Cloud Technology (AWS)

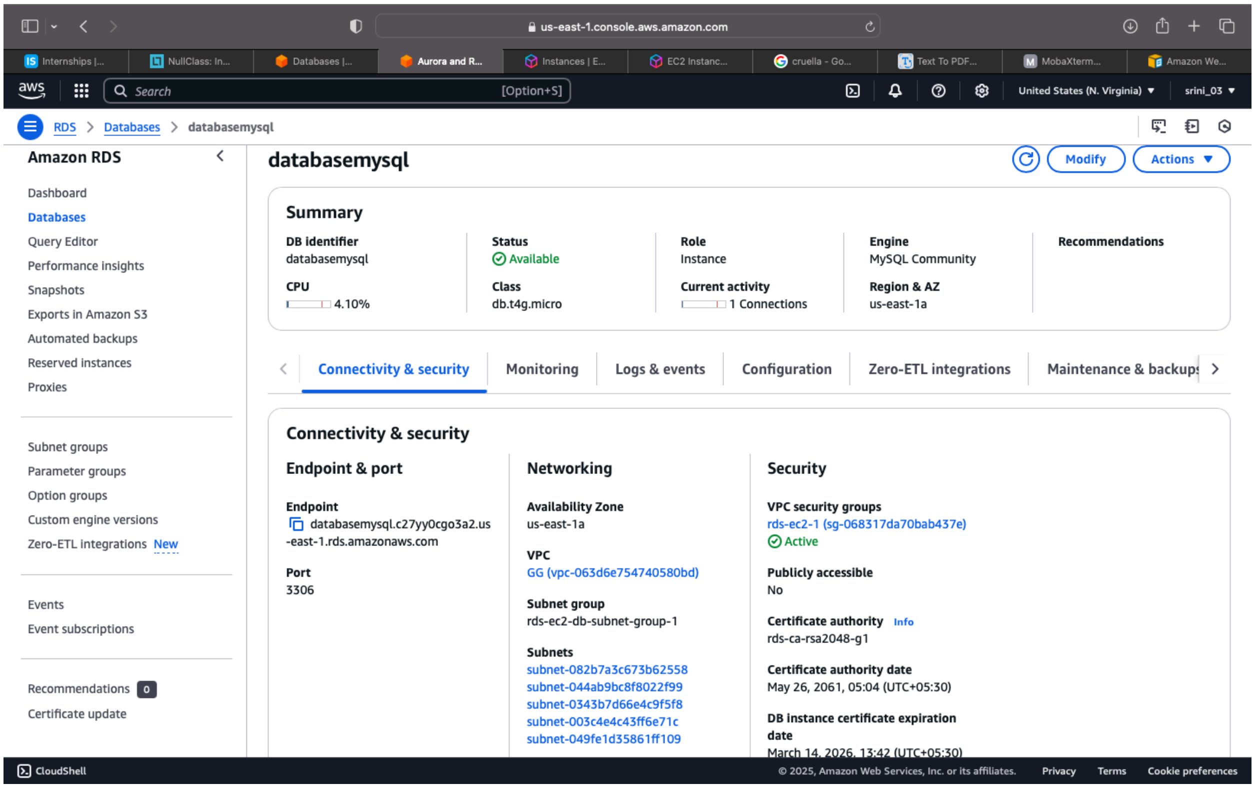
Create an RDS MySQL Database and Connect from EC2 Launch an RDS MySQL instance inside your VPC. Modify the security group to allow connections from your EC2 instance. Connect to the database from your EC2 instance using MySQL CLI.

Setting Up RDS MySQL and Connecting from EC2

This guide will walk you through launching an **RDS MySQL instance** inside your **VPC**, configuring security settings, and connecting to it from an **EC2 instance**.

