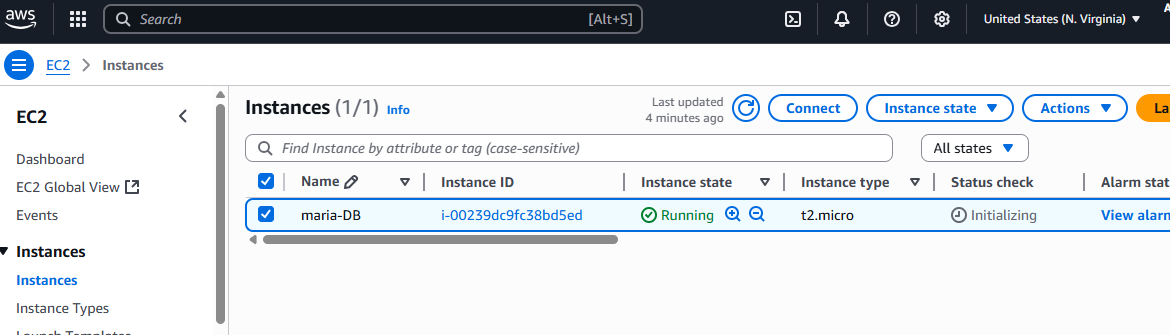
**Relational Database service(RDS)**

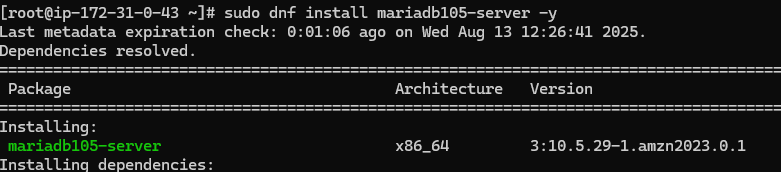
1. **Create mariadb db on ec2.**

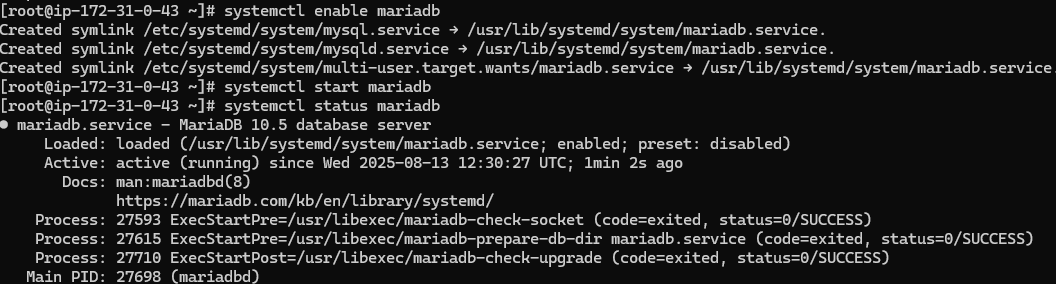
**STEP:1**

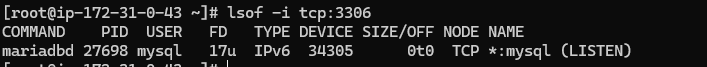
**Launch instance**

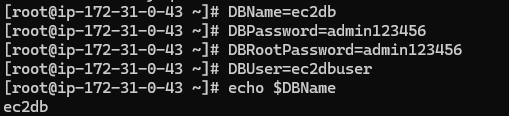
Go to **AWS Console → EC2 → Launch instance**.

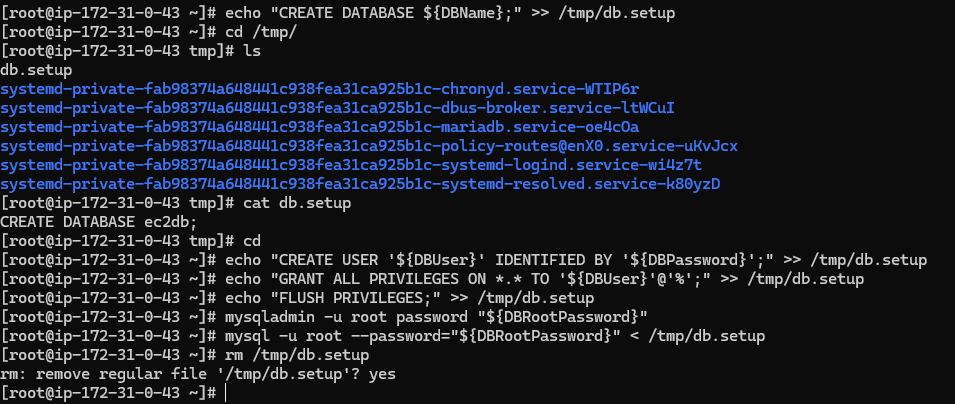




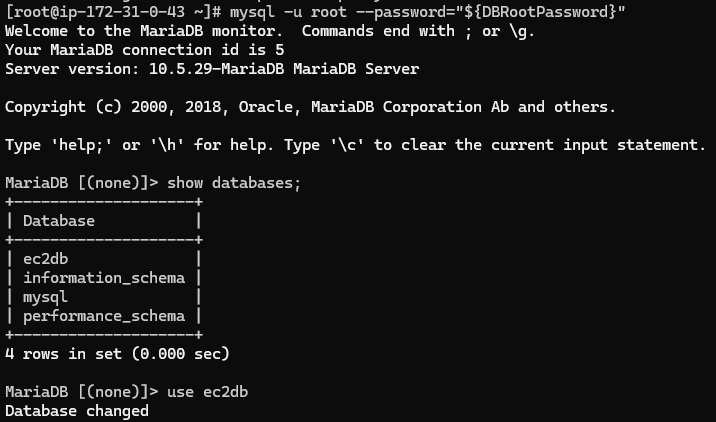


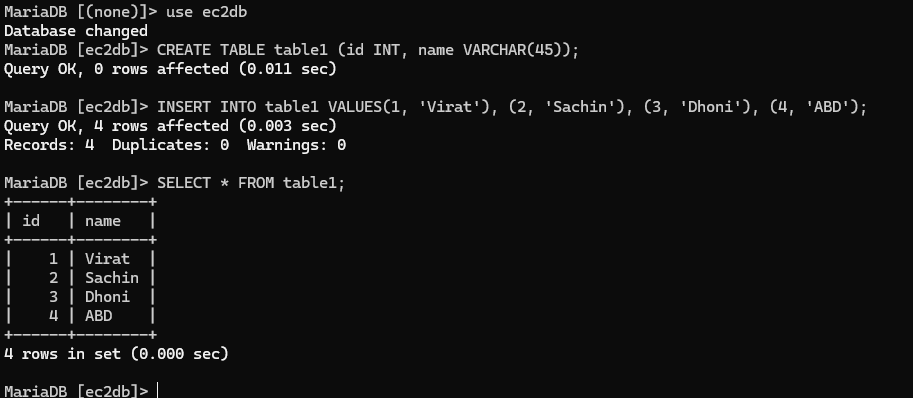




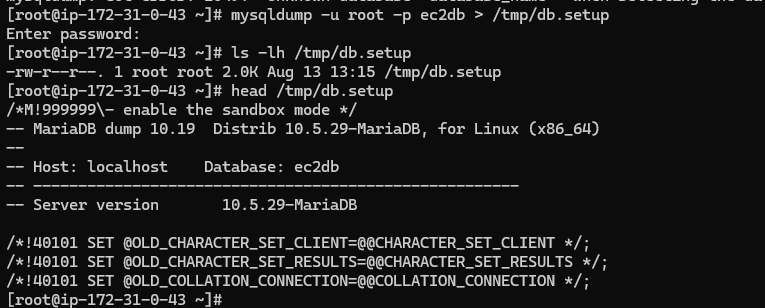


**2) Insert some dummy data**



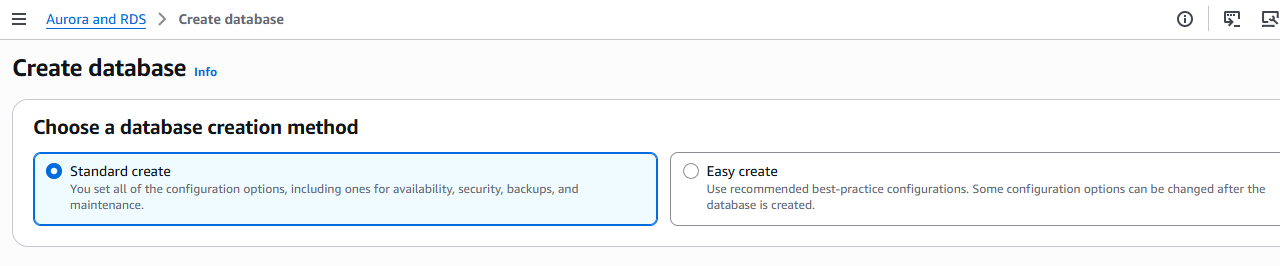


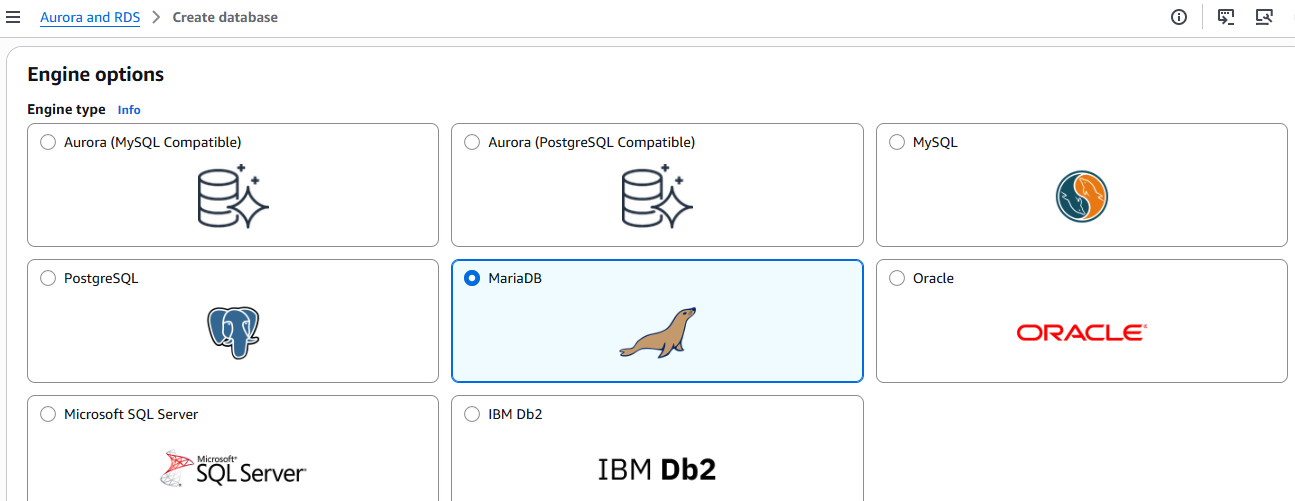
**3) Take the backup of dummy data on ec2**

