

Guided Lab: Creating an Amazon RDS database

Description

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 AWS Cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

Objectives

In this lab, you will:

- Learn how to create a MySQL database using Amazon RDS
- Configure security groups to manage access to an RDS DB instance
- Connect to an RDS DB instance using MySQL Workbench

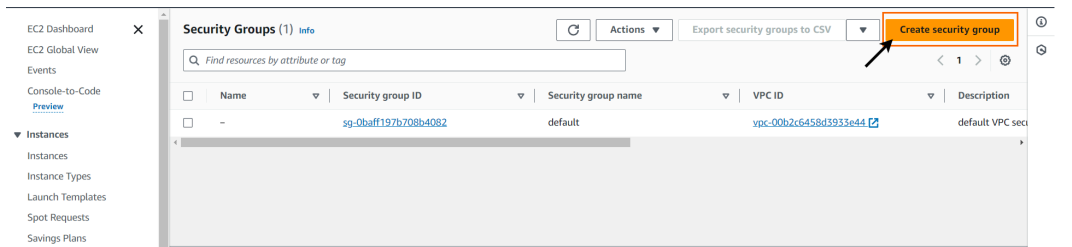
Subscribe to access AWS

PlayCloud Labs

Lab Steps

Creating security groups to manage access to the RDS DB instance

1. Go to EC2 → Network & Security → Security Groups, then create a security group with the following configurations.



- Security group name: **db-access**
- Description: **Allows MYSQL access to developers**
- VPC: **Choose the default VPC**

EC2 > Security Groups > Create security group

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a security group, you must specify a name, description, and VPC.

Basic details

Security group name [Info](#)

Name cannot be edited after creation.

Description [Info](#)

VPC [Info](#)

- Add an Inbound rule with the following configurations:
 - Under "Type", Choose "MYSQL/Aurora" from the dropdown menu.
 - Specify the IP addresses that should have access to your RDS DB instance. You can choose "Anywhere" for testing purposes, but it's recommended to restrict access to known IP addresses for security reasons. For this lab, choose My IP.

Inbound rules [Info](#)

Type	Protocol	Port range	Source	Description - optional
MYSQL/Aurora	TCP	3306	Custom	

Add rule

Custom

Anywhere-IPv4

Anywhere-IPv6

My IP

Outbound rules [Info](#)

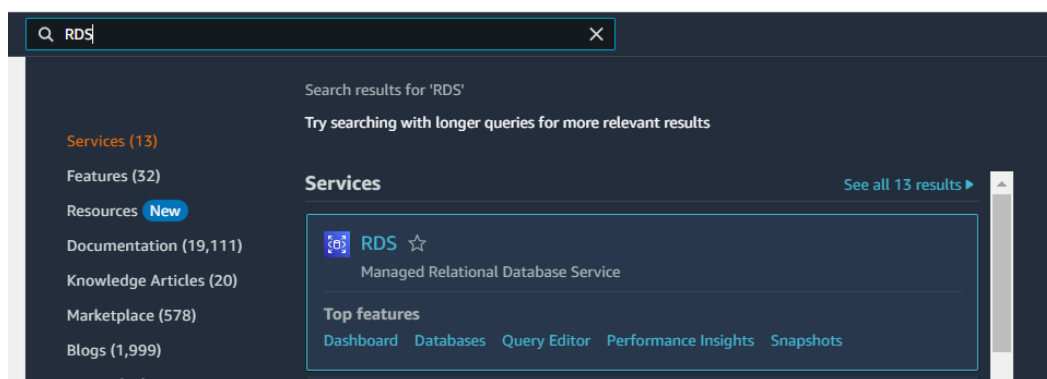
2. Click the "Create security group" button.

To enhance the security of your instance, it is important to only authorize a specific IP address or range of addresses when setting up a rule to access it. Using 0.0.0.0/0 will allow all IPv4 addresses to access your instance. Similarly, using ::/0 will enable all IPv6