Step 1: Launch an RDS MySQL Instance

1. **Go to AWS Console** → **RDS** → Click **Create Database**.
2. Select **Standard Create**.
3. **Choose Engine**:
    a. **Engine type**: MySQL
    b. **Version**: Select the latest stable version.
4. **Choose Deployment Option**: Single DB instance.
5. **Settings**:
    a. **DB Instance Identifier** : databasemysql
    b. **Master Username**: databasemysql
    c. **Master Password**: databasemysql
6. **Instance Size**: Select db.t3.micro (free tier eligible).
7. **Storage**: 20GB (default).
8. **VPC and Security Settings**:
    a. **VPC**: Choose your existing VPC or create a new one.
    b. **Subnet Group**: Choose default.
    c. **Public Access**: No (Recommended for security).
    d. **Security Group**: Create a new one or select an existing one.
9. **Database Authentication**: Use **Password authentication**.
10. **Click Create Database** and wait for it to become **"Available"**.



databasemysql

Summary

DB identifier
databasemysql

CPU
4.10%

Status
Available

Class
db.t4g.micro

Role
Instance

Current activity
1 Connections

Engine
MySQL Community

Region & AZ
us-east-1a

Recommendations

Connectivity & security

Endpoint & port

Endpoint
databasemysql.c27yy0cgo3a2.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1a

VPC
GG (vpc-063d6e754740580bd)

Subnet group
rds-ec2-db-subnet-group-1

Subnets
subnet-082b7a3c673b62558
subnet-044ab9bc8f8022f99
subnet-0343b7d66e4c9f5f8
subnet-003c4e4c43ff6e71c
subnet-049fe1d35861ff109

Security

VPC security groups
rds-ec2-1 (sg-068317da70bab437e)
Active

Publicly accessible
No

Certificate authority Info
rds-ca-rsa2048-g1

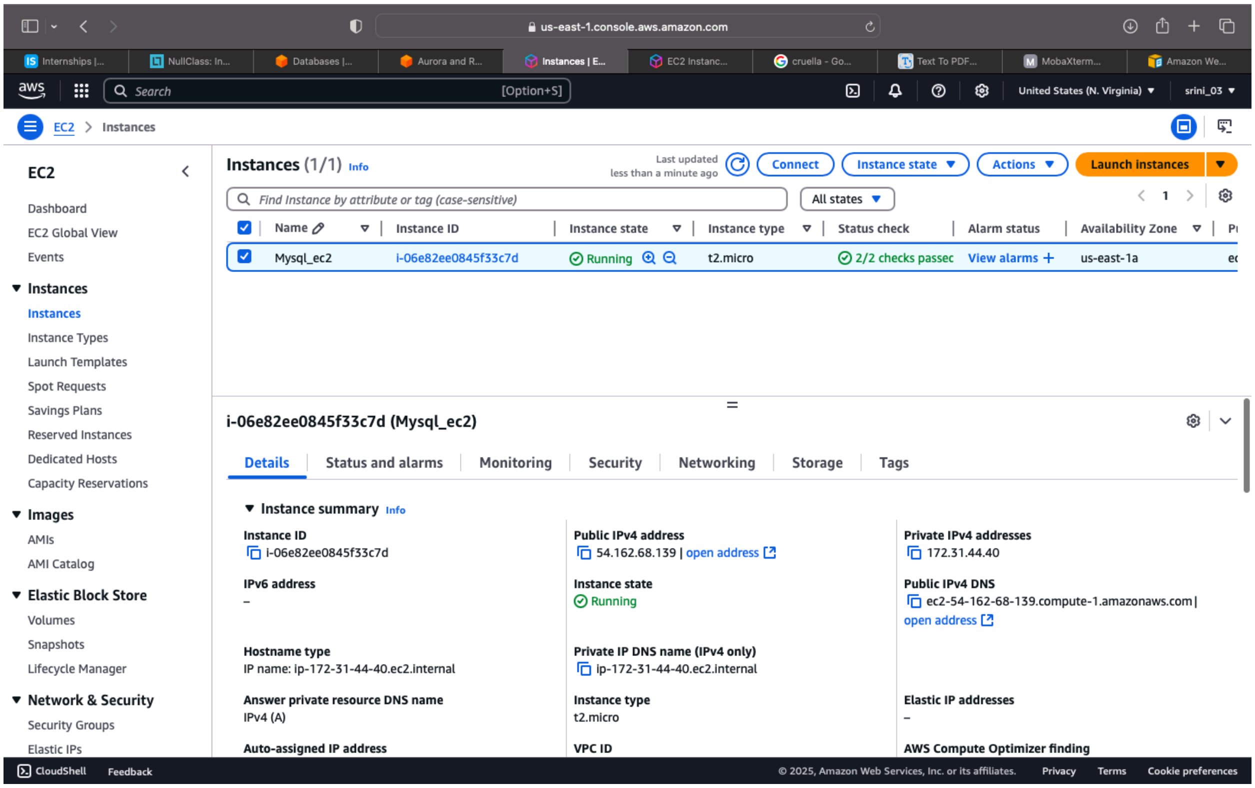
Certificate authority date
May 26, 2061, 05:04 (UTC+05:30)