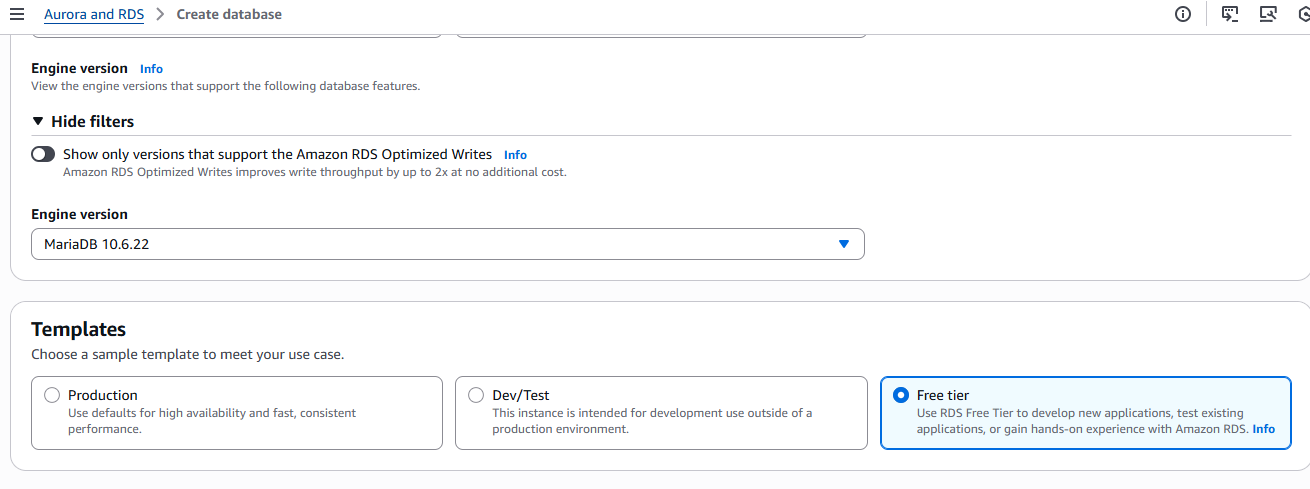


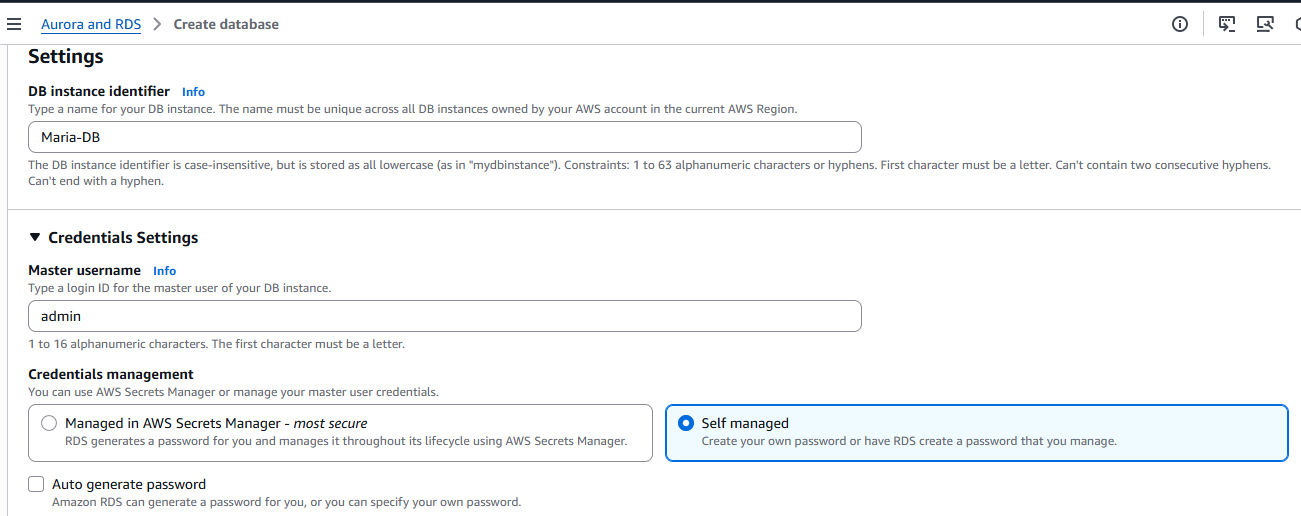
**4) launch MariaDB RDS instance.**

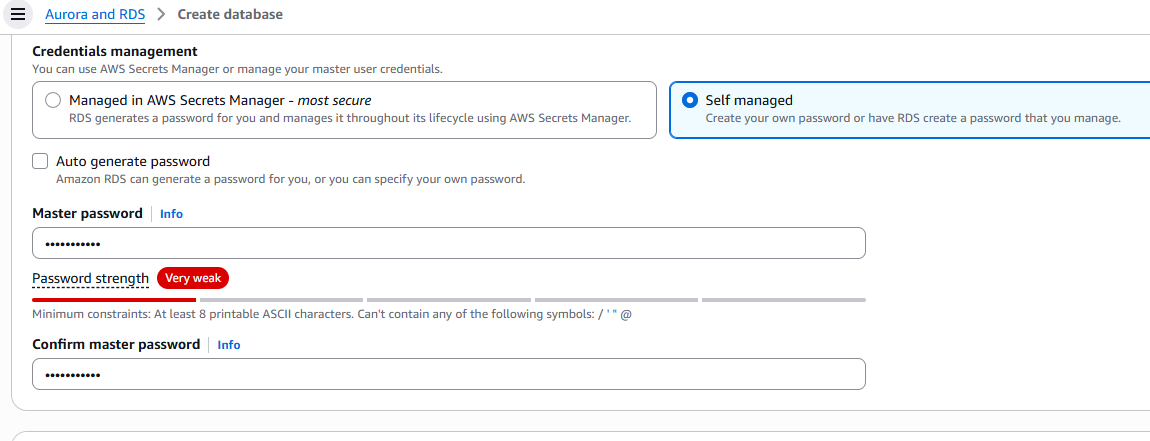
**Aurora and RDS console ---> Database --->** click **database**

