MARIADB ASSIGNMENT

Name: Vikram

Problem Statement:

You work for **XYZ Corporation**. Their application requires a SQL service that can store data which can be retrieved if required. Implement a suitable RDS engine for the same.

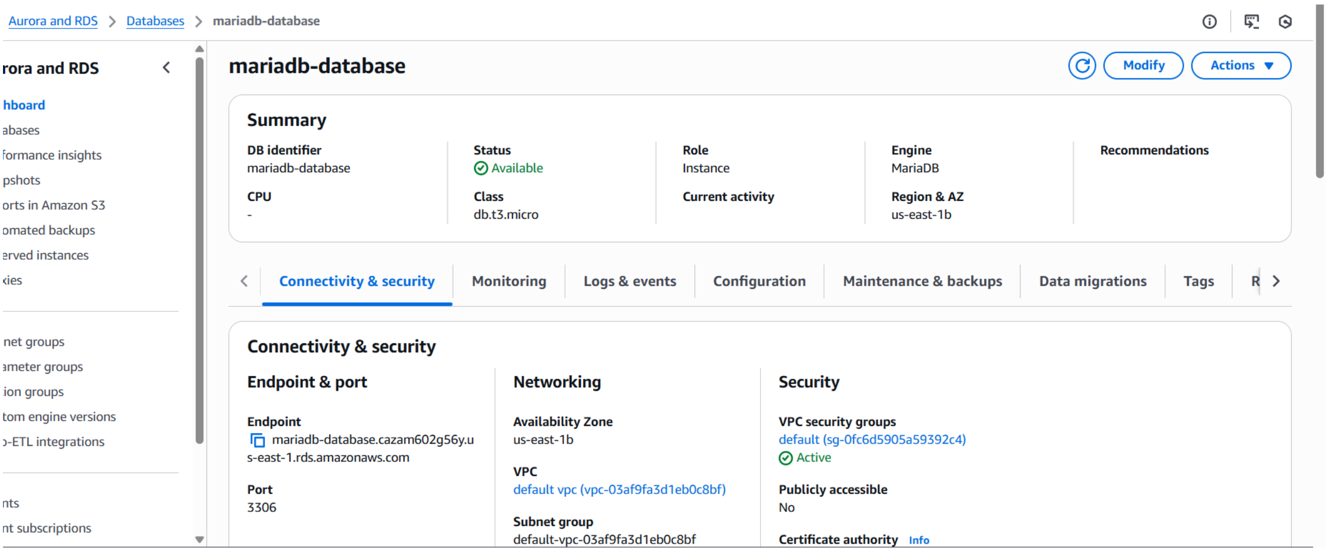
While migrating, you are asked to perform the following tasks:

1. Create a **MariaDB Engine based RDS Database**.
2. Connect to the DB using the following ways:
   a. SQL Client for Windows
   b. Linux based EC2 Instance

Solution Implementation Steps

1. Create MariaDB Engine Based RDS Database

1. Logged in to the **AWS Management Console** → Opened **RDS Service**.
2. Clicked on **Create Database** → Selected **Standard Create**.
3. Under **Engine options**, selected **MariaDB**.
4. Configured the following settings:
   - **DB instance identifier:** mariadb-database
   - **Master username:** admin
   - **Master password:** *(secure password)*
   - **Instance type:** db.t3.micro (Free-tier eligible)
5. Kept **Public accessibility** as **yes**
6. Created or selected a **VPC** and **DB Subnet Group**.
7. Selected a default **VPC Security Group** allowing inbound port **3306** (MySQL/Aurora) from both:
   - The **EC2 instance's Security Group** (for private access)
   - The **local system IP** (for Windows SQL Client access).
8. Clicked **Create database** and waited until the status changed to **Available**.
9. Noted down the **RDS Endpoint** and **Port (3306)**.



mariadb-database

Summary

DB identifier	Status	Role	Engine	Recommendations
mariadb-database	✓ Available	Instance	MariaDB	
CPU	Class	Current activity	Region & AZ	
-	db.t3.micro		us-east-1b	

Connectivity & security

Endpoint & port

Endpoint
mariadb-database.cazam602g56y.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1b

VPC
default vpc (vpc-03af9fa3d1eb0c8bf)

