1. Create mariadb db on ec2.

**1. Launch an EC2 Instance**

**a. Go to AWS Console → EC2 → Launch Instance**

* **Name**: mariadb-server
* **AMI**: Amazon Linux 2 (or Ubuntu if you prefer)
* **Instance type**: t2.micro (free tier eligible)
* **Key pair**: Select or create one to connect via SSH
* **Security group**: Allow:
  + SSH (port 22) – your IP

A screenshot of a computer

AI-generated content may be incorrect.

**2. Connect to EC2 via SSH**

ssh -i "car.pem" [ec2-user@3.92.138.195](mailto:ec2-user@3.92.138.195)

A computer screen shot of a computer program

AI-generated content may be incorrect.

3. Install MariaDB

sudo yum update -y

sudo amazon-linux-extras enable mariadb10.5

sudo yum install -y mariadb-server

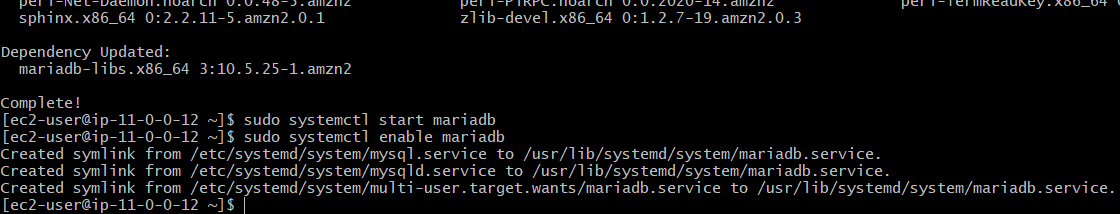
A computer screen with a black background

AI-generated content may be incorrect.