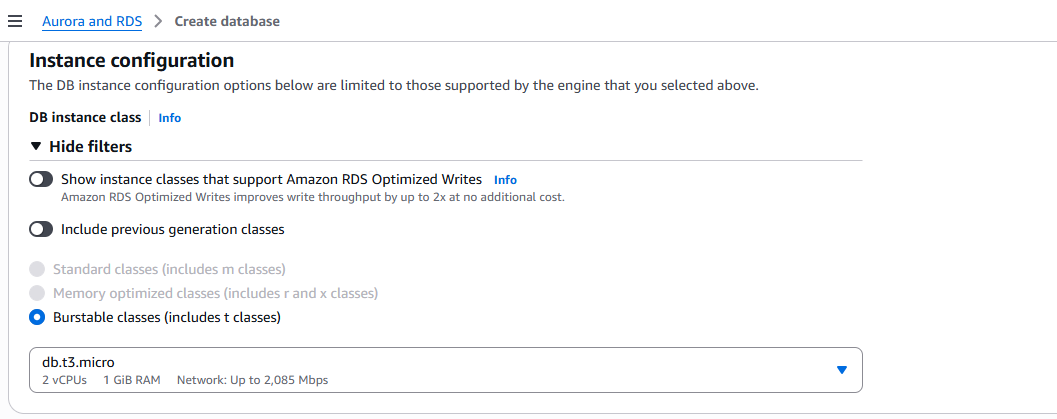


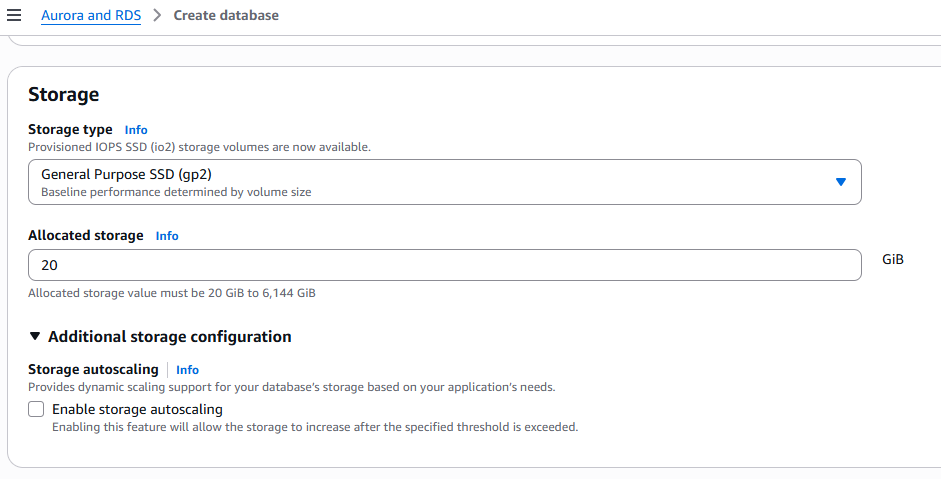


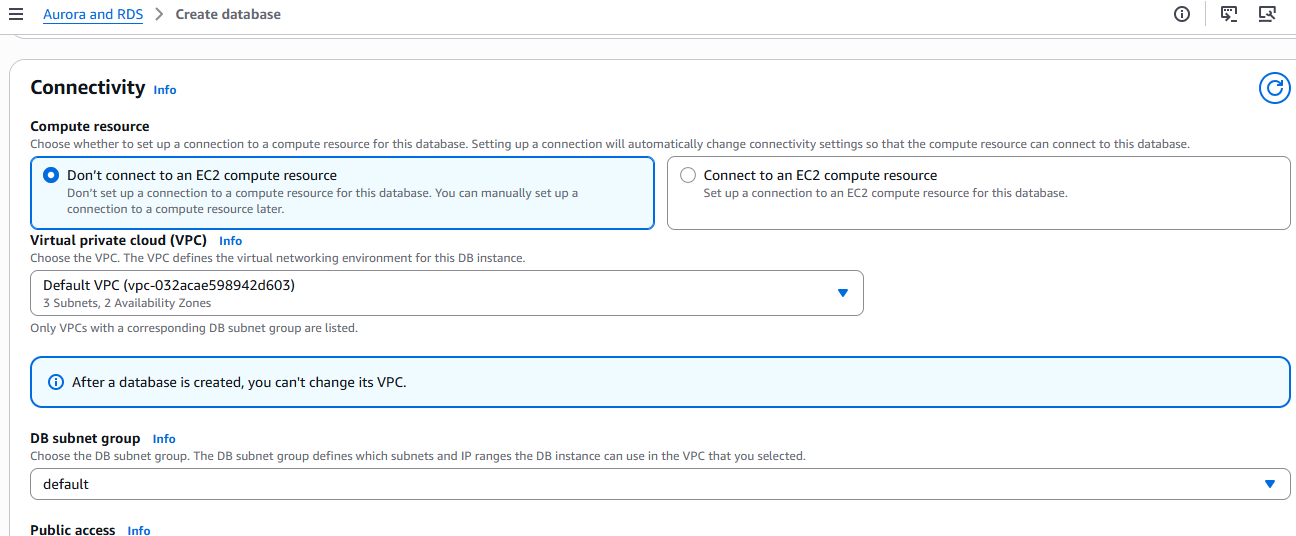


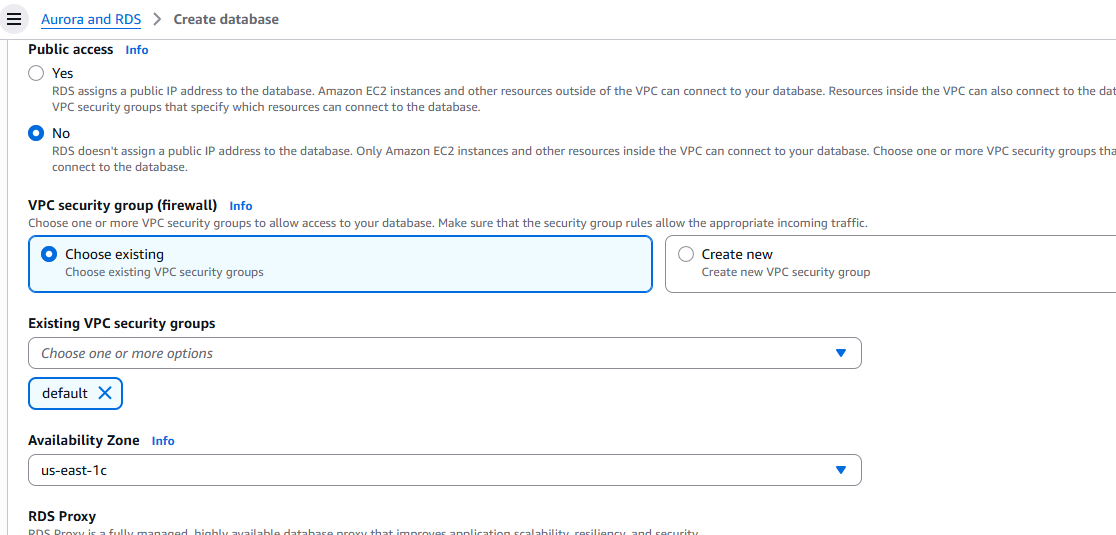


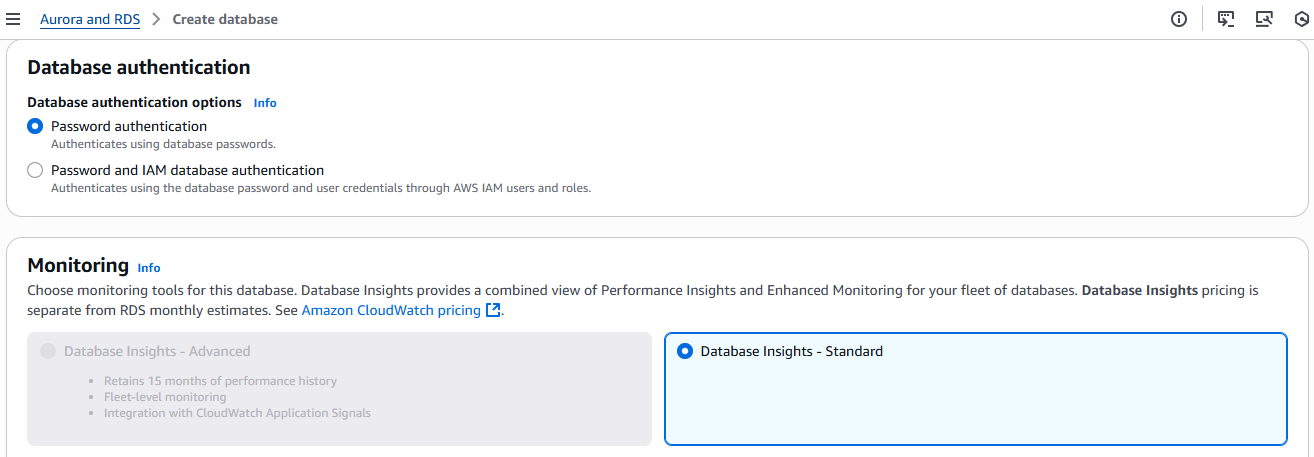


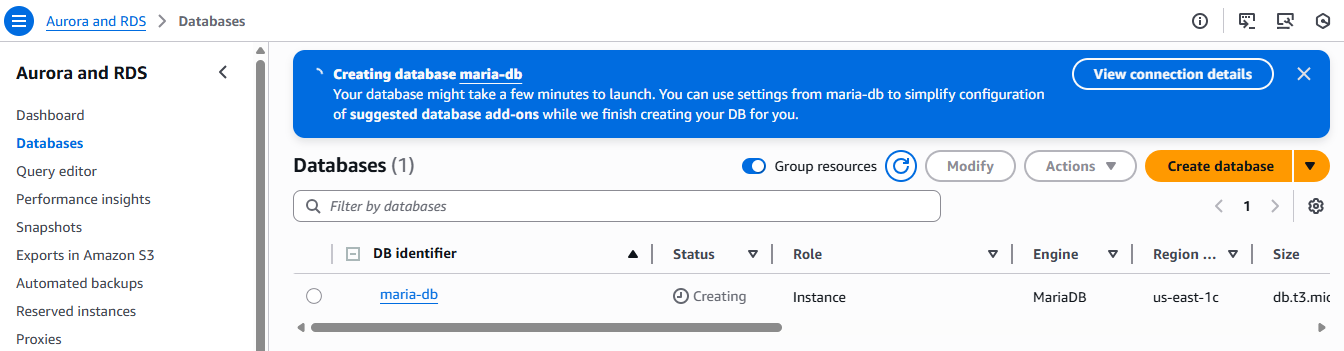




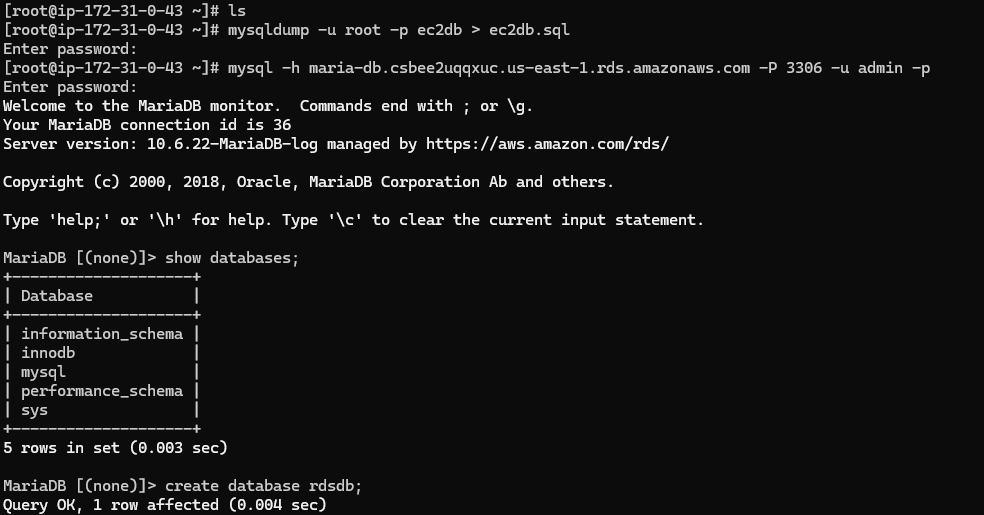


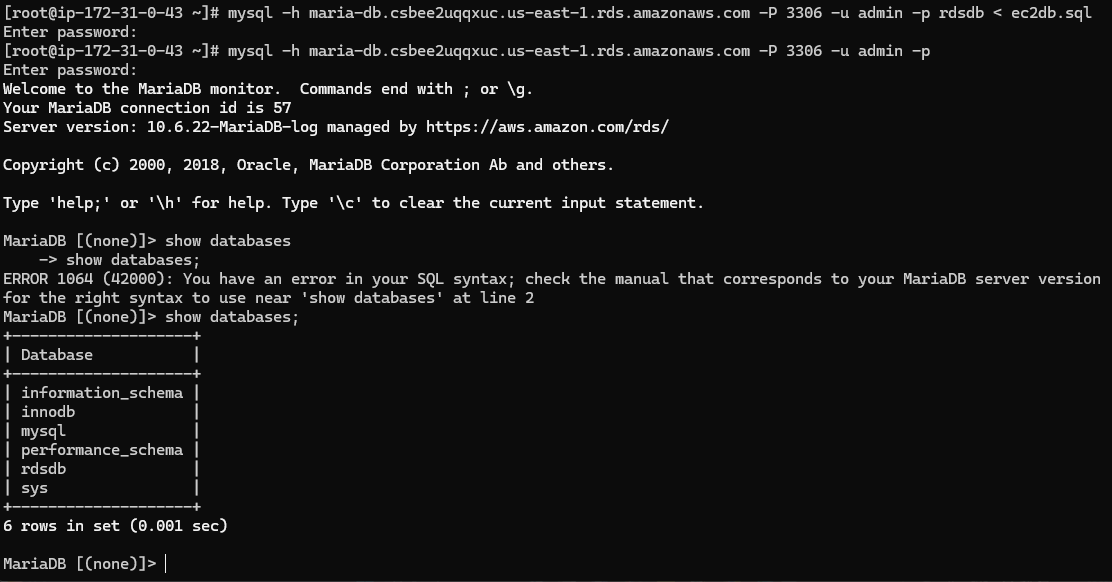


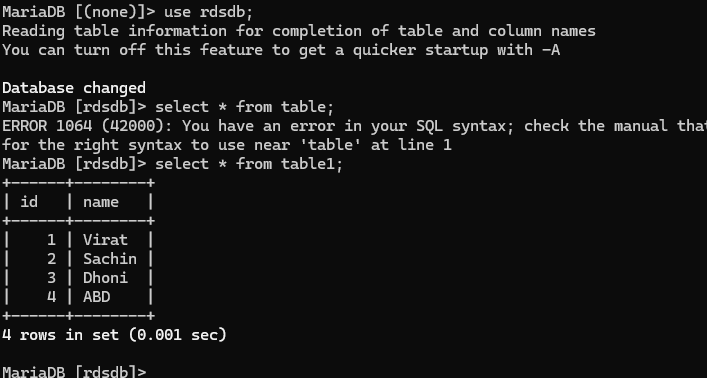




**5) Migrate database from ec2 to RDS.**







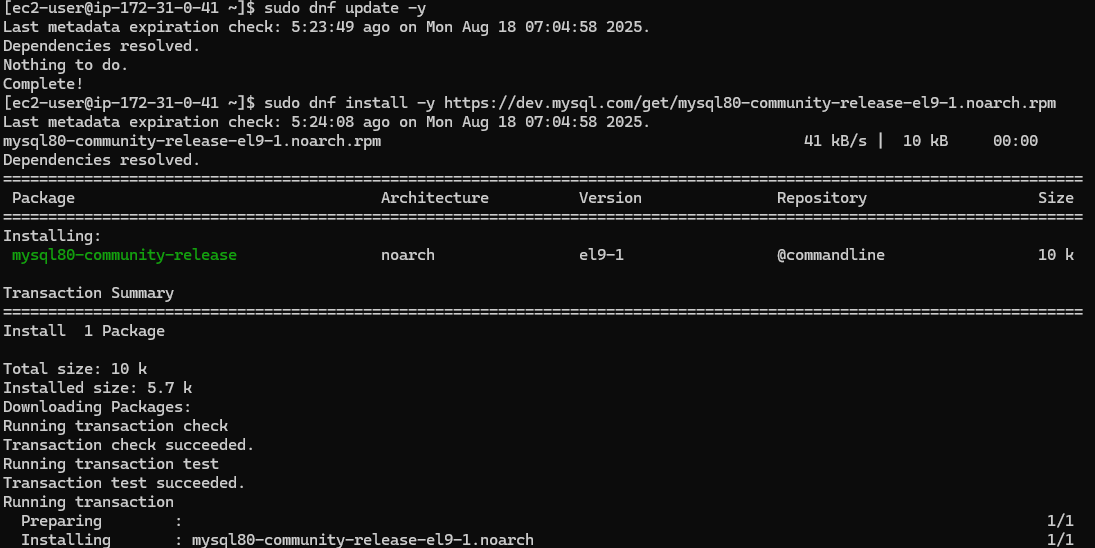
**6) Install MySQL DB on ec2**

**Update system**

sudo dnf update -y

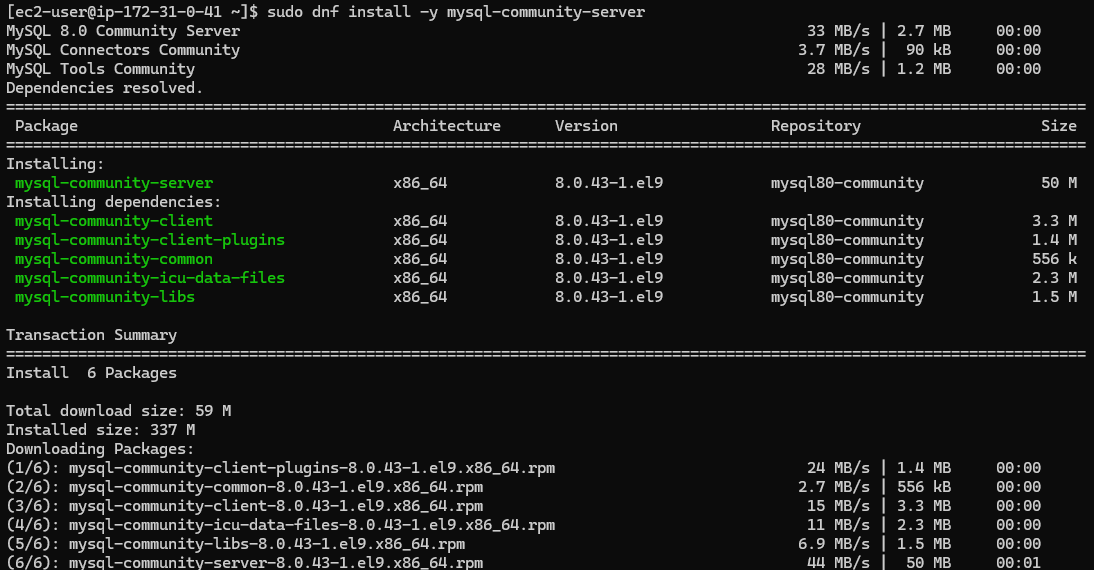