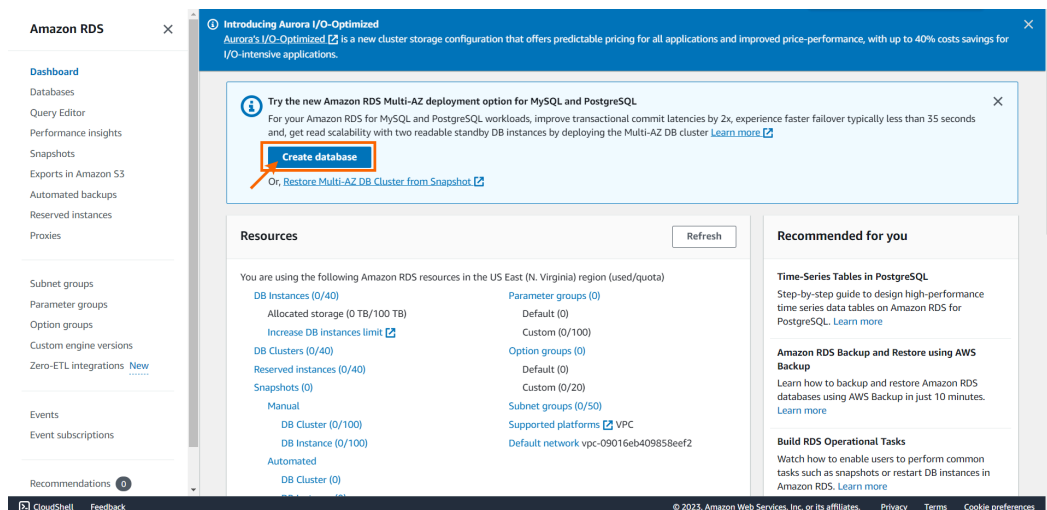
addresses to access your instance. To avoid these two options and provide a more secure solution, it is recommended to specify a particular IP address or range of addresses. You can also reference another security group. For example, instead of using an EC2 instance's IP address, you can specify the security group associated with it.

Create a MySQL database using Amazon RDS

1. In the AWS Management Console, search for "RDS" using the search bar and select the RDS result under Services.



2. Click on the "Create database" button. You will be prompted to configure your database.



3. Choose Database Creation Method. In the "Create database" page, ensure that "Standard Create" is chosen.

Create database

Choose a database creation method [Info](#)

☒ Standard create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☐ Easy create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

4. Choose Database Engine: In the “Engine options” section, choose either “MySQL” as your database engine.

Create database

Choose a database creation method [Info](#)

☒ Standard create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☐ Easy create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

☐ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☒ MySQL



☐ MariaDB



☐ PostgreSQL



☐ Oracle

ORACLE®

☐ Microsoft SQL Server

5. Choose DB Instance Size: In the “Templates” section, choose “Free tier”. This will automatically select the “db.t2.micro” or “db.t3.micro” instance type.

Templates

Choose a sample template to meet your use case.

☐ **Production**

Use defaults for high availability and fast, consistent performance.

☐ **Dev/Test**

This instance is intended for development use outside of a production environment.

☒ **Free tier**

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

6. Configure DB Instance:

- **DB instance identifier:** Give your DB instance a unique name.
- **Master username & password:** Set the master username and password for your database.

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. The first character must be a letter.

☐ **Manage master credentials in AWS Secrets Manager**

Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

[i](#) If you manage the master user credentials in Secrets Manager, some RDS features aren't supported.

[Learn more](#) [↗](#)

☐ **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 master password [Info](#)



7. Set the Allocated storage to 20 GiB only.

Storage

Storage type [Info](#)

General Purpose SSD (gp2)

Baseline performance determined by volume size

Allocated storage [Info](#)

20

GiB

The minimum value is 20 GiB and the maximum value is 6,144 GiB

8. Under Connectivity, enable the Public Access option.

Connectivity [Info](#)



Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.



Don't connect to an EC2 compute resource

Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.



Connect to an EC2 compute resource

Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) [Info](#)

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

vpc-00b2c6458d3933e44

3 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.



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

DB subnet group [Info](#)

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

default-vpc-00b2c6458d3933e44

3 Subnets, 3 Availability Zones

Public access [Info](#)