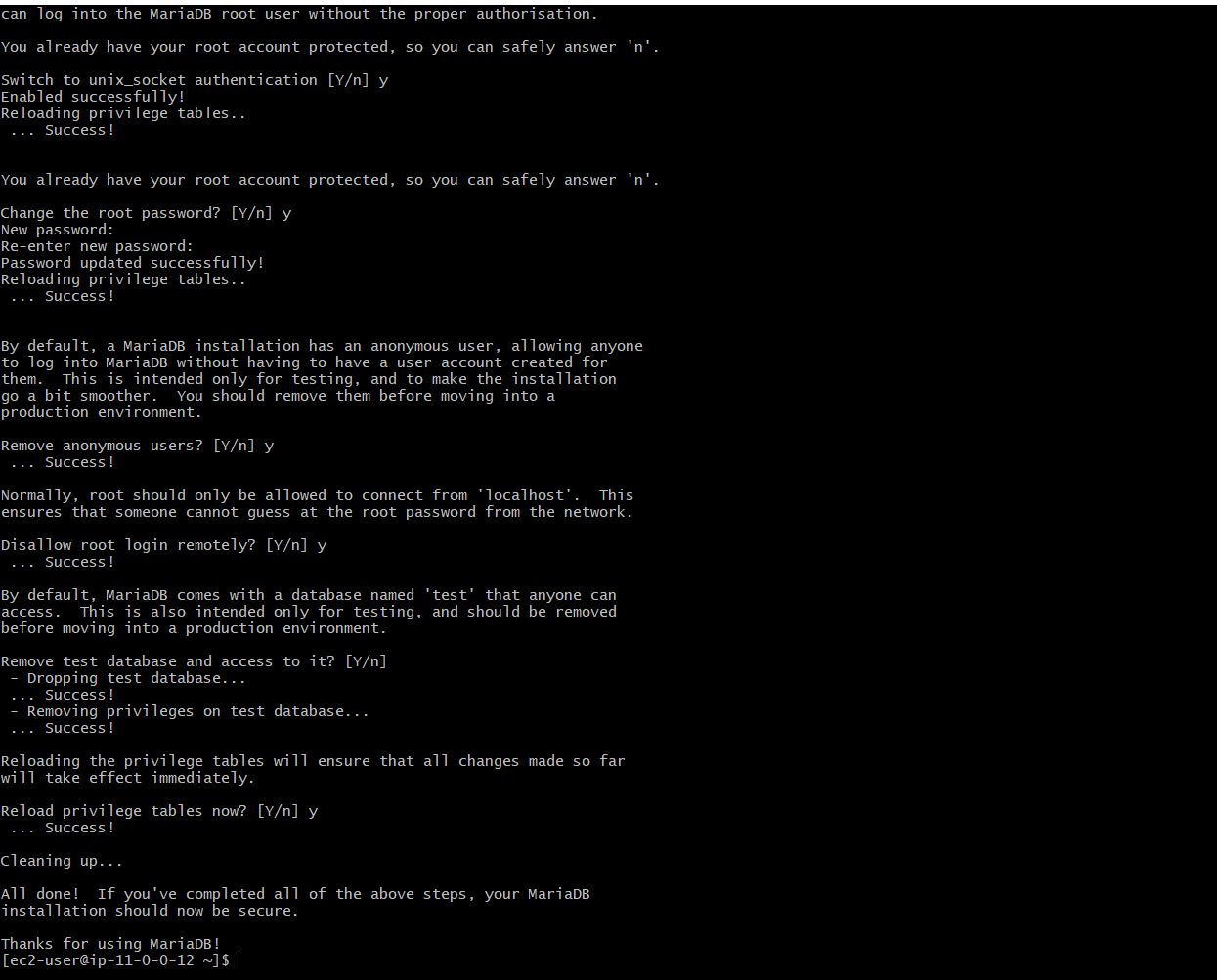
4. Start and Enable MariaDB Service

sudo systemctl start mariadb

sudo systemctl enable mariadb



5. Secure MariaDB Installation



6. Create a Database and User

sudo mysql -u root -p

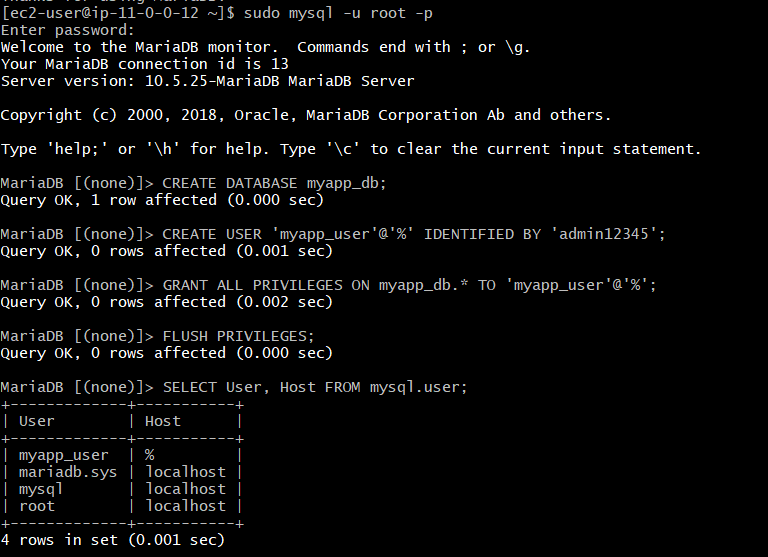
CREATE DATABASE myapp\_db;

CREATE USER 'myapp\_user'@'%' IDENTIFIED BY 'admin12345';

GRANT ALL PRIVILEGES ON myapp\_db.\* TO 'myapp\_user'@'%';

FLUSH PRIVILEGES;

EXIT;



1. Insert some dummy data

Login to MariaDB

🡺sudo mysql -u root -p

Use the myapp\_db Database

🡺 USE myapp\_db;

Create a Table

🡺 CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100),

email VARCHAR(100),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

Insert Dummy Data

🡺INSERT INTO users (name, email) VALUES

('Alice Smith', 'alice@example.com'),

('Bob Johnson', 'bob@example.com'),

('Charlie Brown', 'charlie@example.com'),

('David Lee', 'david@example.com'),

('Eva Green', 'eva@example.com');

View the Inserted Data

🡺 SELECT \* FROM users;

A computer screen shot of a program code

AI-generated content may be incorrect.

1. Take the backup of dummy data on ec2
2. SSH into EC2

🡺 ssh -i "car.pem" [ec2-user@3.92.138.195](mailto:ec2-user@3.92.138.195)

A computer screen shot of a black screen

AI-generated content may be incorrect.

Run the mysqldump Command

🡺 mysqldump -u root -p myapp\_db > myapp\_db\_backup.sql

* -u root: username
* -p: prompt for password
* myapp\_db: your database name
* >: redirect output to a file
* myapp\_db\_backup.sql: the backup file created

This will create the file in your **current directory**

Verify the Backup File

🡺ls -lh myapp\_db\_backup.sql

A black screen with white text

AI-generated content may be incorrect.