Subnet group
default-vpc-03af9fa3d1eb0c8bf

Security

VPC security groups
default (sg-0fc6d5905a59392c4)
✓ Active

Publicly accessible
No

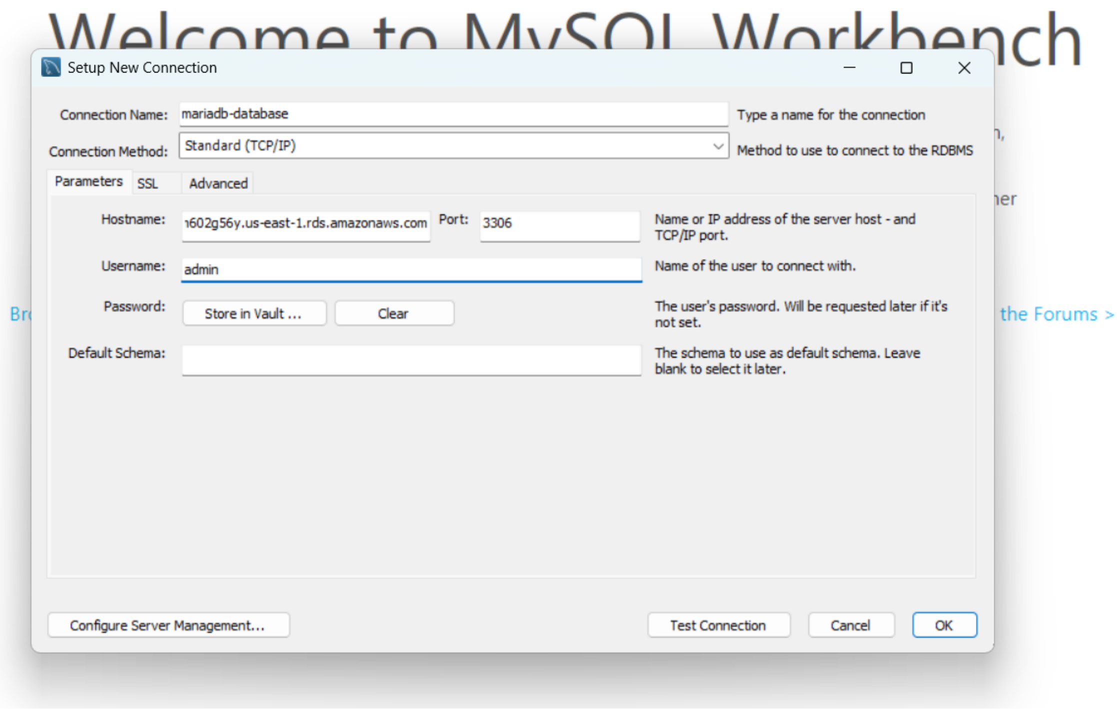
Certificate authority Info

2. Connect to the RDS Database Using SQL Client for Windows

1. Installed **MySQL Workbench** on Windows.
2. Opened Workbench → **Database** → **Manage Connections** → **New**.
3. Entered the connection details:
   - **Connection Name:** mariadb-database
   - **Hostname:** <your-rds-endpoint> (e.g.,mariadb-database.abc123xyz.us-east-1.rds.amazonaws.com)
   - **Port:** 3306
   - **Username:** admin
   - **Password:** entered and saved securely.
4. Clicked **Test Connection** → Received **"Successfully made the MySQL connection"** message.
5. Clicked **OK** to save the connection and opened it.
6. Verified by running sample SQL commands:

```
SHOW DATABASES;
CREATE DATABASE winclientdb;
USE winclientdb;
CREATE TABLE employee (id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50));
INSERT INTO employee(name) VALUES ('Vikram'), ('Soumya');
SELECT * FROM employee;
```

☑ Data successfully created and retrieved.