**Add MySQL Yum repository**

sudo dnf install -y <https://dev.mysql.com/get/mysql80-community-release-el9-1.noarch.rpm>



**Install MySQL server**

sudo dnf install -y mysql-community-server



**Start & enable service**

sudo systemctl enable --now mysqld

sudo systemctl start mysqld



If doesn’t start/enable

Then

# Clean up failed cache

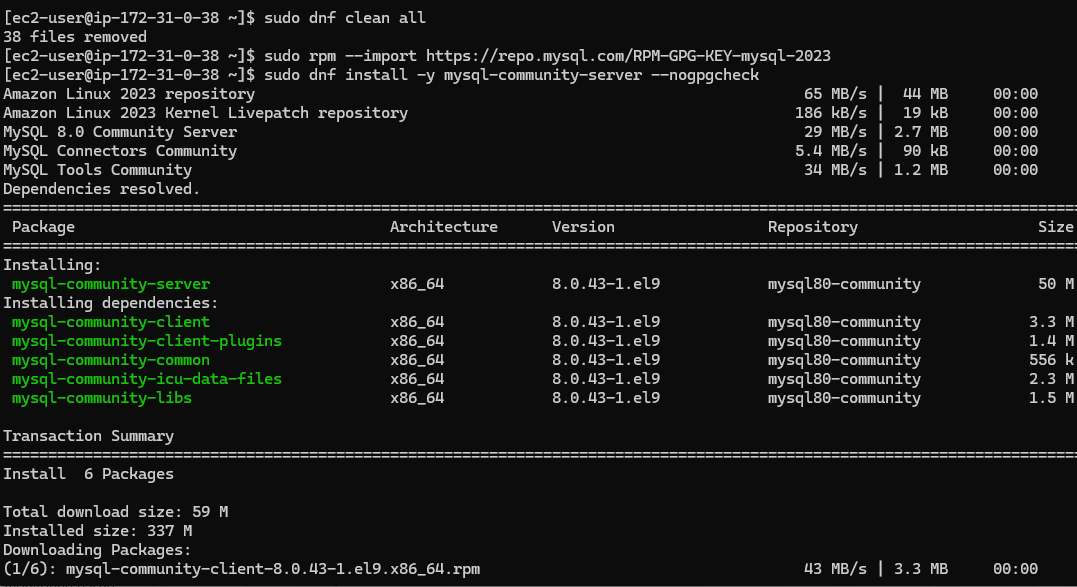
sudo dnf clean all

# Import MySQL GPG Key properly

sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2023

# Try installing again

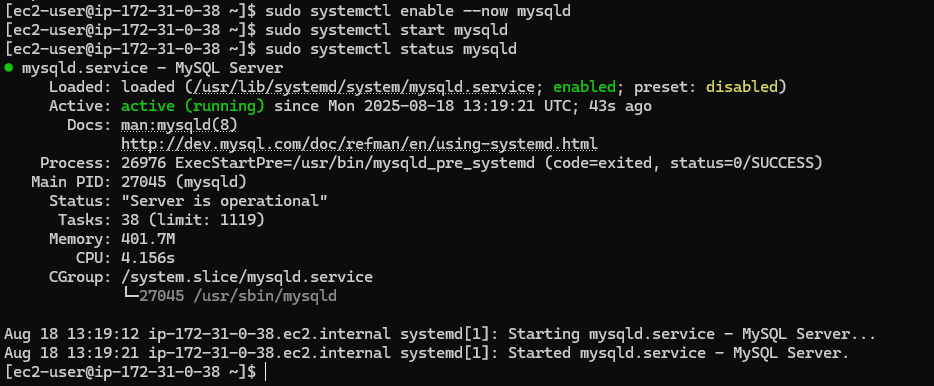
sudo dnf install -y mysql-community-server --nogpgcheck