1. launch Mariadb RDS instance.

**Step 1: Go to RDS Console**

* Open the AWS Console
* Navigate to: **Services → RDS**

**✅ Step 2: Click "Create Database"**

1. **Choose a database creation method**:
   * Select: **Standard create**
2. **Engine options**:
   * Engine type: **MariaDB**

A screenshot of a computer

AI-generated content may be incorrect.

**Step 3: Specify DB Details**

**a. Templates**

* Choose: Free tier (for practice/lab)

**b. DB instance identifier**

* Example: mariadb-demo

**c. Credentials settings**

* Master username: admin
* Master password: (choose a strong password)
* Confirm password

A screenshot of a computer

AI-generated content may be incorrect.

**Step 4: Configure Instance**

* **DB instance class**: db.t3.micro (Free Tier eligible)
* **Storage**: 20 GB (default is okay)
* Enable storage autoscaling

A screenshot of a computer

AI-generated content may be incorrect.

**Step 5: Connectivity**

**a. Virtual Private Cloud (VPC):**

* Use default or select your custom VPC

**b. Subnet group:**

* Use default subnet group

**c. Public access:**

* Select **Yes** to connect from outside (e.g., your laptop)

**d. VPC security group:**

* Select existing

**Step 7: Create Database**

* Scroll down and click: **Create database**
* RDS will now start provisioning your instance (takes ~5–10 mins)

A screenshot of a computer

AI-generated content may be incorrect.

1. Migrate database from ec2 to RDS.

Dump the EC2 MariaDB Database

🡺mysqldump -u root -p myapp\_db > myapp\_db\_backup.sql

🡺 mariadb-demo.can2qi2mm5t5.us-east-1.rds.amazonaws.com

**Verify the Migration**

**🡺** **mysql -h mariadb-demo.can2qi2mm5t5.us-east-1.rds.amazonaws.com -u admin -p**

**🡺** **USE myapp\_db;**

**SHOW TABLES;**

**SELECT \* FROM your\_table LIMIT 5;**

A screenshot of a computer program

AI-generated content may be incorrect.

1. Install mysql db on ec2

Connect to Your EC2 Instance

Install MySQL Server

sudo yum update -y

sudo yum install -y mariadb-server

sudo systemctl start mariadb

sudo systemctl enable mariadb

Test MySQL Access

sudo mysql

CREATE DATABASE testdb;

SHOW DATABASES;

EXIT;

A screen shot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

1. Launch mysql RDS image

**Step 1: Go to RDS Console**

* Sign in to the [AWS Console](https://console.aws.amazon.com)
* Navigate to: **Services → RDS**
* Click **“Create database”**

**✅ Step 2: Choose Database Creation Method**

* **Standard Create** → allows custom settings

**✅ Step 3: Engine Options**

* **Engine type**: Select **MySQL**
* **Version**: Choose the latest supported version (e.g., 8.0.xx)

**✅ Step 4: Choose Use Case**

* **Templates**: Choose based on your use:
  + **Free tier** (recommended for learning/testing)
  + Production / Dev-Test for real workloads

**✅ Step 5: Settings**

* **DB instance identifier**: mysql-demo
* **Master username**: admin
* **Master password**: Choose a strong one and note it
* Confirm the password

**✅ Step 6: DB Instance Size**

* For Free Tier: Select **db.t3.micro**
* Storage: General Purpose (SSD), 20 GB (default)

**✅ Step 7: Connectivity**

1. **VPC**: Use default or your custom VPC
2. **Subnet group**: Default
3. **Public access**: Choose **“Yes”** if you want to connect from your laptop or outside
4. **VPC security group**:
   * Choose existing or **create new**
   * Make sure to allow **port 3306** from your IP

**✅ Step 8: Additional Configuration**

* **Initial DB name** (optional): e.g., myapp\_db
* Leave backups, monitoring, maintenance as default or customize as needed

**✅ Step 9: Launch the DB**

* Scroll down and click **Create database**
* Wait a few minutes for status to become **"Available"**

A screenshot of a computer

AI-generated content may be incorrect.