Welcome to MySQL Workbench

Setup New Connection

Connection Name: mariadb-database — Type a name for the connection

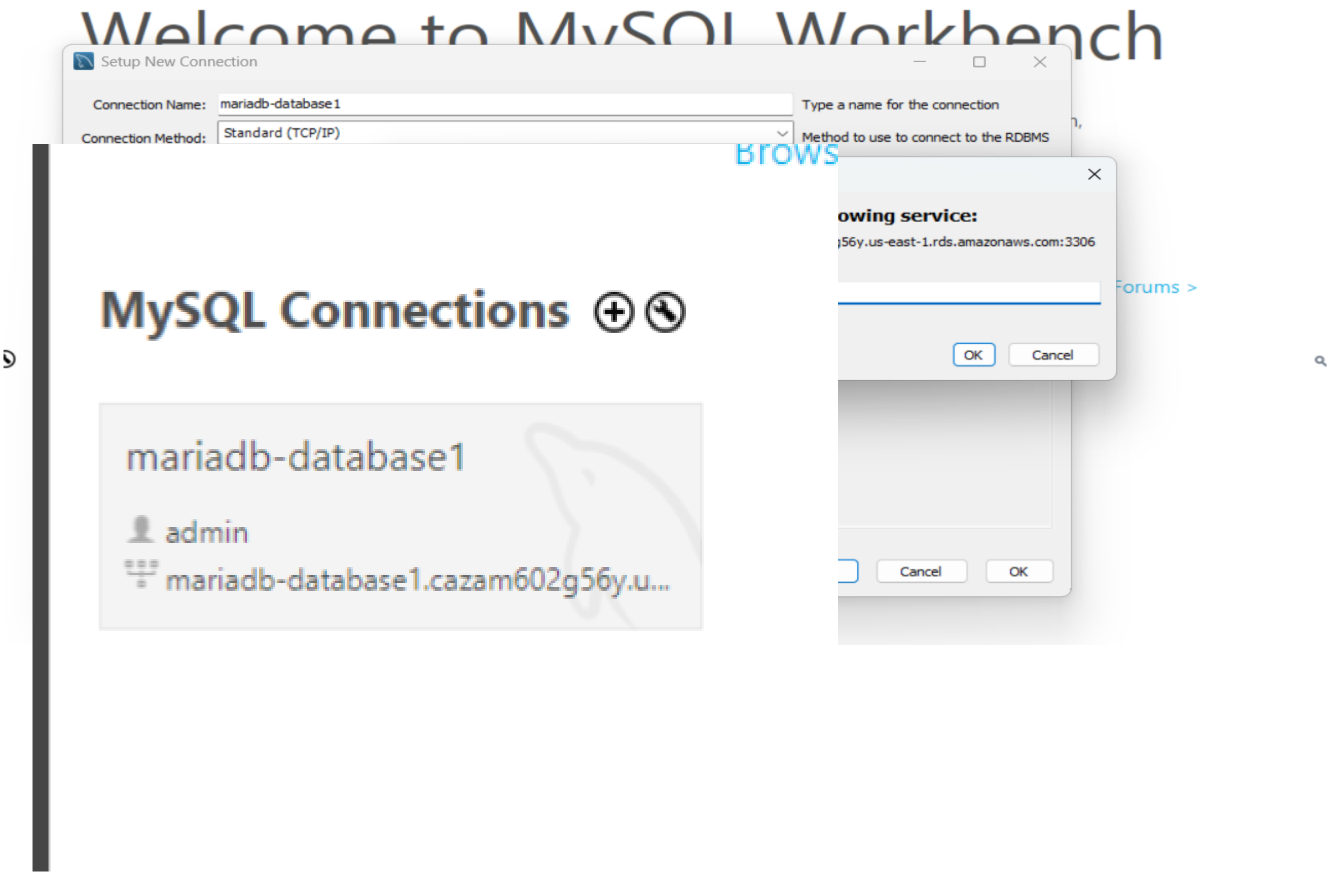
Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

Hostname: ı602g56y.us-east-1.rds.amazonaws.com Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: admin — Name of the user to connect with.

Password: Store in Vault ... Clear — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.



Welcome to MySQL Workbench

Setup New Connection

Connection Name: mariadb-database1 — Type a name for the connection

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

owing service:

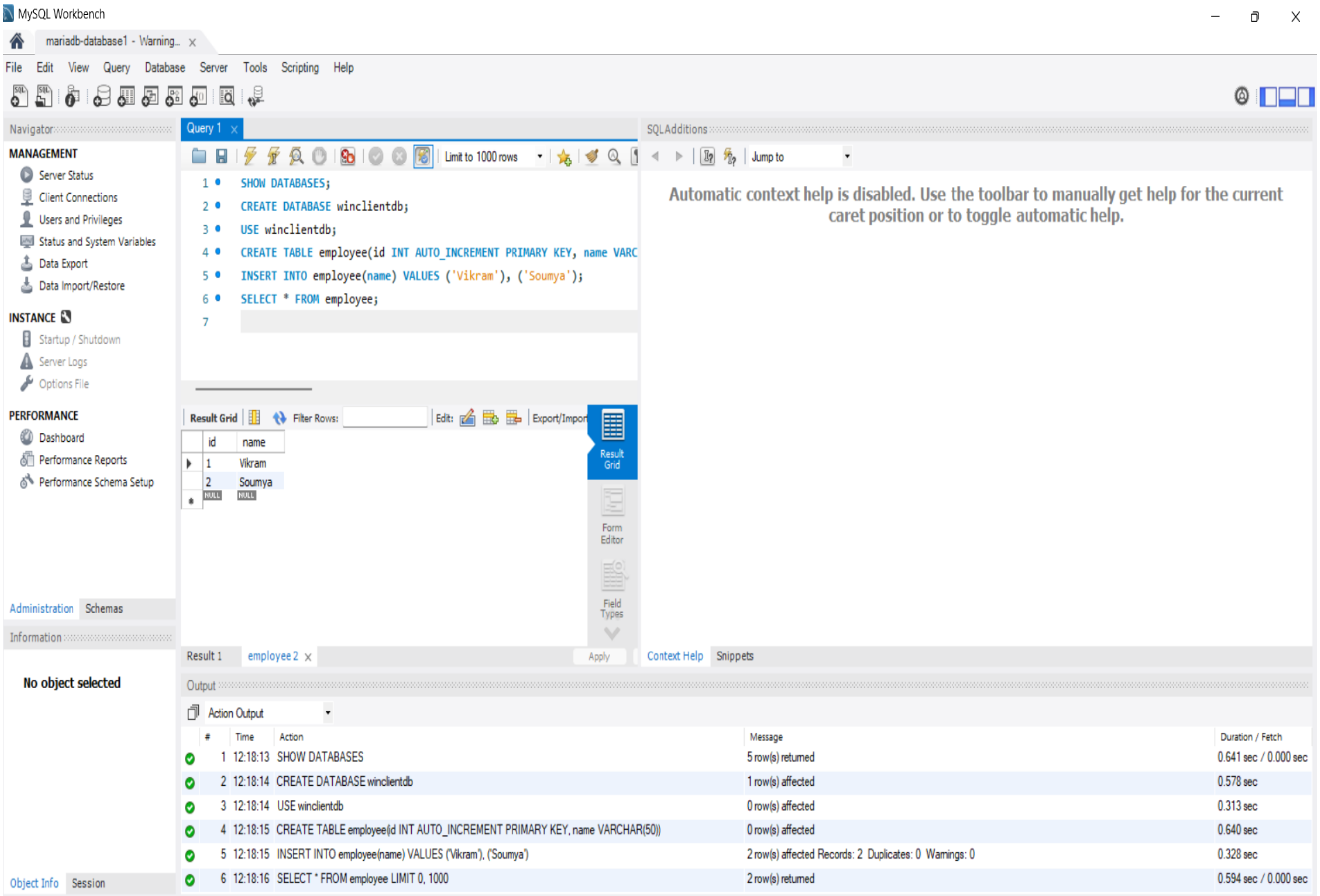
ı56y.us-east-1.rds.amazonaws.com:3306

MySQL Connections ⊕ 🔧

mariadb-database1

admin

mariadb-database1.cazam602g56y.u...



```
SHOW DATABASES;
CREATE DATABASE winclientdb;
USE winclientdb;
CREATE TABLE employee(id INT AUTO_INCREMENT PRIMARY KEY, name VARC
INSERT INTO employee(name) VALUES ('Vikram'), ('Soumya');
SELECT * FROM employee;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

id	name
1	Vikram
2	Soumya
NULL	NULL

No object selected

#	Time	Action	Message	Duration / Fetch
1	12:18:13	SHOW DATABASES	5 row(s) returned	0.641 sec / 0.000 sec
2	12:18:14	CREATE DATABASE winclientdb	1 row(s) affected	0.578 sec
3	12:18:14	USE winclientdb	0 row(s) affected	0.313 sec
4	12:18:15	CREATE TABLE employee(id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50))	0 row(s) affected	0.640 sec
