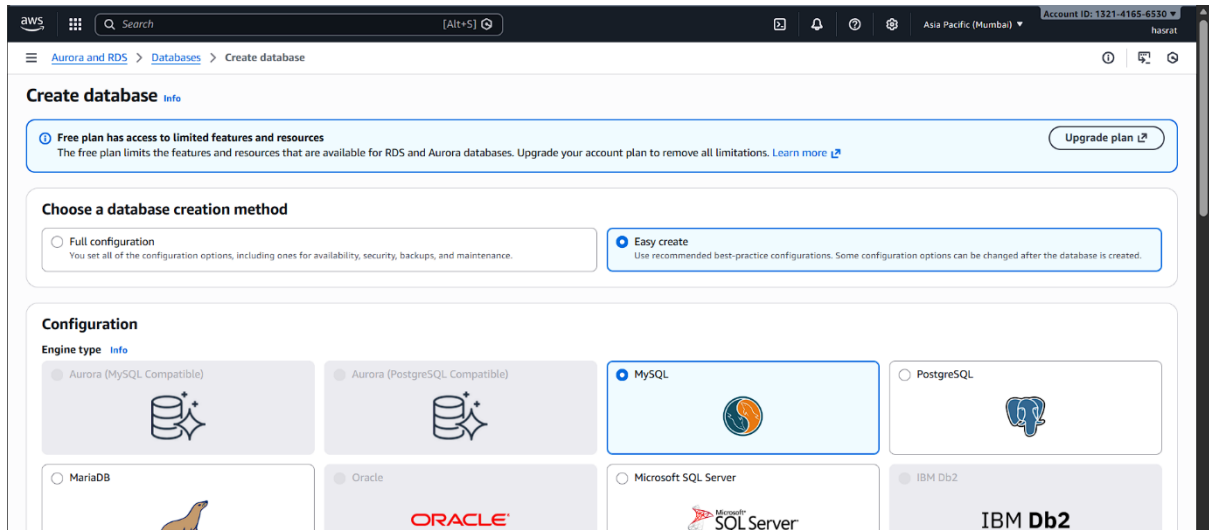


1. Search Aurora and and Create MySQL database



Create database Info

Free plan has access to limited features and resources
The free plan limits the features and resources that are available for RDS and Aurora databases. Upgrade your account plan to remove all limitations. [Learn more](#) Upgrade plan

Choose a database creation method

☐ Full configuration
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.

Configuration

Engine type Info

☐ Aurora (MySQL Compatible)

☐ Aurora (PostgreSQL Compatible)

☒ MySQL

☐ PostgreSQL

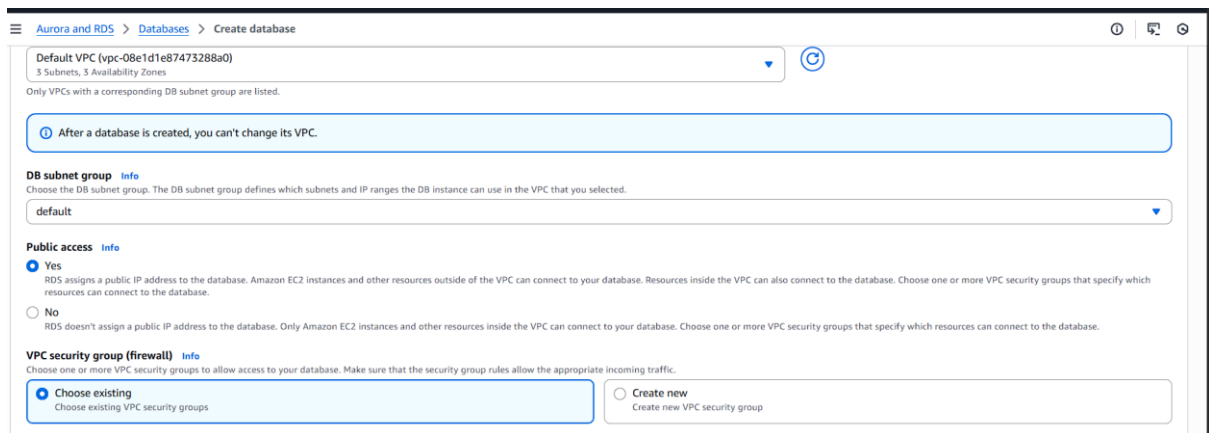
☐ MariaDB

☐ Oracle

☐ Microsoft SQL Server

☐ IBM Db2

2. Select subnet , give public Access and choose VPC security group and create database



Create database

Default VPC (vpc-08e1d1e87473288a0)
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