sudo systemctl enable --now mysqld

sudo systemctl start mysqld

sudo systemctl status mysqld

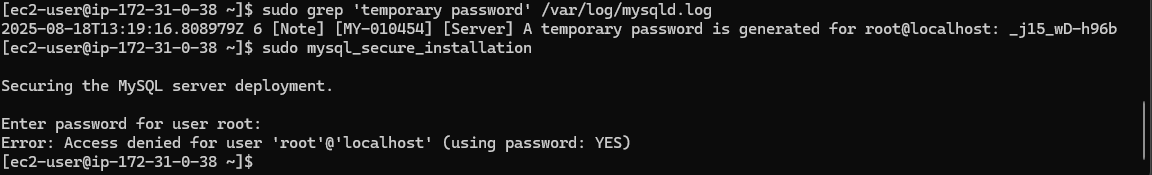


**Get temporary root password**

sudo grep 'temporary password' /var/log/mysqld.log

**Secure MySQL**

sudo mysql\_secure\_installation



**7) Launch MySQL RDS image**

**Sign in to AWS Console**

Go to → Services → RDS → Databases → Create database.

**Choose Database Creation Method**

****Standard create (gives you full control)

Or Easy create (simpler, but fewer options)

Select Standard create for more flexibility.

Engine Options

Engine type → Select MySQL

Version → Choose a stable version (e.g., MySQL 8.0.x LTS)

**Templates**

**Choose based on use case:**

**Relational Data Base Task**

****Production (Multi-AZ, backups enabled, more costly)

Dev/Test (Single-AZ, cheaper)

**Settings**

**DB instance identifier →** e.g., mydb-mysq**l**

**Master username →** e.g., admin

**Master password →** choose a strong password (or auto-generate and

download)

**Instance Configuration**

****Instance type → e.g., db.t3.micro (free-tier eligible) or db.t3.medium for small

**workloads**

****Storage → General Purpose (gp3), start with 20GB (auto-scaling optional)

**Connectivity**

****VPC → Choose your VPC (default or custom)

Subnet group → Usually default unless you made custom subnets

Public access →

-Yes if you want to connect from outside AWS (not recommended for

production)

-No if only accessed inside VPC (best practice)

VPC security group → Create/attach one allowing TCP 3306 from your IP orapp servers

**Additional Config**

****Initial database name → e.g., appdb

Automatic backups → Enable (set retention, e.g., 7 days)

Encryption → Enable if required

Monitoring → Enable Enhanced monitoring if needed

**Create Database**

Click Create database → AWS will provision your RDS instance (may take 5–10

minutes).

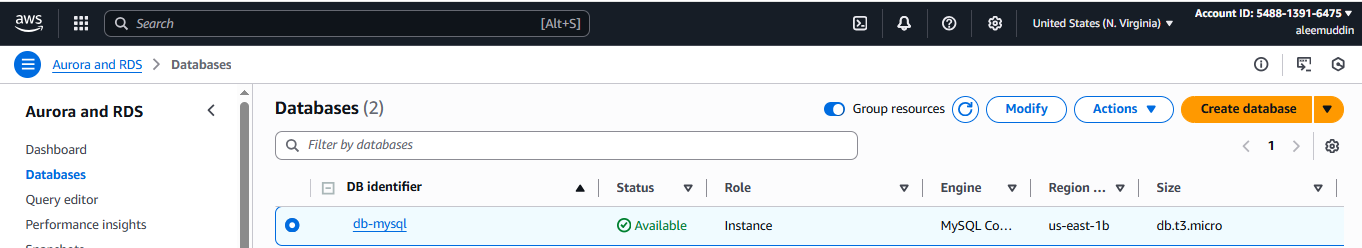
**Connect to RDS**

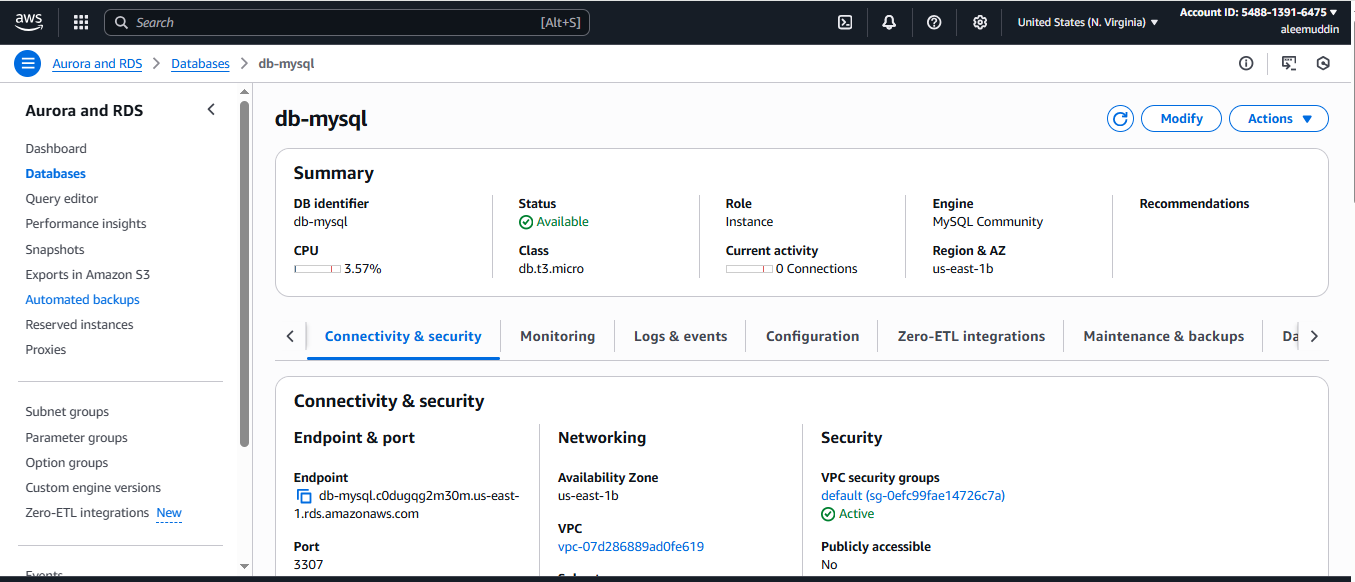
**Once status = Available:**

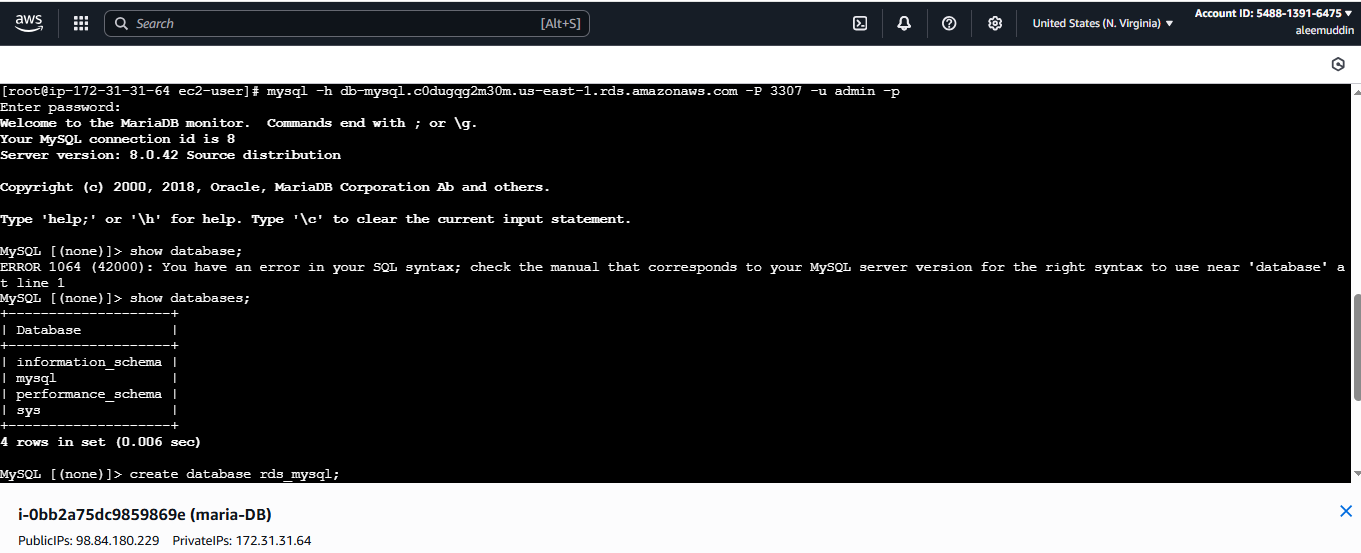
Copy the Endpoint (e.g., mydb-mysql.abc123xyz.us-east-1.rds.amazonaws.com)