5	12:18:15	INSERT INTO employee(name) VALUES ('Vikram'), ('Soumya')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.328 sec
6	12:18:16	SELECT * FROM employee LIMIT 0, 1000	2 row(s) returned	0.594 sec / 0.000 sec

3. Connect to the RDS Database Using Linux EC2 Instance

1. Launched an **EC2 Instance (Amazon Linux 2023)** in the **same VPC** as the RDS instance.
2. Configured Security Groups:
    - **EC2 SG:** Allowed **SSH (port 22)** from **ipv4** anywhere
    - **RDS SG:** Allowed inbound **MySQL/Aurora (port 3306)** from **EC2 SG**.
3. Connected to EC2 using EC2 instance connect:
4. Installed the MySQL client (MariaDB package):

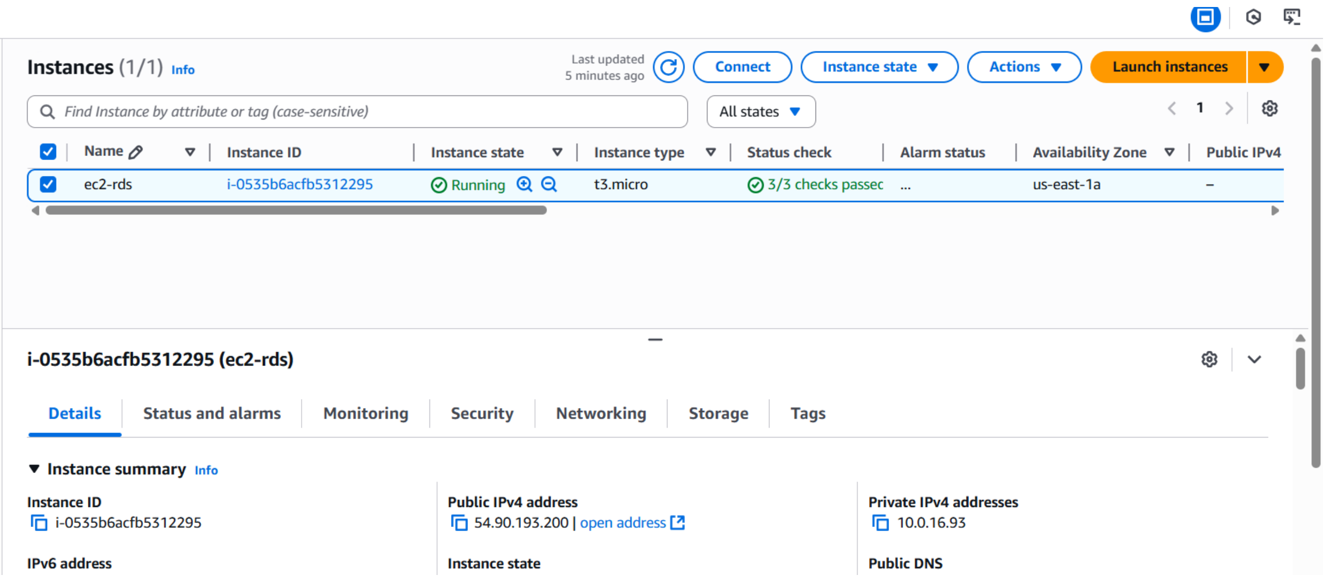
```
sudo dnf install -y mariadb105
```

☑ Installed successfully.

5. Connected to RDS from EC2:
6. mysql -h <your-rds-endpoint> -P 3306 -u admin -p
7. Verified connection and executed test queries:

```
CREATE DATABASE testdb;
USE testdb;
CREATE TABLE demo (id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50));
INSERT INTO demo (name) VALUES ('Vikram'), ('Soumya');
SELECT * FROM demo;
```

☑ Data displayed correctly, confirming successful EC2-to-RDS connectivity.



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
☑	ec2-rds	i-0535b6acfb5312295	✓ Running	t3.micro	✓ 3/3 checks passec	...	us-east-1a	–

i-0535b6acfb5312295 (ec2-rds)

▼ Instance summary Info

Instance ID
i-0535b6acfb5312295

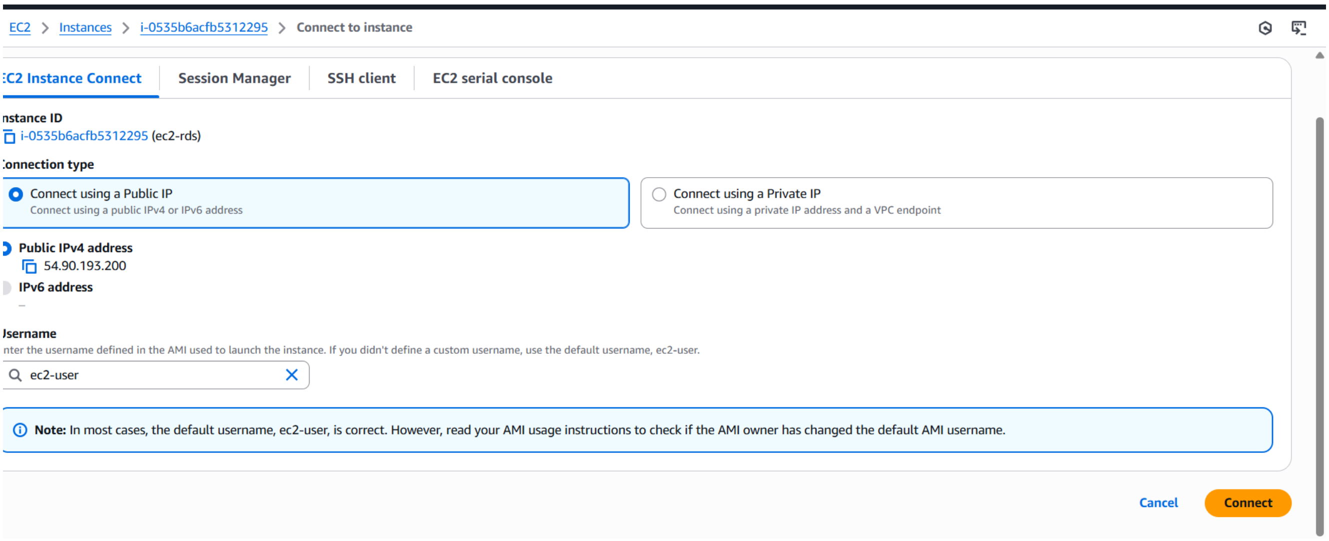
Public IPv4 address
54.90.193.200 | open address

Private IPv4 addresses
10.0.16.93

IPv6 address

Instance state

Public DNS



nstance ID
i-0535b6acfb5312295 (ec2-rds)