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.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☒ Choose existing
Choose existing VPC security groups

☐ Create new
Create new VPC security group

3. Database is created:

Creating database hasrat-db
Your database might take a few minutes to launch. You can use settings from hasrat-db to simplify configuration of suggested database add-ons while we finish creating your DB for you. [View connection details](#)

Databases (1) Group resources Modify Actions Create database

Filter by databases

DB identifier	Status	Role	Engine	Upgrade rollout order	Region ...	Size
hasrat-db	Creating	Instance	MySQL Co...	SECOND	-	db.t4g.micro

+ Connecting EC2 with Database

4. Create EC2 instance and connect

Instance summary for i-0ef02e0f8a3e9aa5b (hasrat-web-forDB) [Info](#)

Updated less than a minute ago

[Connect](#) [Instance state](#) [Actions](#)

Instance ID i-0ef02e0f8a3e9aa5b	Public IPv4 address 3.108.53.96 open address	Private IPv4 addresses 172.31.3.15
IPv6 address -	Instance state Running	Public DNS ec2-3-108-53-96.ap-south-1.compute.amazonaws.com open address
Hostname type IP name: ip-172-31-3-15.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-3-15.ap-south-1.compute.internal	Elastic IP addresses -
Answer private resource DNS name IPv4 (A)	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 3.108.53.96 [Public IP]	VPC ID vpc-08e1d1e87473288a0	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0fd08baaf710cf896	Managed false
IMDSv2 Required	Instance ARN arn:aws:ec2:ap-south-1:132141656530:instance/i-0ef02e0f8a3e9aa5b	

5. Install Docker in the Instance

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

ec2-user@ip-172-31-3-15 ~]$ sudo yum install docker -y
amazon Linux 2023 Kernel Livepatch repository
dependencies resolved.
268 kB/s | 29 kB    00:00