**Step 10: Connect to MySQL RDS**

1. Go to **RDS → Databases → mysql-demo**
2. Copy the **endpoint**

mysql -h mysql-demo.can2qi2mm5t5.us-east-1.rds.amazonaws.com -u admin -p

**Important: Security Group Inbound Rule**

To allow connections from outside:

1. Go to **EC2 → Security Groups**
2. Find the security group used by your RDS instance
3. Edit inbound rules:
   * **Type**: MySQL/Aurora
   * **Port**: 3306
   * **Source**: Your public

Example Use Commands Once Connected

CREATE DATABASE testdb;

USE testdb;

CREATE TABLE users (

id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100)

);

INSERT INTO users (name) VALUES ('Alice'), ('Bob');

SELECT \* FROM users;

A computer screen shot of a black screen

AI-generated content may be incorrect.

1. COnfigure multi AZ

**Configure Multi-AZ During RDS Creation**

When creating a new MySQL RDS instance:

1. Go to **RDS → Create database**
2. Under **Availability & durability**:
   * ✅ **Enable Multi-AZ deployment**
   * Choose either:
     + **Multi-AZ DB instance** (1 standby replica)
     + **Multi-AZ DB cluster** (2+ read-write nodes, newer architecture)
3. Complete other steps as normal and click **Create database**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1. Take Backup of db and restore the DB

Take Backup from RDS:  
mysql -h mysql-demo.can2qi2mm5t5.us-east-1.rds.amazonaws.com -u admin -p myapp\_db < myapp\_db\_backup.sql



1. Create ReadReplca

**Step-by-Step: Create a Read Replica**

**🖥️ Option 1: Using AWS Console**

**Login** to the [AWS Management Console](https://console.aws.amazon.com/)

Go to **RDS → Databases**

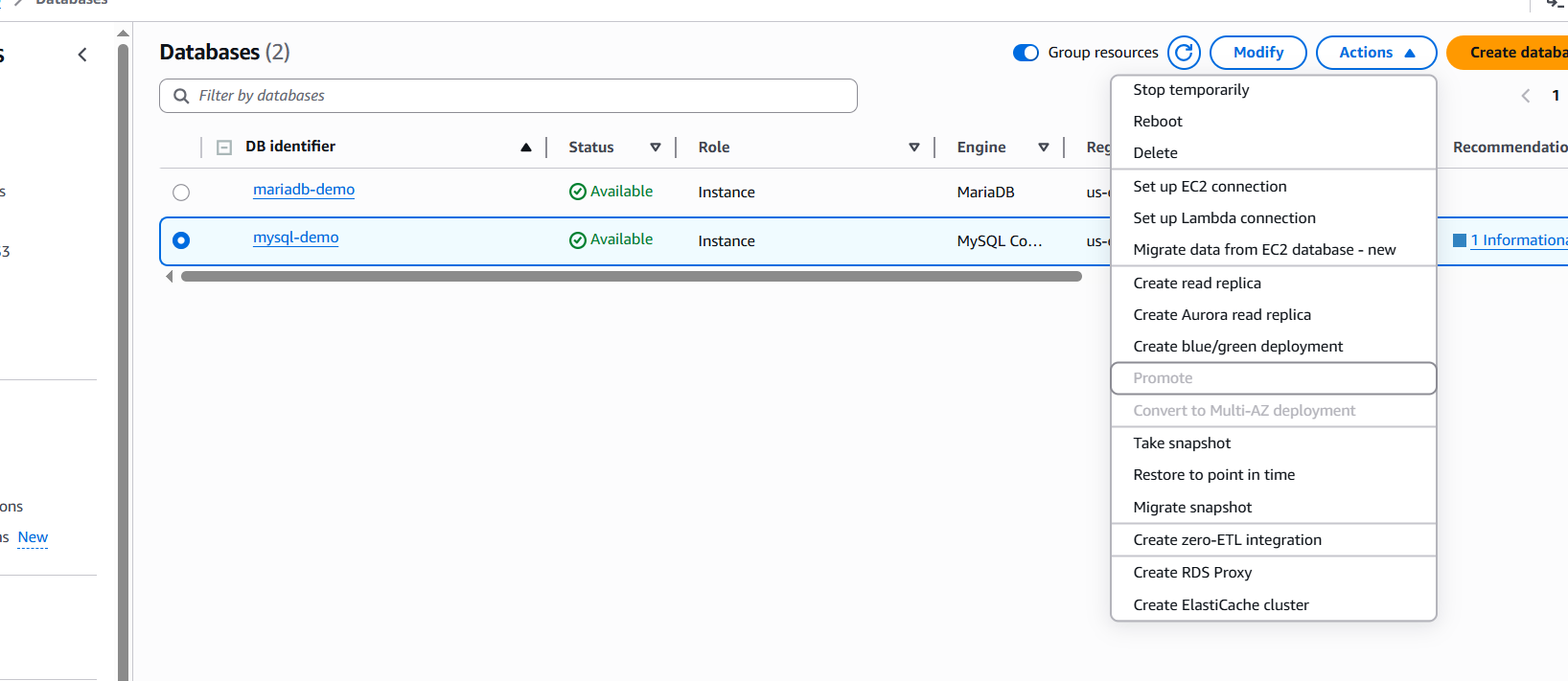
Click on your **source DB instance** (e.g., mysql-demo)