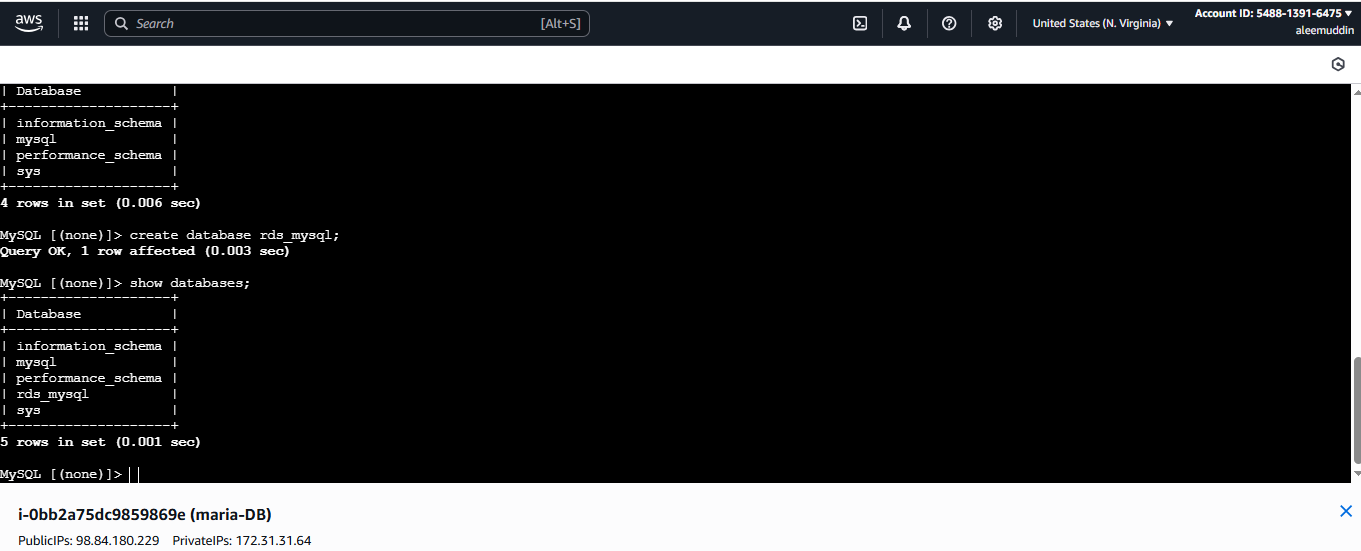
Connect from EC2 or laptop**:**

**mysql -h db-mysql.c0dugqg2m30m.us-east-1.rds.amazonaws.com -P 3307 -u admin -p**

****

****

****

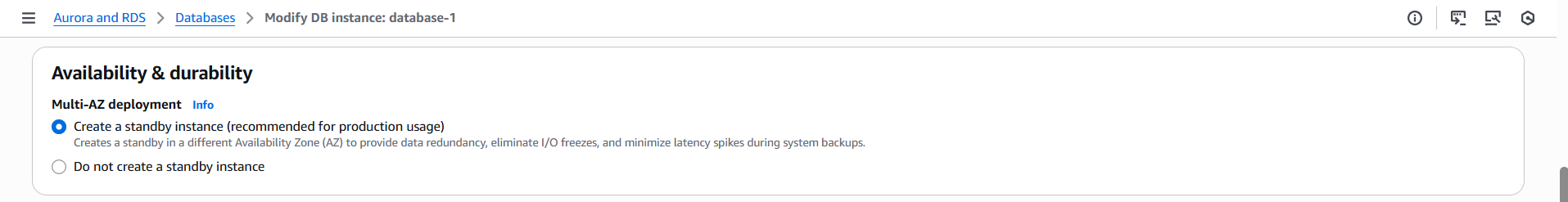
****

**8) Configure multi AZ**

In AWS Console → RDS → Select your DB → **Modify or (we can go from action → convert to Multi-AZ)**

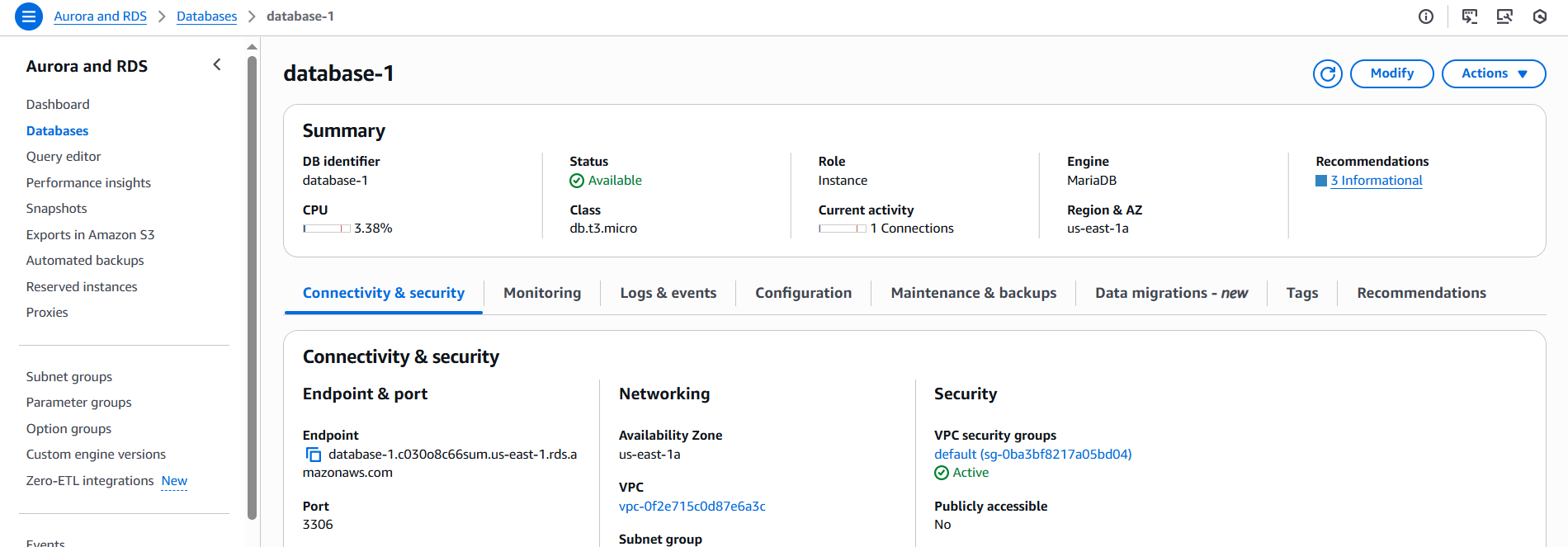
Under **Availability & durability**, enable **Multi-AZ deployment**

**Apply immediately**

****

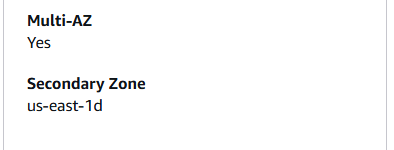
**The primary AZ showing in → Connectivity and security**

**us-east-1a**

****

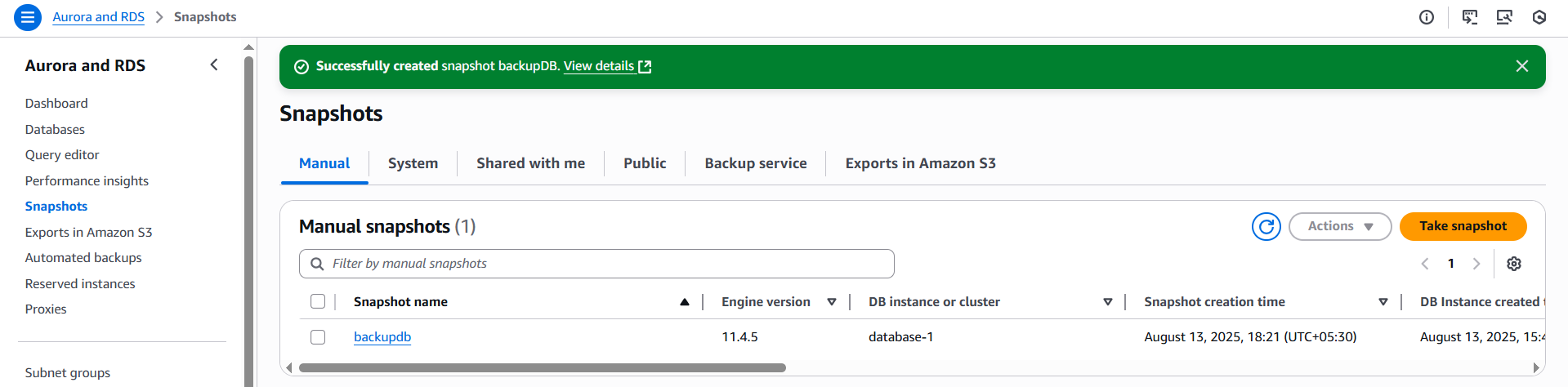
In configuration tab we can check multi-AZ

