify deletion protection, then click on Create Database to create the DB server.



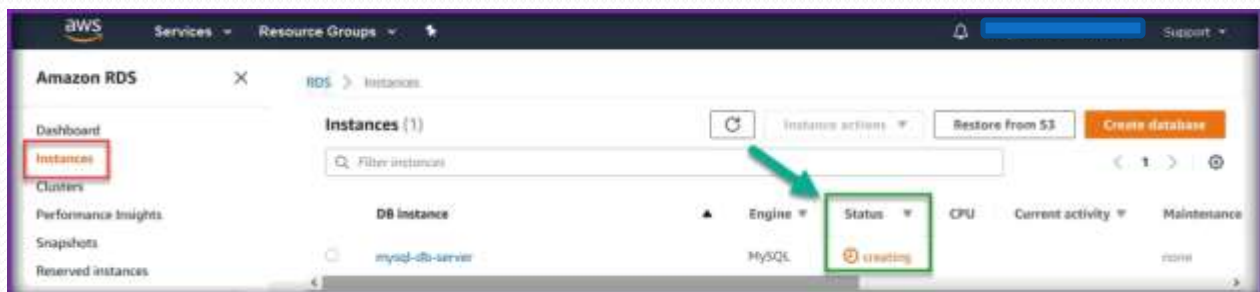
Click on View your DB Instances to see the instances.



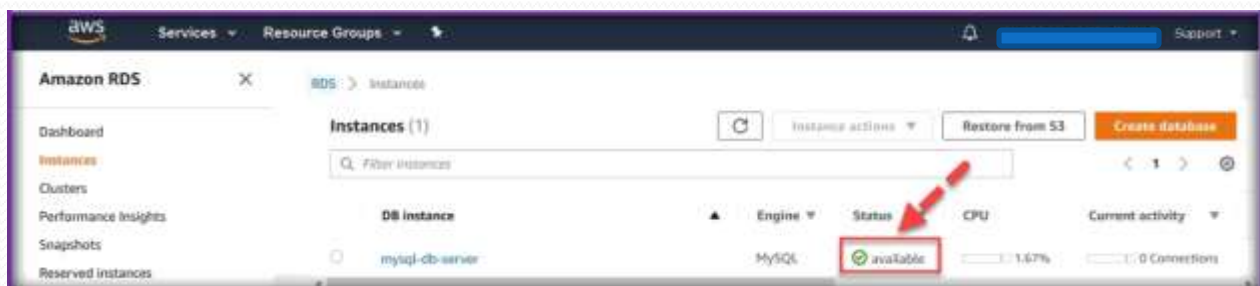
It will display all the configuration details of the DB server.



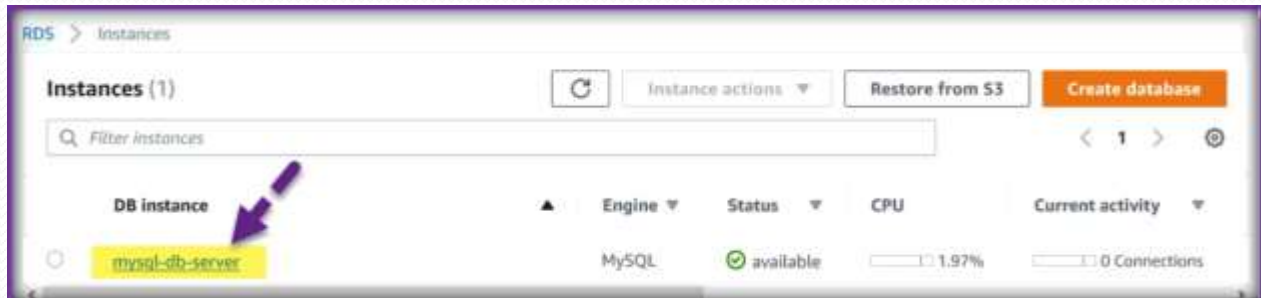
Go to Instances from the left pane to see the instance creation status.



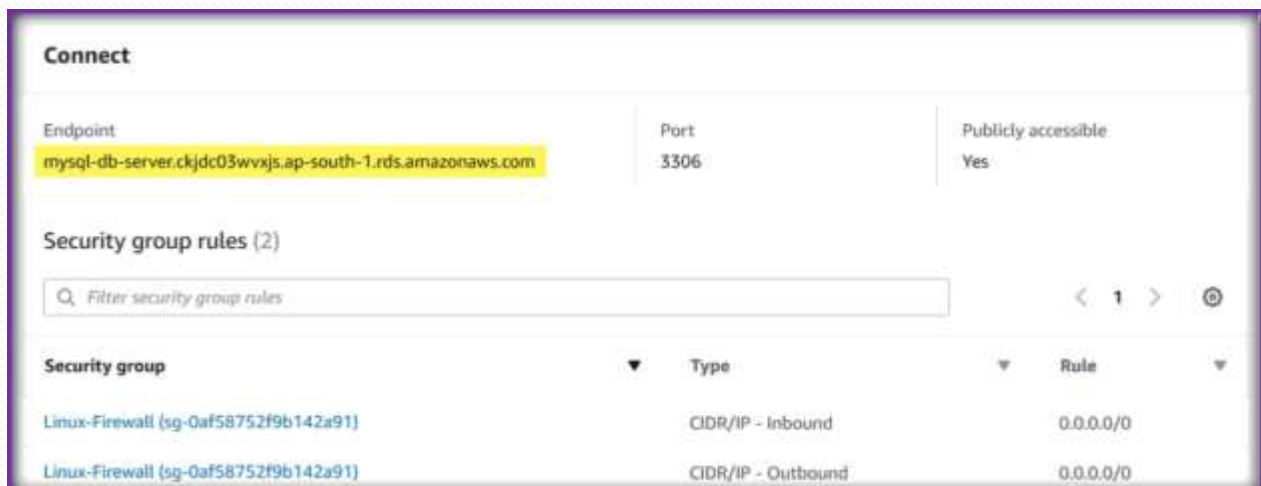
Once RDS creation completed, the status will show as available.