:onnection type

● Connect using a Public IP
Connect using a public IPv4 or IPv6 address

○ Connect using a Private IP
Connect using a private IP address and a VPC endpoint

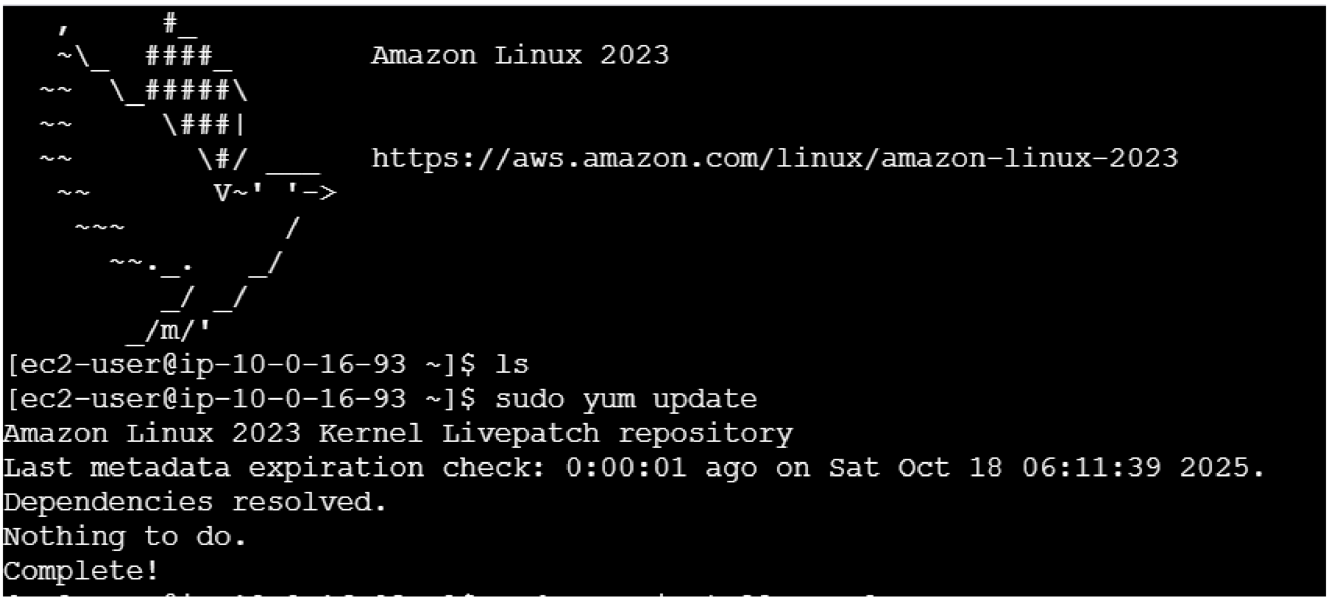
Public IPv4 address
54.90.193.200

IPv6 address
–

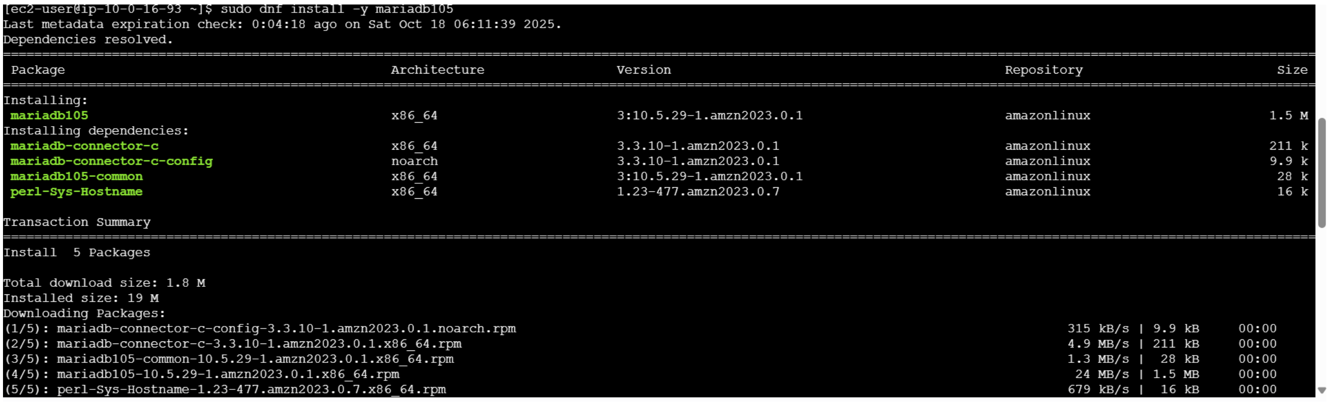
Jsername
nter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.



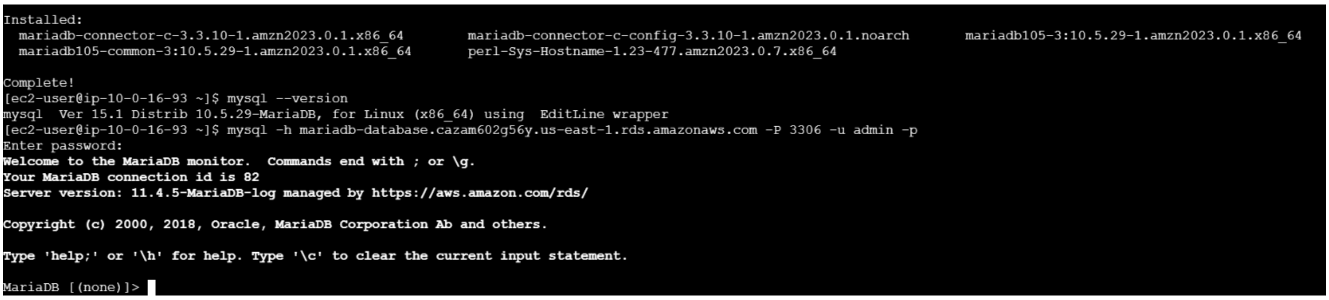
```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-16-93 ~]$ ls
[ec2-user@ip-10-0-16-93 ~]$ sudo yum update
Amazon Linux 2023 Kernel Livepatch repository
Last metadata expiration check: 0:00:01 ago on Sat Oct 18 06:11:39 2025.
Dependencies resolved.
Nothing to do.
Complete!
```