Package Architecture Version Repository Size
-----
Installing:
docker x86_64 25.0.13-1.amzn2023.0.2 amazonlinux 46 M
Installing dependencies:
container-selinux noarch 4:2.242.0-1.amzn2023 amazonlinux 58 k
containerd x86_64 2.1.5-1.amzn2023.0.1 amazonlinux 23 M
iptables-libs x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 401 k
iptables-nft x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 183 k
libnftnl x86_64 3.0-1.amzn2023.0.1 amazonlinux 75 k
libnftnl-devel x86_64 1.0.8-2.amzn2023.0.2 amazonlinux 58 k
libnftnl-devel x86_64 1.0.1-19.amzn2023.0.2 amazonlinux 30 k
libnftnl-devel x86_64 1.2.2-2.amzn2023.0.2 amazonlinux 84 k
libnftnl-devel x86_64 2.5-1.amzn2023.0.3 amazonlinux 83 k
libnftnl-devel x86_64 1.3.3-2.amzn2023.0.1 amazonlinux 3.9 M
```

6. Starting docker and checking status

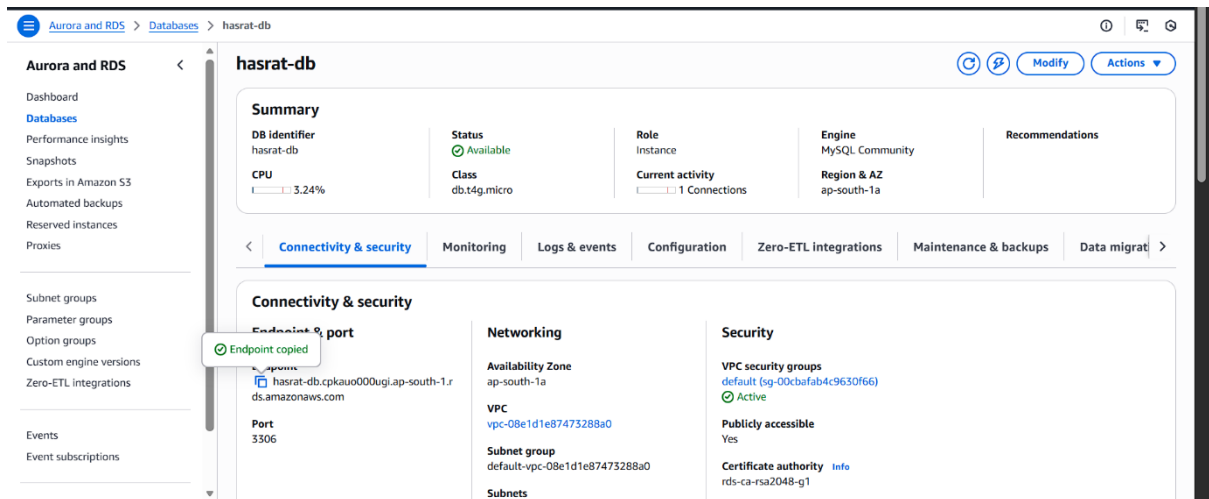
```
Complete!
[ec2-user@ip-172-31-3-15 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-3-15 ~]$ sudo systemctl docker status
Unknown command verb docker.
[ec2-user@ip-172-31-3-15 ~]$ sudo systemctl status docker
* docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; preset: disabled)
   Active: active (running) since Thu 2025-12-18 16:06:04 UTC; 33s ago
     TriggeredBy: * docker.socket
   Docs: https://docs.docker.com
   Process: 27286 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, status=0/SUCCESS)
   Process: 27287 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes.sh (code=exited, status=0/SUCCESS)
   Main PID: 27288 (dockerd)
     Tasks: 9
    Memory: 30.3M
       CPU: 343ms
   CGroup: /system.slice/docker.service
           └─27288 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --default-ulimit nofile=32768:65536

Dec 18 16:06:03 ip-172-31-3-15.ap-south-1.compute.internal systemd[1]: Starting docker.service - Docker Application Container Engine...
Dec 18 16:06:03 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:03.835229201Z" level=info msg="Starting up"
Dec 18 16:06:03 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:03.898887461Z" level=info msg="Loading containers: start."
Dec 18 16:06:04 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:04.387985060Z" level=info msg="Loading containers: done."
Dec 18 16:06:04 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:04.416964656Z" level=info msg="Docker daemon" commit=165516e containerd=
Dec 18 16:06:04 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:04.417145066Z" level=info msg="Daemon has completed initialization"
Dec 18 16:06:04 ip-172-31-3-15.ap-south-1.compute.internal dockerd[27288]: time="2025-12-18T16:06:04.465804775Z" level=info msg="API listen on /run/docker.sock"
Dec 18 16:06:04 ip-172-31-3-15.ap-south-1.compute.internal systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-22/22 (END)
```

7. Pulling Docker Image named : philippaul/node-mysql-app:02 (simple web app for database)