☒ Yes

RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

☐ No

RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

9. Under Connectivity, choose the created security group.

Existing VPC security groups

Choose one or more options

default



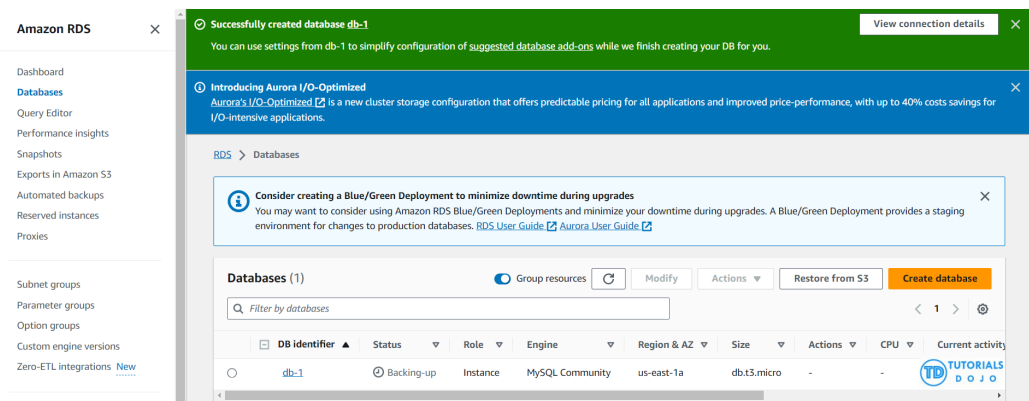
db-access



10. Expand the other sections, and take a moment to review the available configurations. It is not necessary to make any changes to the remaining configurations. You can leave them as they are set by default.

11. Click on the "Create database" button at the bottom of the page to create your MySQL database.

12. After clicking the 'Create database' button, a confirmation will appear to let you know that the process has started and the database has been created.

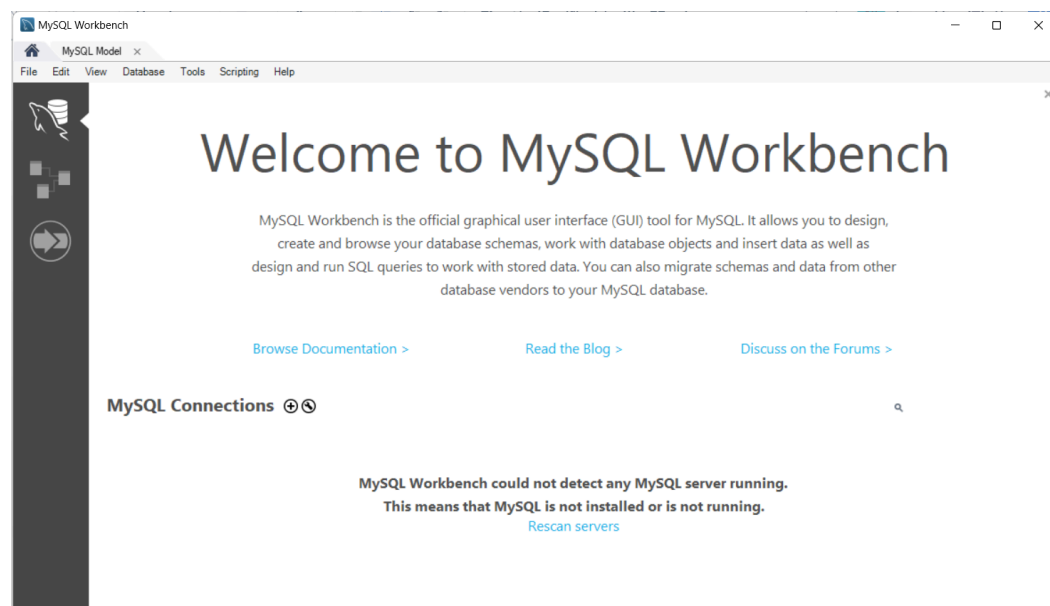


Connect to the RDS DB instance using MySQL Workbench

1. Once the database is ready (this might take a few minutes). you can connect to it using any MySQL client. In this lab, we'll be using MySQL Workbench.