Next, click on the database tag name to go in to the db server details.



Summary page will open, on that page come to Connect section to see the connection details to connect to the database server.

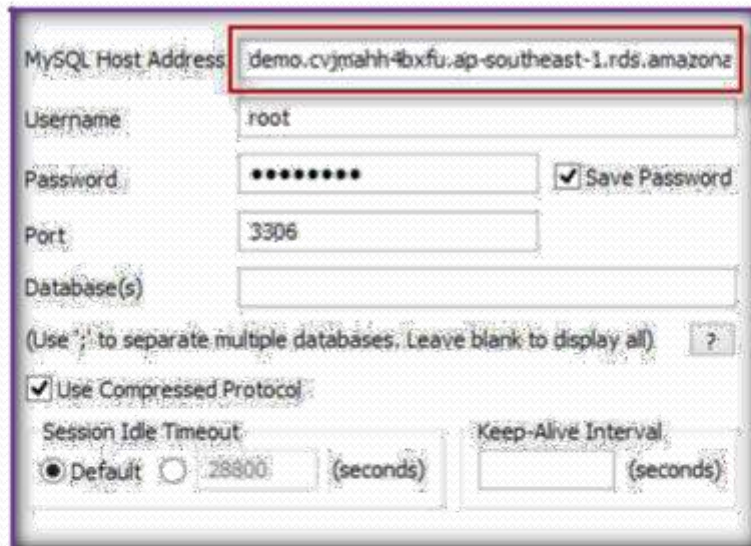


Copy the Endpoint of the db server, will use this db endpoint to connect to this db server.

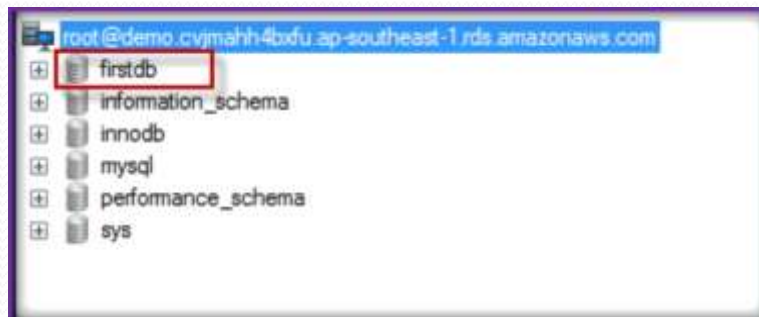
CONNECT TO RDS INSTANCE USING CLIENTS

Use MySQL clients to connect to RDS instances.

Specify End Point as Host or IP Address, Port 3006, and specify user and password combination to connect to RDS DB instance.



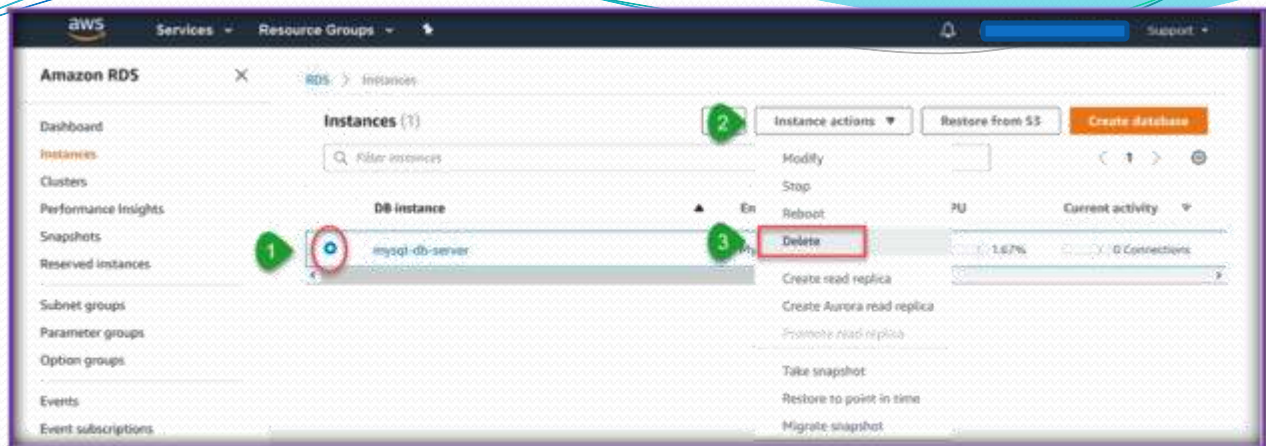
Once connected you can be able see the databases available on RDS.



Make sure to open DB instance port to connect in RDS security Group, otherwise you will not be able to connect.

TERMINATE RDS DB INSTANCE

Once you are in RDS page, click in Instances tab.
Then select instance, go to Instance Actions tab, choose Delete to RDS DB Instance.



Next, no for create final snapshot, acknowledge it, then enter “delete me”, next click on Delete.