DB instance certificate expiration date
March 14, 2026, 13:42 (UTC+05:30)

Step 2: Modify Security Group to Allow EC2 Connections

1. **Go to EC2 Dashboard → Security Groups**.
2. Find the security group attached to your **RDS instance**.
3. Click **Inbound Rules → Edit Inbound Rules**.
4. **Add a new rule**:
    a. **Type**: MySQL/Aurora
    b. **Protocol**: TCP
    c. **Port Range**: 3306
    d. **Source**: Select **"Custom"** and enter your **EC2 Security Group ID**.
5. **Save Rules**.

Now, your EC2 instance can connect to RDS MySQL.



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
☑	Mysql_ec2	i-06e82ee0845f33c7d	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	ec

i-06e82ee0845f33c7d (Mysql_ec2)

Instance summary Info

Instance ID
i-06e82ee0845f33c7d

Public IPv4 address
54.162.68.139 | open address

Private IPv4 addresses
172.31.44.40

IPv6 address
–

Instance state
Running

Public IPv4 DNS
ec2-54-162-68-139.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-44-40.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-44-40.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

Auto-assigned IP address

VPC ID

AWS Compute Optimizer finding

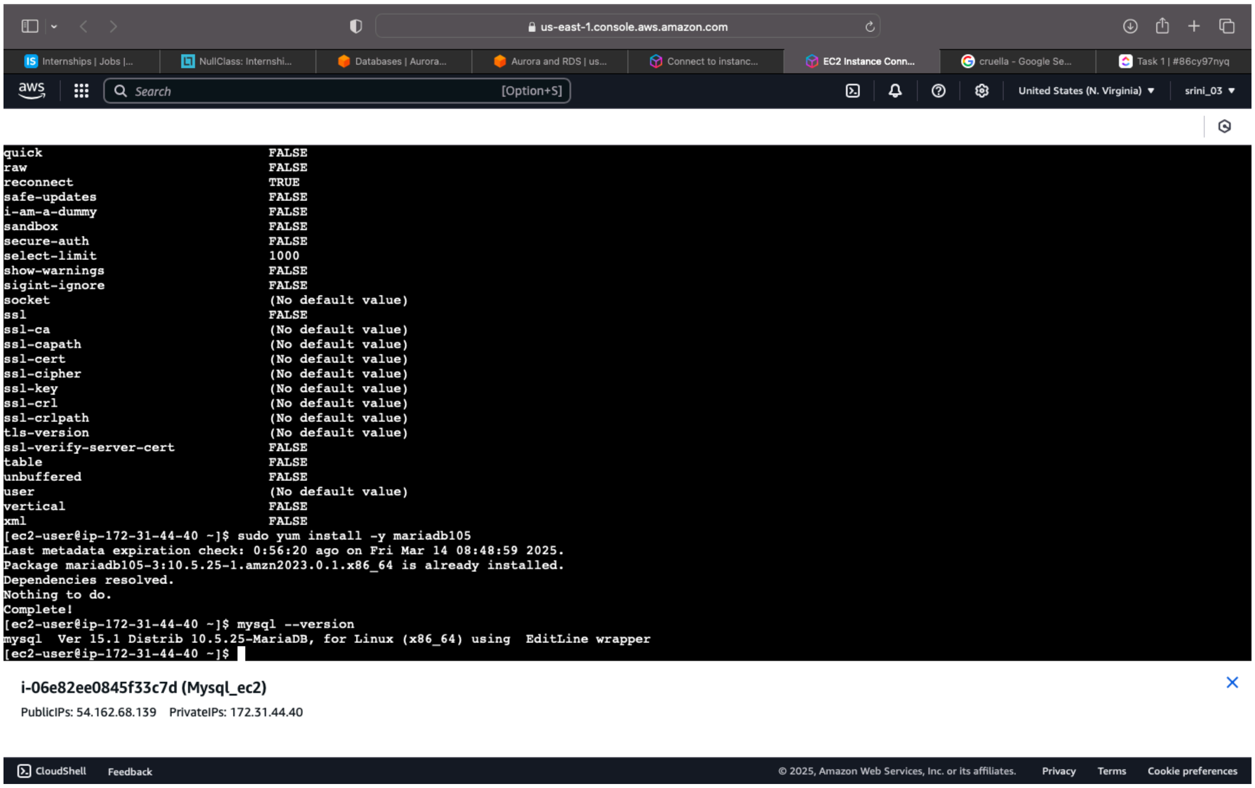
Step 3: Install MySQL Client on EC2

1. **Connect to your EC2 instance** via SSH: basht

   ssh -i "MYSQL.pem" ec2-user@ec2-54-162-68-139.compute-1.amazonaws.com

2. **Install MySQL client** (if not already installed): bash

   sudo yum install -y mysql



```
quick                             FALSE
raw                               FALSE
reconnect                         TRUE
safe-updates                      FALSE
i-am-a-dummy                      FALSE
sandbox                           FALSE
secure-auth                       FALSE