```
[ec2-user@ip-10-0-16-93 ~]$ sudo dnf install -y mariadb105
Last metadata expiration check: 0:04:18 ago on Sat Oct 18 06:11:39 2025.
Dependencies resolved.
================================================================================================================================================================
 Package                                         Architecture                 Version                                          Repository                  Size
================================================================================================================================================================
Installing:
 mariadb105                                      x86_64                       3:10.5.29-1.amzn2023.0.1                         amazonlinux                1.5 M
Installing dependencies:
 mariadb-connector-c                             x86_64                       3.3.10-1.amzn2023.0.1                            amazonlinux                211 k
 mariadb-connector-c-config                      noarch                       3.3.10-1.amzn2023.0.1                            amazonlinux                9.9 k
 mariadb105-common                               x86_64                       3:10.5.29-1.amzn2023.0.1                         amazonlinux                 28 k
 perl-Sys-Hostname                               x86_64                       1.23-477.amzn2023.0.7                            amazonlinux                 16 k

Transaction Summary
================================================================================================================================================================
Install  5 Packages

Total download size: 1.8 M
Installed size: 19 M
Downloading Packages:
(1/5): mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch.rpm                                                              315 kB/s | 9.9 kB     00:00
(2/5): mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64.rpm                                                                     4.9 MB/s | 211 kB     00:00
(3/5): mariadb105-common-10.5.29-1.amzn2023.0.1.x86_64.rpm                                                                      1.3 MB/s |  28 kB     00:00
(4/5): mariadb105-10.5.29-1.amzn2023.0.1.x86_64.rpm                                                                              24 MB/s | 1.5 MB     00:00
(5/5): perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64.rpm                                                                       679 kB/s |  16 kB     00:00
```



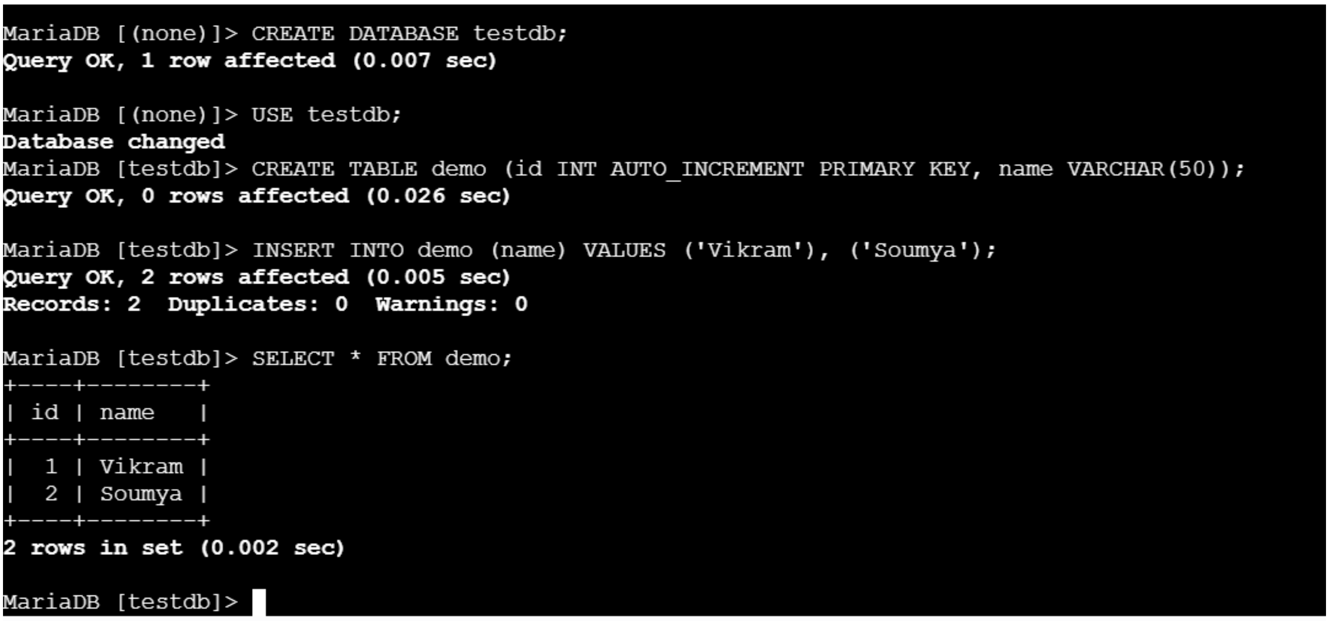
```
Installed:
  mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64        mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch        mariadb105-3:10.5.29-1.amzn2023.0.1.x86_64
  mariadb105-common-3:10.5.29-1.amzn2023.0.1.x86_64       perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64

Complete!
[ec2-user@ip-10-0-16-93 ~]$ mysql --version
mysql  Ver 15.1 Distrib 10.5.29-MariaDB, for Linux (x86_64) using  EditLine wrapper
[ec2-user@ip-10-0-16-93 ~]$ mysql -h mariadb-database.cazam602g56y.us-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 82
Server version: 11.4.5-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```



```
MariaDB [(none)]> CREATE DATABASE testdb;
Query OK, 1 row affected (0.007 sec)

MariaDB [(none)]> USE testdb;
Database changed
MariaDB [testdb]> CREATE TABLE demo (id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50));
Query OK, 0 rows affected (0.026 sec)

MariaDB [testdb]> INSERT INTO demo (name) VALUES ('Vikram'), ('Soumya');
Query OK, 2 rows affected (0.005 sec)
Records: 2  Duplicates: 0  Warnings: 0

MariaDB [testdb]> SELECT * FROM demo;
+----+--------+
| id | name   |
+----+--------+
|  1 | Vikram |
|  2 | Soumya |
+----+--------+
2 rows in set (0.002 sec)

MariaDB [testdb]>
```

4. **Verification Output**

Query Result (Common for Both Connections):

```
+----+--------+

| id | name   |

+----+--------+

|  1 | Vikram |

|  2 | Soumya |

+----+--------+
```

5. Result / Conclusion

- Successfully created a **MariaDB RDS Database** on AWS.
- Verified connectivity from **both SQL Client for Windows (MySQL Workbench)** and **Linux EC2 Instance**.
- Properly configured **VPC Security Groups** to ensure secure access.
- Verified full CRUD operations — confirming the database functions as required.