Download and install this on your computer: MySQL: Download MySQL Workbench

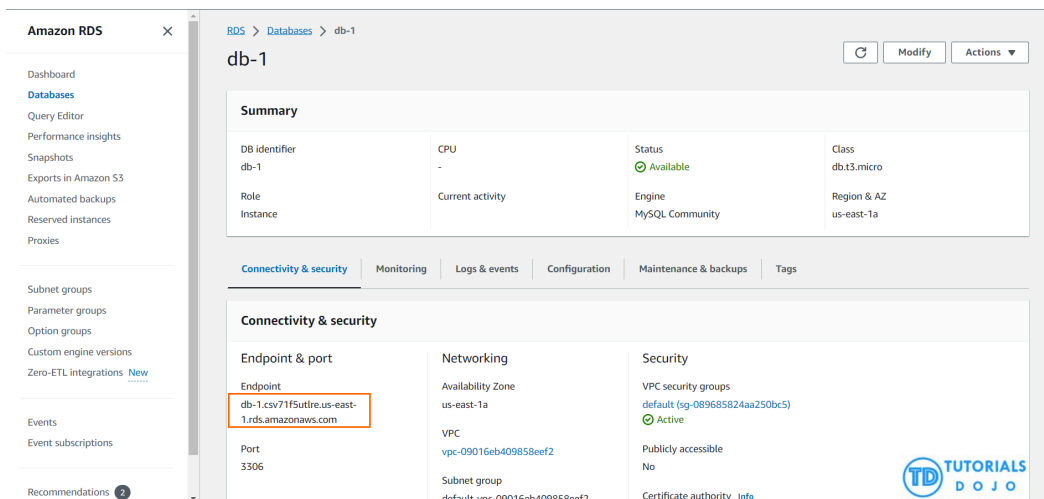
2. Start the MySQL Workbench application on your computer.



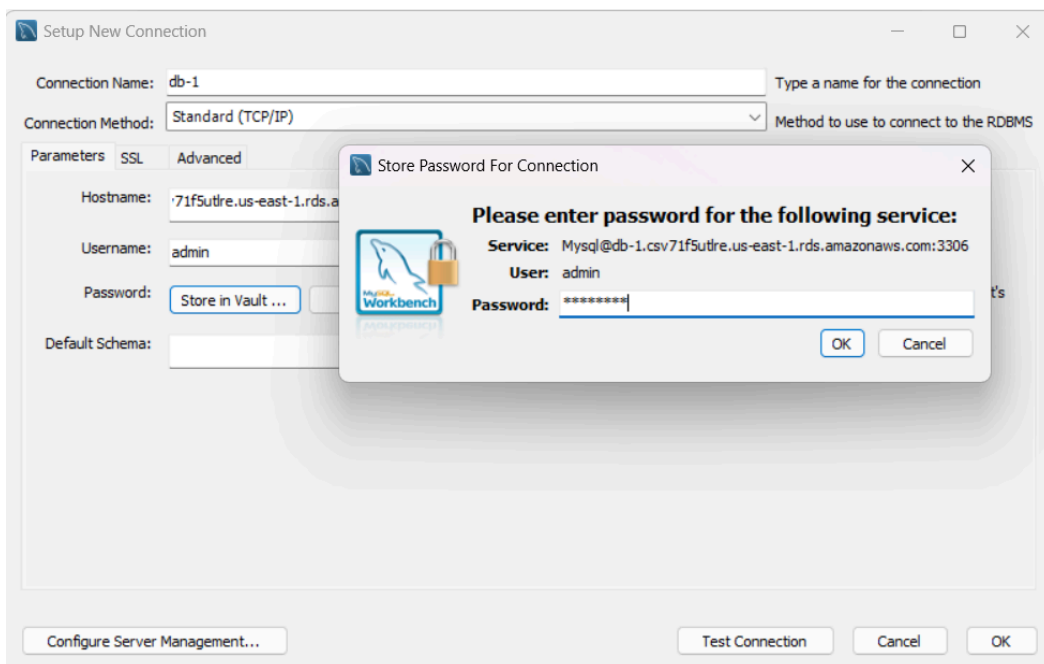
3. Click on the "+" icon next to "MySQL Connections" to create a new connection.