select-limit                      1000
show-warnings                     FALSE
sigint-ignore                     FALSE
socket                            (No default value)
ssl                               FALSE
ssl-ca                            (No default value)
ssl-capath                        (No default value)
ssl-cert                          (No default value)
ssl-cipher                        (No default value)
ssl-key                           (No default value)
ssl-crl                           (No default value)
ssl-crlpath                       (No default value)
tls-version                       (No default value)
ssl-verify-server-cert            FALSE
table                             FALSE
unbuffered                        FALSE
user                              (No default value)
vertical                          FALSE
xml                               FALSE
[ec2-user@ip-172-31-44-40 ~]$ sudo yum install -y mariadb105
Last metadata expiration check: 0:56:20 ago on Fri Mar 14 08:48:59 2025.
Package mariadb105-3:10.5.25-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-44-40 ~]$ mysql --version
mysql  Ver 15.1 Distrib 10.5.25-MariaDB, for Linux (x86_64) using  EditLine wrapper
[ec2-user@ip-172-31-44-40 ~]$
```

i-06e82ee0845f33c7d (Mysql_ec2)

PublicIPs: 54.162.68.139   PrivateIPs: 172.31.44.40

Step 4: Connect to RDS MySQL from EC2

1. Find your **RDS Endpoint**:

   a. **Go to RDS Dashboard** → Click your database.

   b. Copy the **"Endpoint"**

   databasemysql.c27yy0cgo3a2.us-east-1.rds.amazonaws.com

2. **Run the MySQL command**: bash

   $ mysql -h databasemysql.c27yy0cgo3a2.us-east-1.rds.amazonaws.com -u databasemysql -p

3. **Enter your database password** when prompted.