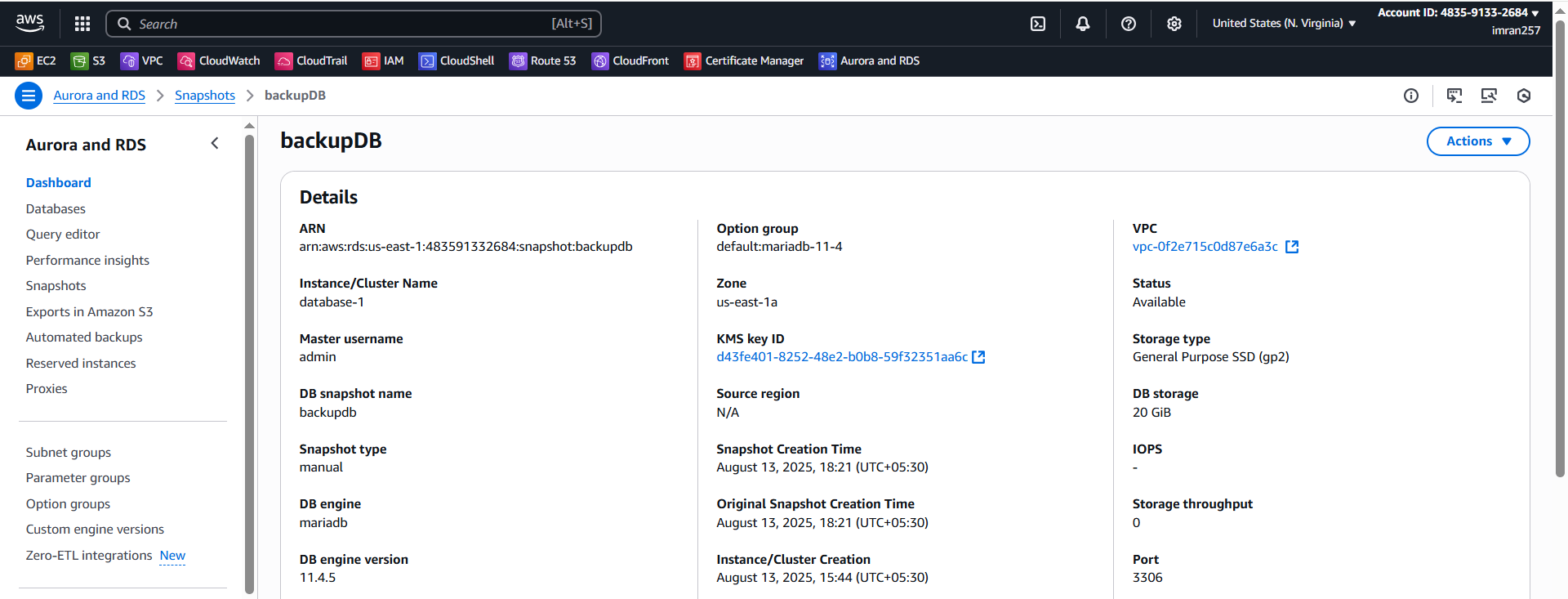
us-east-1d

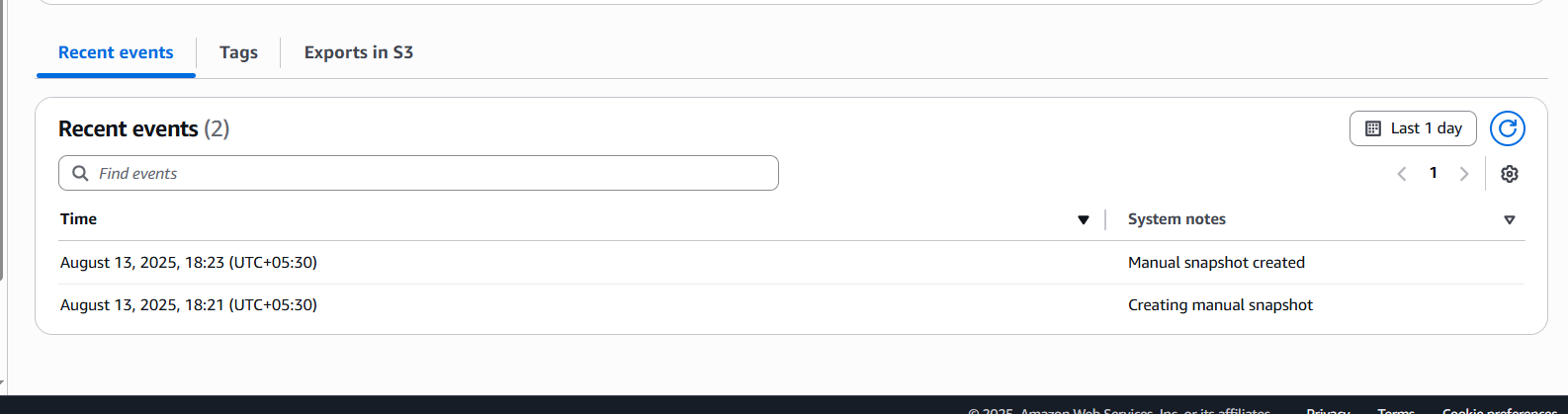


**9) Take Backup of DB and restore the DB**

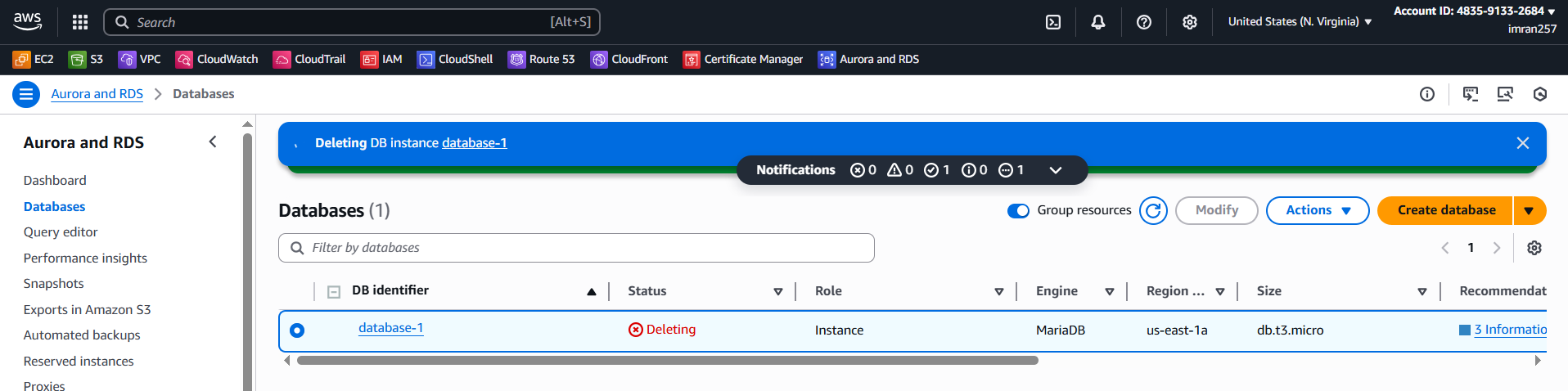
Go to actions → snapshot and name it and create