4. Configure Connection Details:

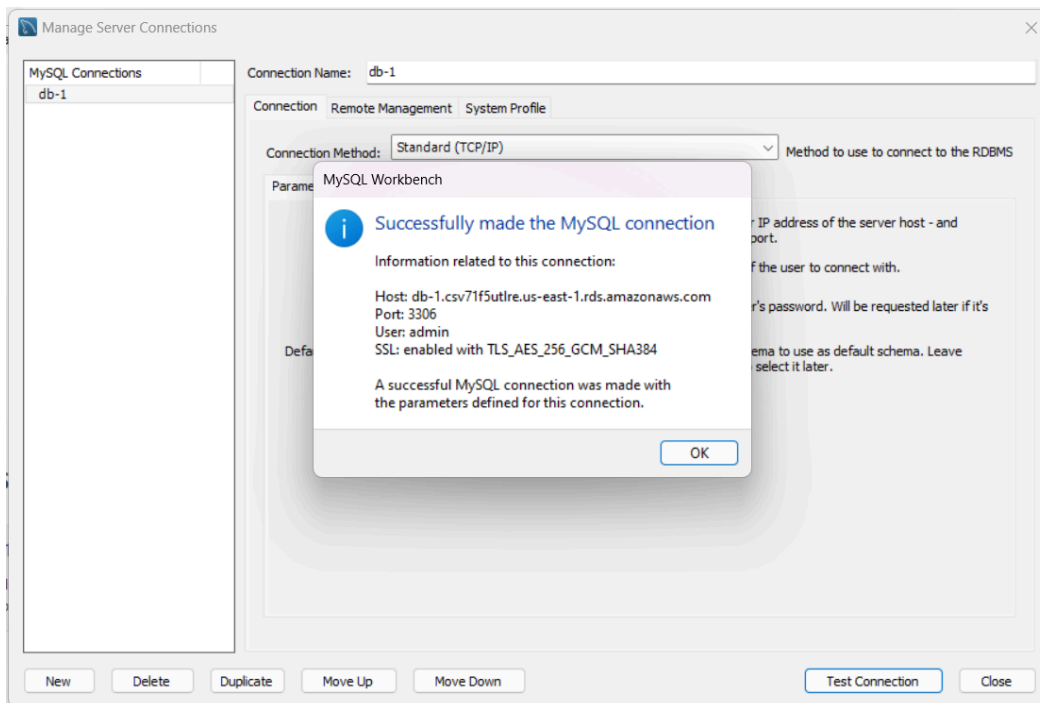
- **Connection Name:** Give your connection a unique name.
- **Hostname:** Use the endpoint provided in the RDS console.
- **Port:** Use the port provided in the RDS console.



- **Username & Password:** Use the master username and password you set earlier to establish the connection.



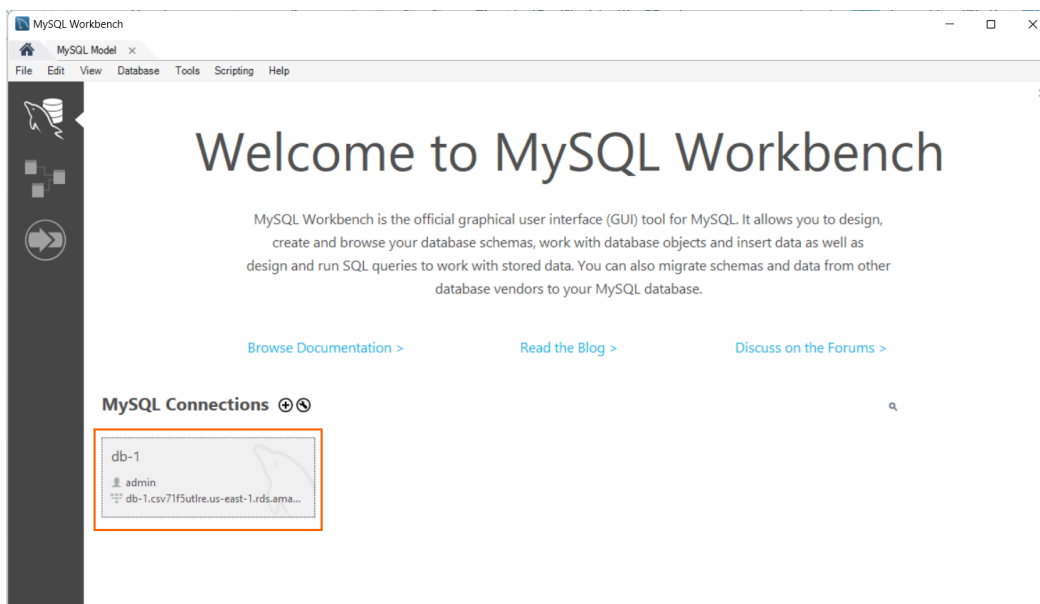
5. Click "OK", then test the connection by clicking the "Test Connection" button.