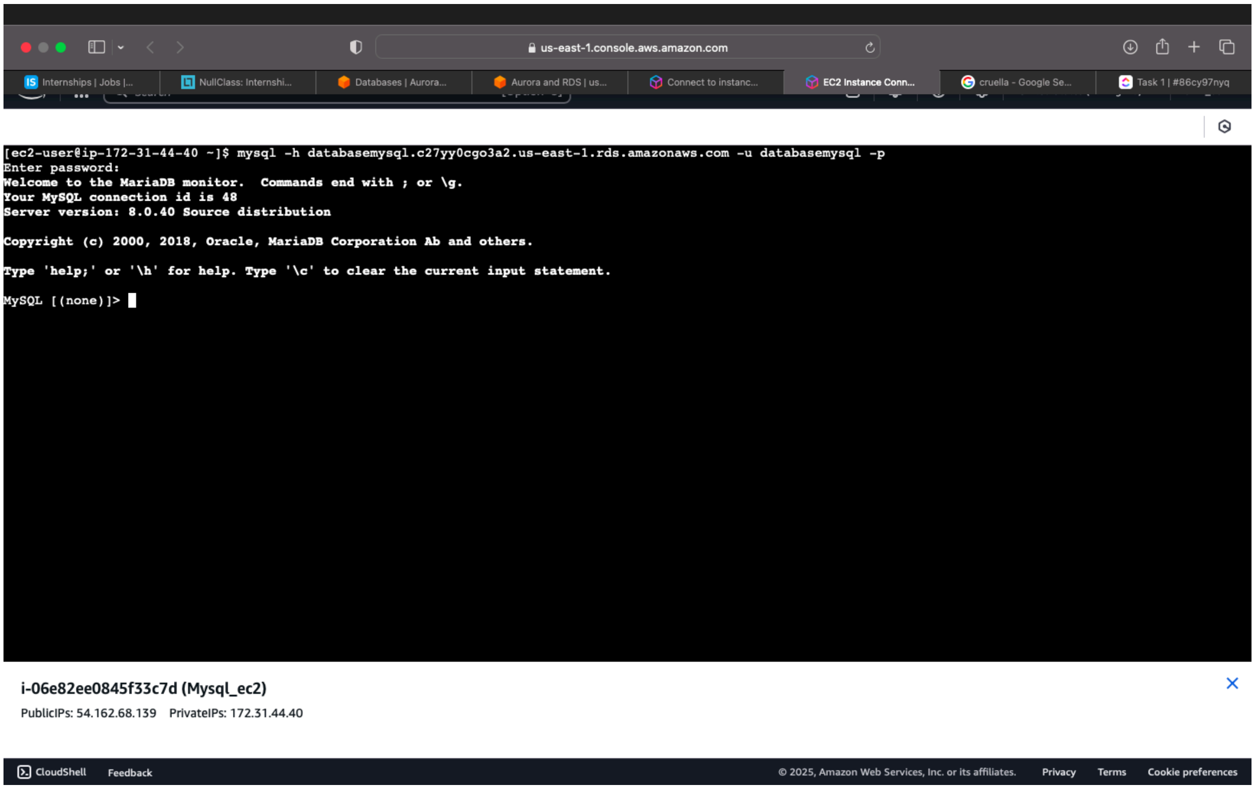
   Password Won't be Visible:

   Password: databasemysql

If successful, you'll see the MySQL prompt:

Shell

mysql>



```
[ec2-user@ip-172-31-44-40 ~]$ mysql -h databasemysql.c27yy0cgo3a2.us-east-1.rds.amazonaws.com -u databasemysql -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 48
Server version: 8.0.40 Source distribution

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

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

MySQL [(none)]> 
```

i-06e82ee0845f33c7d (Mysql_ec2)

PublicIPs: 54.162.68.139   PrivateIPs: 172.31.44.40

Step 5: Create a Database and Table

Once inside MySQL, create a sample database and table:

MYSQL

1.Verify Connection

Once connected, check if your database exists:

SHOW DATABASES;

2. Create Table

CREATE TABLE users (id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50) NOT NULL, email VARCHAR(100) UNIQUE NOT NULL, created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP);

3: Verify the Table

3.Check if the table was created successfully:

SHOW TABLES;