```
[ec2-user@ip-172-31-3-15 ~]$ sudo docker pull philippaul/node-mysql-app:02
02: Pulling from philippaul/node-mysql-app
2ff1d7c41c74: Pull complete
p253aefaeaa7: Pull complete
3d2201bd995c: Pull complete
1de76e268b10: Pull complete
49a8df589451: Pull complete
6f51ee005dea: Pull complete
5f32ed3c3f27: Pull complete
0c8cc2f24a4d: Pull complete
0d27a8e86132: Pull complete
p35ca9a95db0: Pull complete
46a182df3db1: Pull complete
f5b1a7eb997: Pull complete
ff7978b844b1: Pull complete
Digest: sha256:f7c1cfeb42a2f4a40b626b0d03f8b83bbc8ef3f88d0682cd43f395bf9e42966b
Status: Downloaded newer image for philippaul/node-mysql-app:02
docker.io/philippaul/node-mysql-app:02
[ec2-user@ip-172-31-3-15 ~]$ sudo docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
philippaul/node-mysql-app 02 4b941beb4207 13 months ago 923MB
[ec2-user@ip-172-31-3-15 ~]$
```

- Copy endpoint of Database we created in AWS for using as DB_HOST for connecting docker image with database



- Connect the docker image and Database with the following command

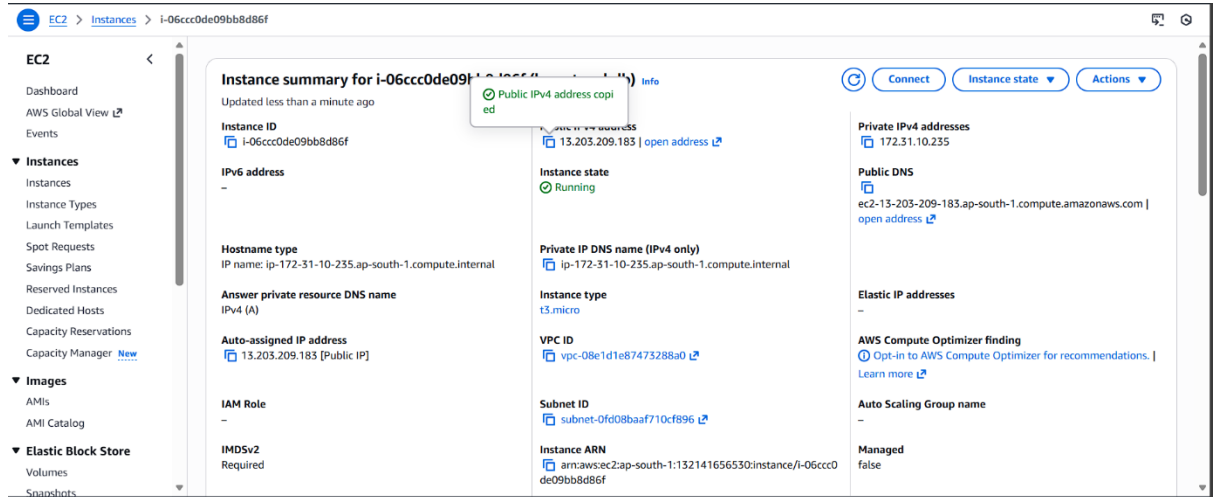
```
[ec2-user@ip-172-31-3-15 ~]$ sudo docker run --rm -p 80:3000 -e DB_HOST="hasrat-db.cpkauo00ugi.ap-south-1.rds.amazonaws.com" -e DB_USER="hasrat-db" -e DB_PASSWORD="hasrat123" -d philippaul/node-mysql-app:02
b6d7c173862a2babe12ceea2ae7a97ee728e026457c4007c52f6fb6c25898dc9
[ec2-user@ip-172-31-3-15 ~]$
```

- Checking status if the web server and database both are running properly

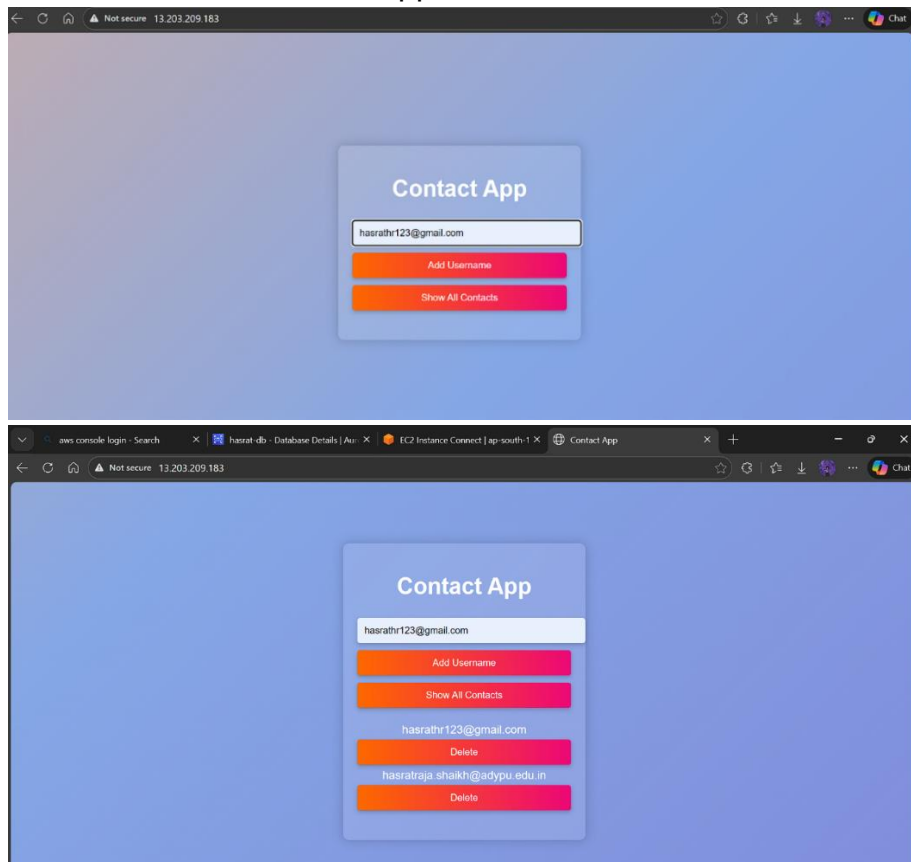
```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Thu Dec 18 17:55:57 2025 from 13.233.177.4
[ec2-user@ip-172-31-10-235 ~]$ sudo docker run --rm -p 80:3000 -e DB_HOST="hasrat-db.cpkauo00ugi.ap-south-1.rds.amazonaws.com" -e DB_USER="admin" -e DB_PASSWORD="hasrat123" -d philippaul/node-mysql-app:02