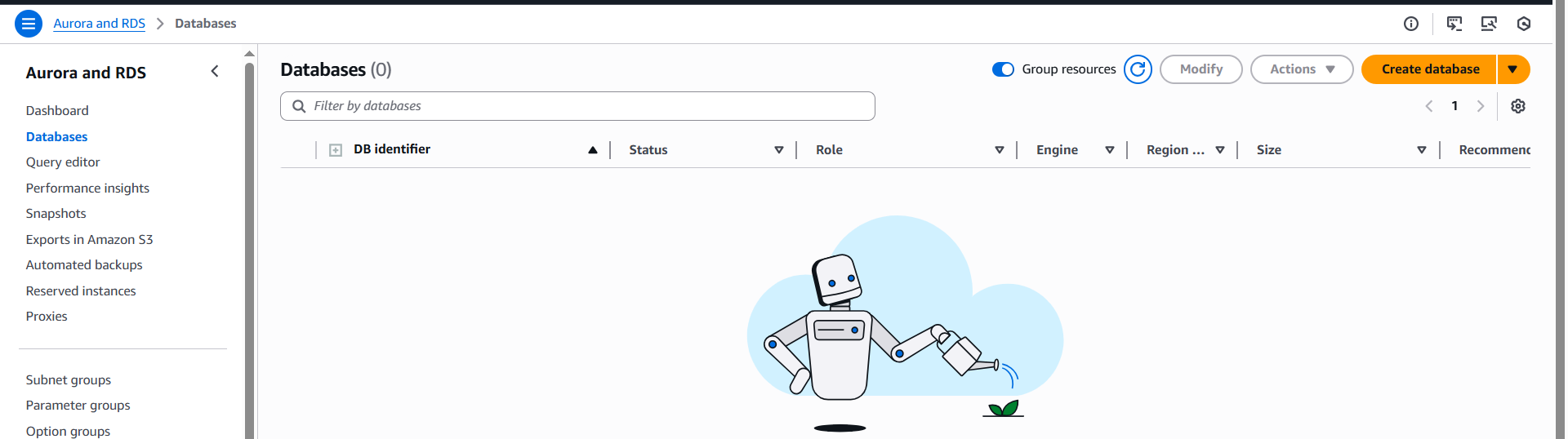


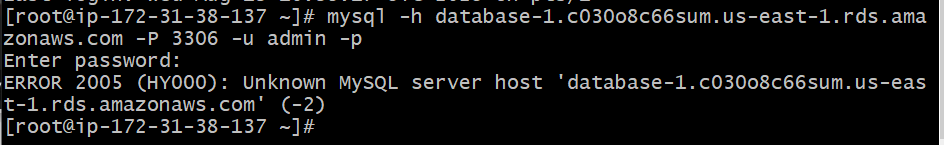




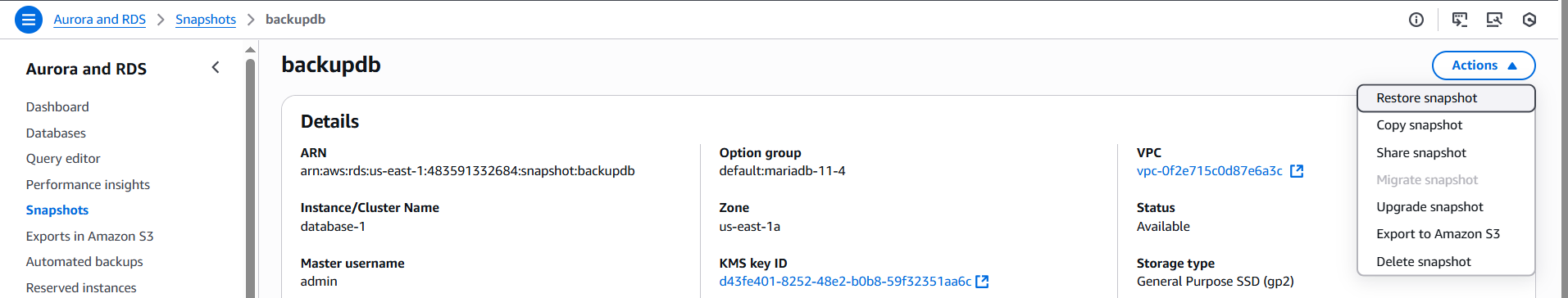
Now deleting database



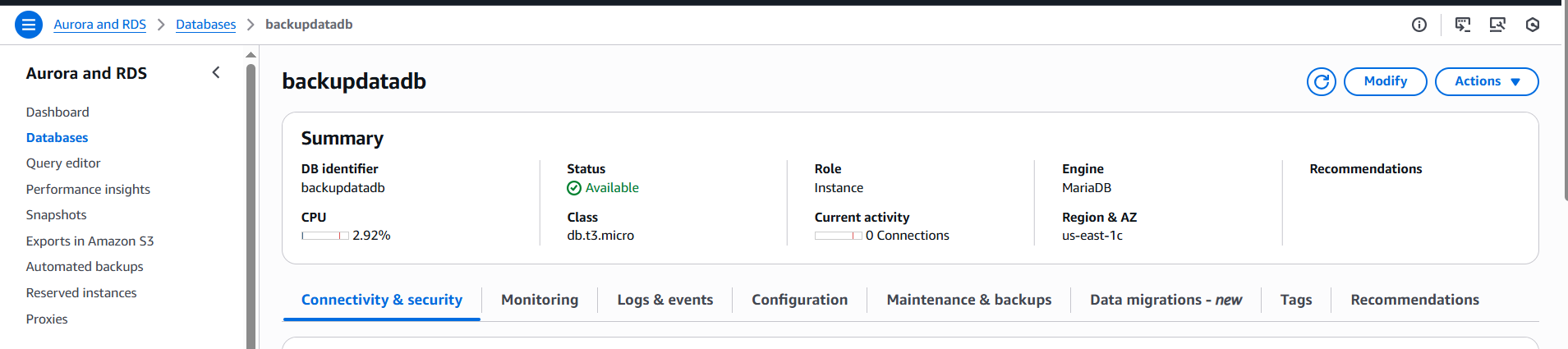




Restoring db from backup

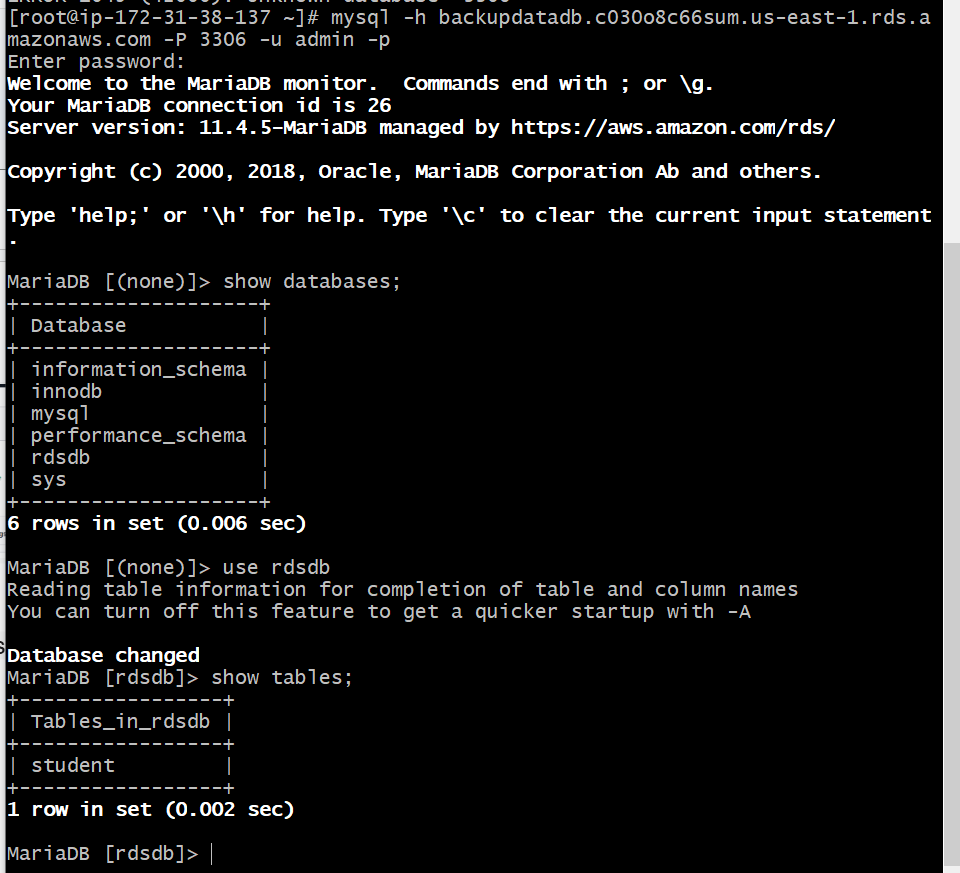


The backupdb restored as backupdatadb



Checking with endpoint

mysql -h backupdatadb.c030o8c66sum.us-east-1.rds.amazonaws.com -P 3306 -u admin -p



**10) Create Read Replica**

**A Read Replica is a copy of your database that automatically gets updated from the primary DB using asynchronous replication.**

**Main Benefits:**

* Read Scaling – Offload heavy SELECT queries from the primary DB to the replica.
* Disaster Recovery – If the primary DB fails, you can promote the replica to become the new primary.
* Reporting – Use the replica for analytics/reporting without affecting primary performance.
* Global Performance – Place replicas in other AWS regions to serve local users faster.

**Go to RDS → Databases.(Make sure your database Enabled Automatic Backup )**

Select your **MySQL instance.**

**Click Actions → Create read replica.**

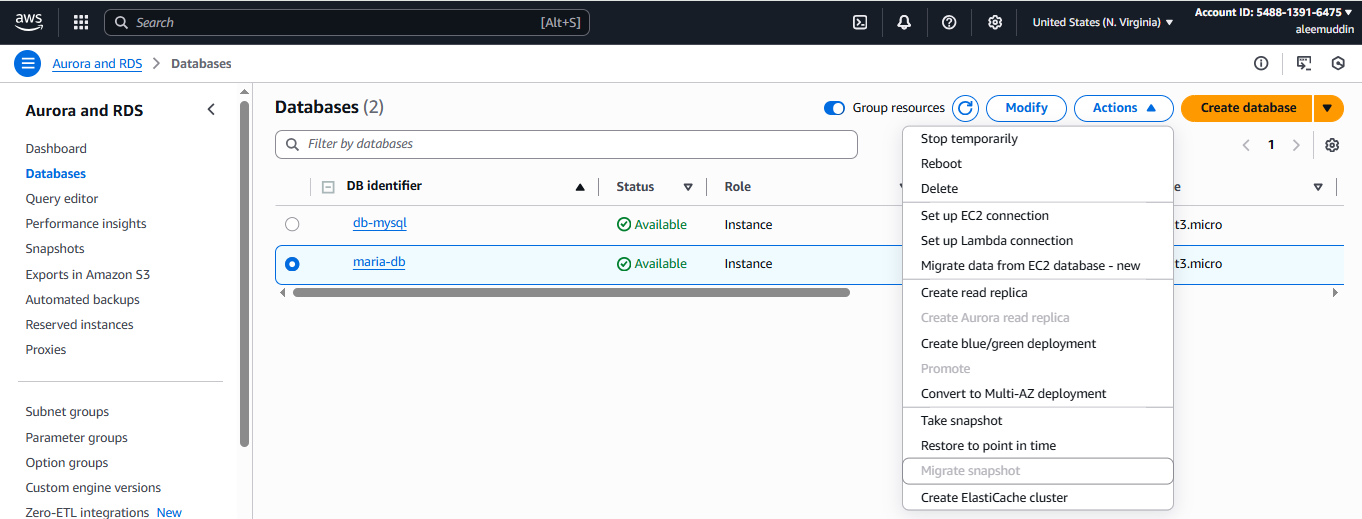
**Settings:**

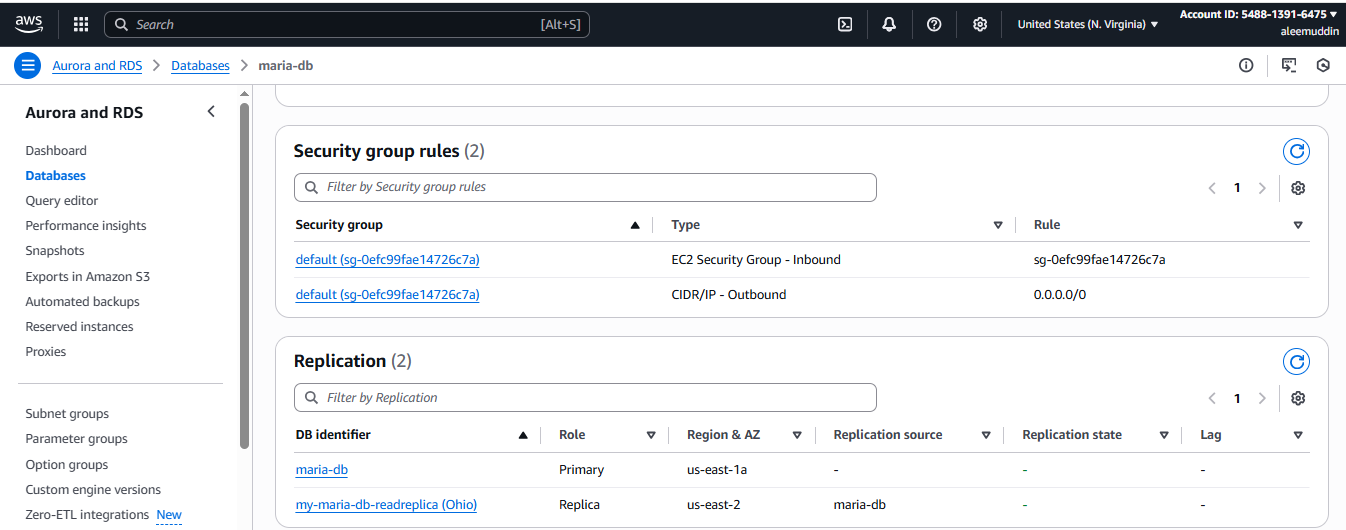
* **DB instance identifier: my-maria-db-readreplica**
* **Optionally choose a different AZ or region (for cross-region).**

**Keep Multi-AZ disabled for read replicas** (they are for scaling reads).

Click **Create read replica.**

Wait until status changes to available.

****

****