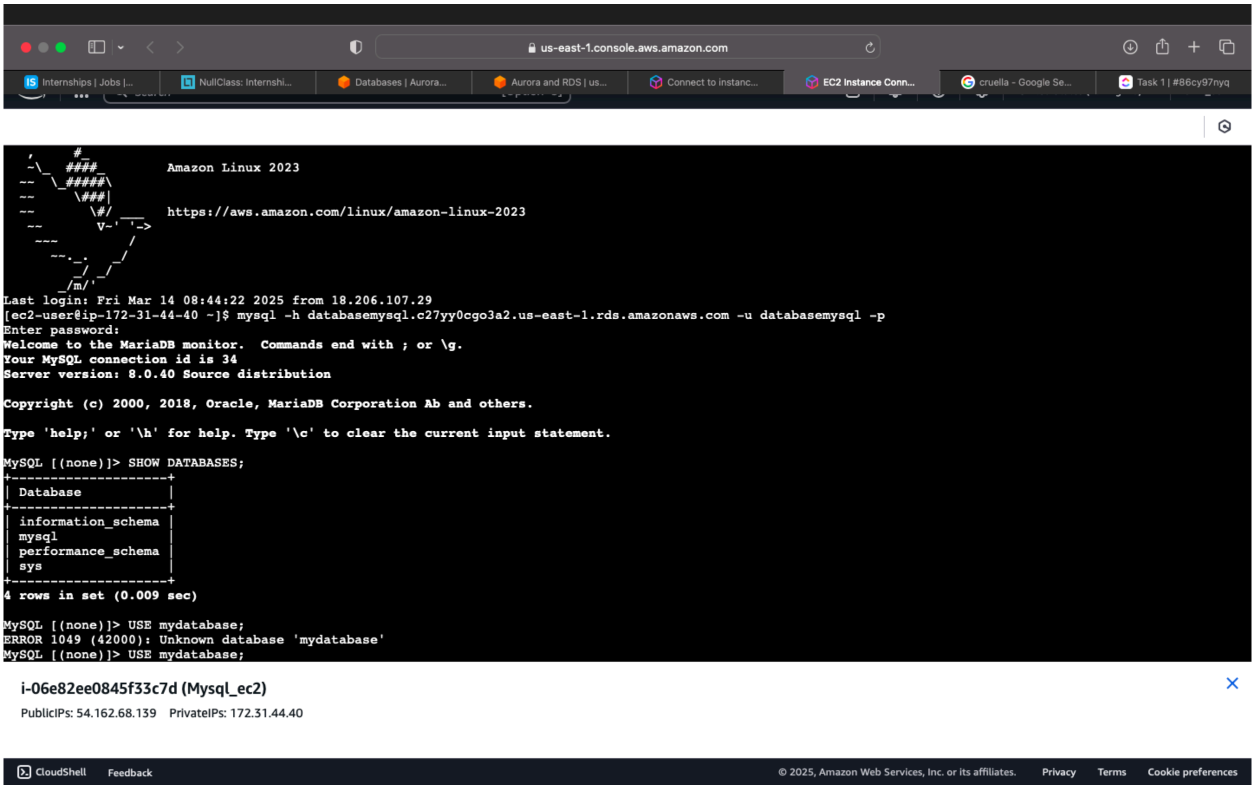
4.Insert Data

INSERT INTO users (name, email) VALUES ('Alice', 'alice@example.com');
INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com');

5: Retrieve Data

Check if the data is stored correctly:

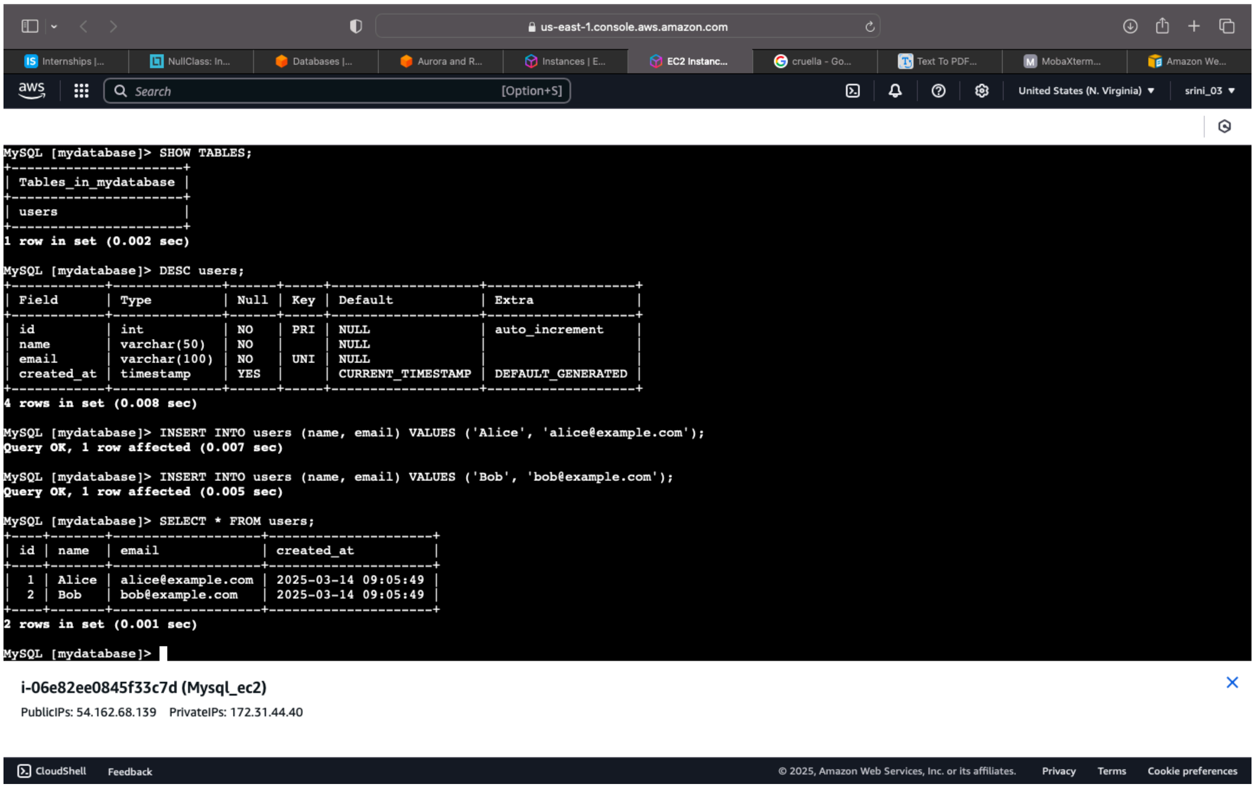
SELECT * FROM users;



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Mar 14 08:44:22 2025 from 18.206.107.29
[ec2-user@ip-172-31-44-40 ~]$ mysql -h databasemysql.c27yy0cgo3a2.us-east-1.rds.amazonaws.com -u databasemysql -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 34
Server version: 8.0.40 Source distribution

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

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

MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.009 sec)

MySQL [(none)]> USE mydatabase;
ERROR 1049 (42000): Unknown database 'mydatabase'
MySQL [(none)]> USE mydatabase;
```

i-06e82ee0845f33c7d (Mysql_ec2)

PublicIPs: 54.162.68.139   PrivateIPs: 172.31.44.40



```
MySQL [mydatabase]> SHOW TABLES;
+----------------------+
| Tables_in_mydatabase |
+----------------------+
| users                |
+----------------------+
1 row in set (0.002 sec)

MySQL [mydatabase]> DESC users;
+------------+--------------+------+-----+-------------------+-------------------+
| Field      | Type         | Null | Key | Default           | Extra             |
+------------+--------------+------+-----+-------------------+-------------------+
| id         | int          | NO   | PRI | NULL              | auto_increment    |
| name       | varchar(50)  | NO   |     | NULL              |                   |
| email      | varchar(100) | NO   | UNI | NULL              |                   |
| created_at | timestamp    | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
+------------+--------------+------+-----+-------------------+-------------------+
4 rows in set (0.008 sec)

MySQL [mydatabase]> INSERT INTO users (name, email) VALUES ('Alice', 'alice@example.com');
Query OK, 1 row affected (0.007 sec)

MySQL [mydatabase]> INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com');
Query OK, 1 row affected (0.005 sec)

MySQL [mydatabase]> SELECT * FROM users;
+----+-------+-------------------+---------------------+
| id | name  | email             | created_at          |
+----+-------+-------------------+---------------------+
|  1 | Alice | alice@example.com | 2025-03-14 09:05:49 |
|  2 | Bob   | bob@example.com   | 2025-03-14 09:05:49 |
+----+-------+-------------------+---------------------+
2 rows in set (0.001 sec)

MySQL [mydatabase]> 
```

i-06e82ee0845f33c7d (Mysql_ec2)

PublicIPs: 54.162.68.139 PrivateIPs: 172.31.44.40