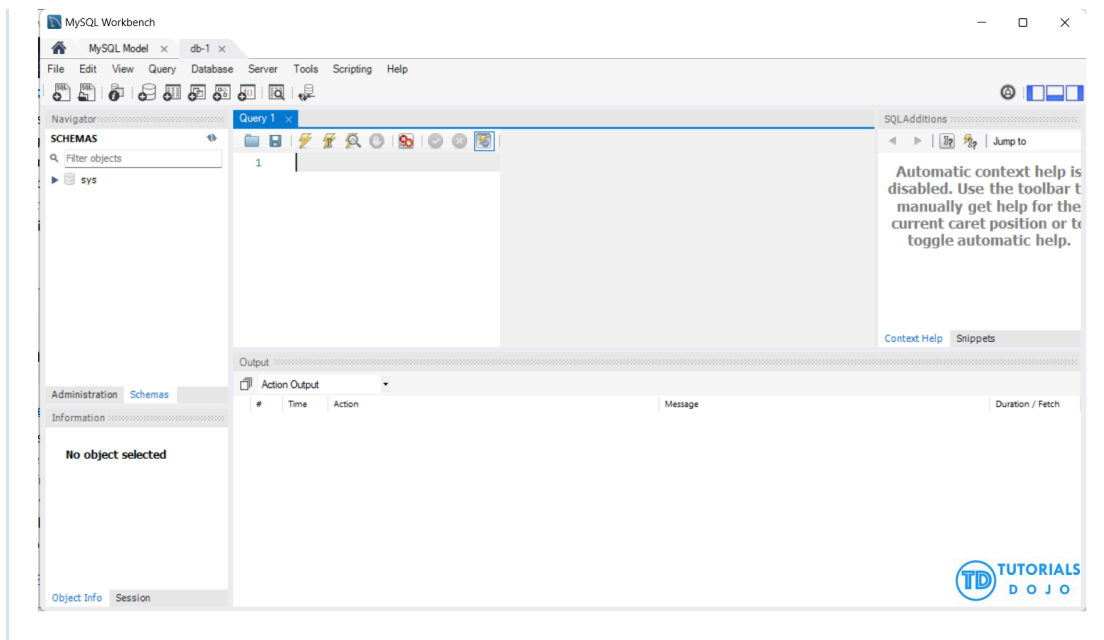
6. If the test connection is successful, you can click the **"OK"** button to save the connection settings.