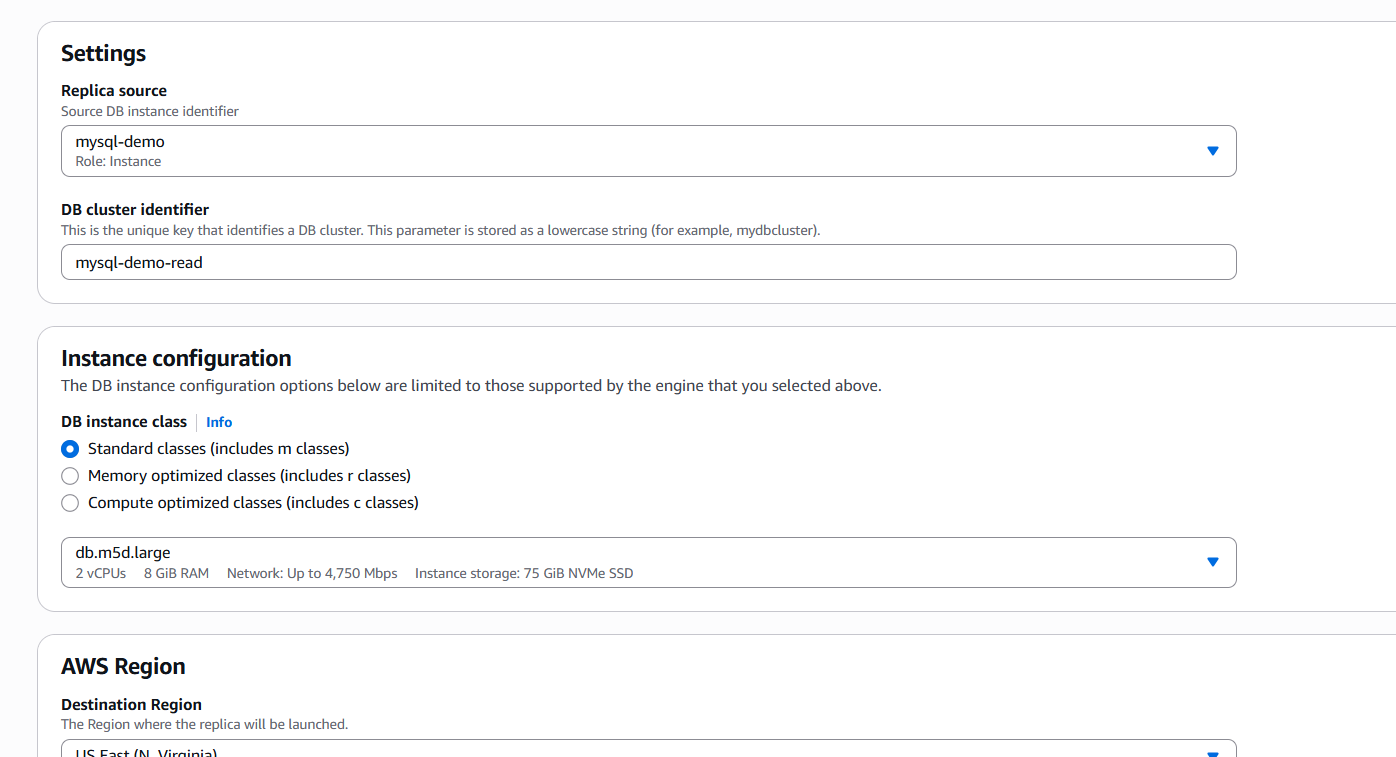
Click **Actions → Create read replica**

Fill out the form:

* + **DB instance identifier**: e.g., mysql-demo-read
  + **DB instance class**: Select size (e.g., db.t3.micro)
  + **Multi-AZ deployment**: (Optional)
  + **Public access**: Set to "Yes" or "No" based on your use case

Click **Create read replica**





A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.