7. In the MySQL Workbench home screen, you should now see your newly created connection under **"MySQL Connections."**

8. Double-click on your connection to establish a connection to the RDS DB instance.



9. Once connected, you can browse and manage your RDS database using the MySQL Workbench interface.

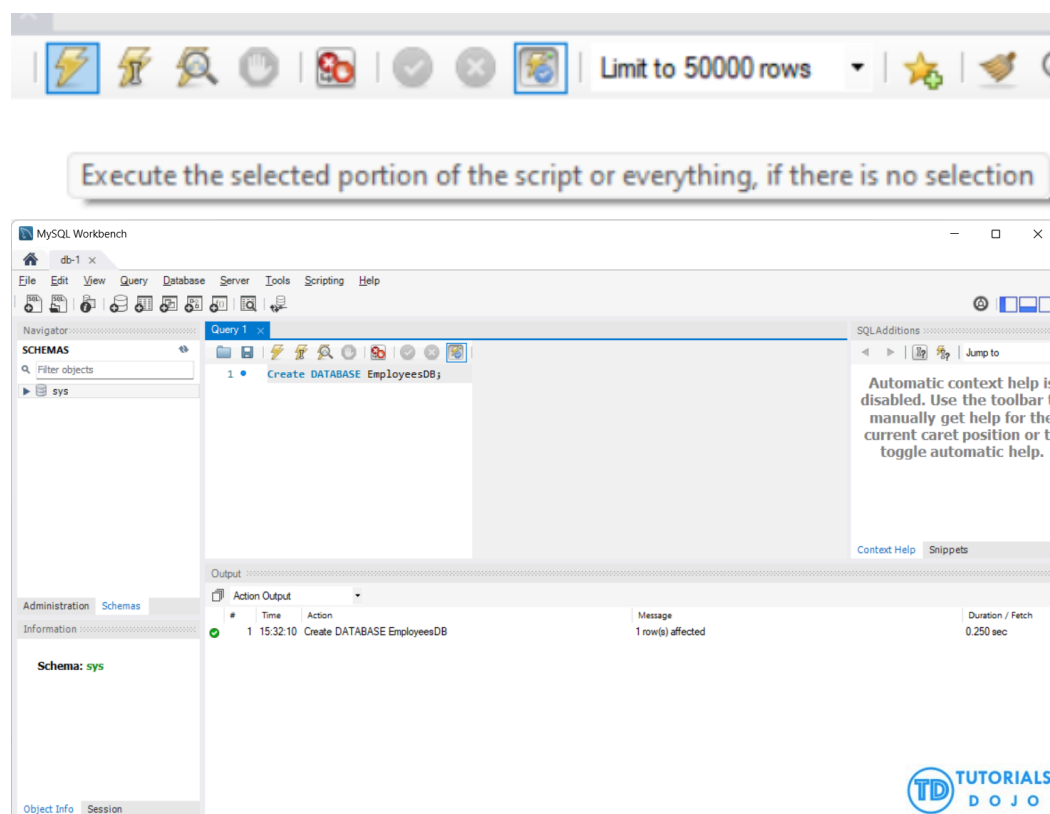


Run SQL Commands

1. Creating a Database.

```
Create DATABASE EmployeesDB;
```

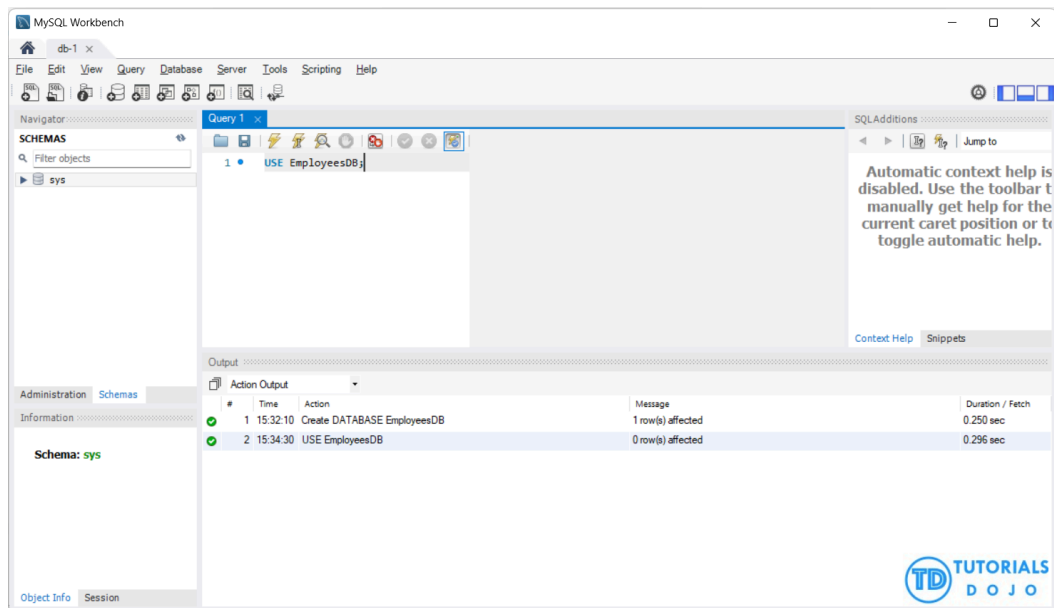
Run the command by clicking the **Lightning** icon.



2. Choose the database that has been created.

```
USE EmployeesDB;
```

Run the command by clicking the **Lightning** icon.



3. Creating a table.

```
CREATE TABLE Employees ( ID INT PRIMARY KEY, Name  
VARCHAR(50), Age INT, Salary DECIMAL(10, 2) );
```

Run the command by clicking the **Lightning** icon.

This command creates a table called `Employees` with four columns: ID, Name, Age, and Salary. The ID column is set as the primary key.

4. Writing to the Table

```
INSERT INTO Employees (ID, Name, Age, Salary)
VALUES (1, 'Jose Rizal', 25, 50000.00);
```

```
INSERT INTO Employees (ID, Name, Age, Salary)
VALUES (2, 'Andres Bonifacio', 25, 50000.00);
```

```
INSERT INTO Employees (ID, Name, Age, Salary)
VALUES (3, 'Emilio Aguinaldo', 25, 50000.00);
```

Run the command by clicking the **Lightning** icon.

This command inserts a new row into the `Employees` table. Each row represents an employee with their respective `ID` , `Name` , `Age` , and `Salary` .

5. Reading from the Table

```
SELECT * FROM Employees;
```

Run the command by clicking the **Lightning** icon.

This command selects all rows and columns from the `Employees` table.

Congratulations! You just learned how to create a MySQL database using Amazon RDS, configure security groups to manage access to an RDS DB instance, and connect to an RDS DB instance using MySQL Workbench.