b6d7c173862a2babe12ceea2ae7a97ee728e026457c4007c52f6fb6c25898dc9
[ec2-user@ip-172-31-10-235 ~]$ sudo docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
b6d7c173862a   philippaul/node-mysql-app:02        "docker-entrypoint.s..." 10 seconds ago Up 8 seconds   0.0.0.0:80->3000/tcp, :::80->3000/tcp interesting_almeida
[ec2-user@ip-172-31-10-235 ~]$ sudo docker logs -f interesting_almeida
Server is running on http://localhost:3000
Database "my_app_db" is ready.
Using database "my_app_db".
Table "contacts" is ready.
```

11. Now copy instance public IP address and paste it on Browser URL to check if the Web app is running



12. Add username in the web app



13. Open mysql to check if the data (User name) is stored in the RDS:

```
[ec2-user@ip-172-31-10-235 ~]$ sudo docker run -it --rm mysql:8.0 mysql -h hasrat-db.cpkau000ugi.ap-south-1.rds.amazonaws.com -u admin -p
Unable to find image 'mysql:8.0' locally
8.0: Pulling from library/mysql
7a5e917526: Pull complete
a3c731ffda3: Pull complete
98c09a342d40: Pull complete
6b8812fa3131: Pull complete
9e662d77f5a0: Pull complete
4f1ef7d4d7cb: Pull complete
3a7bd505aaa0: Pull complete
907afe780e24: Pull complete
8c3cd0b3401c: Pull complete
5f838d8bc363: Pull complete
0d4ab5f772d0: Pull complete
Digest: sha256:0275a35e79c60caae68fac520602d9f6897feb9b0941a1471196b1a01760e581
Status: Downloaded newer image for mysql:8.0
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 65
Server version: 8.0.43 Source distribution

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

14. We see the Data is stored successfully

```
aws [Alt+5] Ask Amazon Q Account ID: 1321-4165-6530 hasrat

+-----+
| Database |
+-----+
| information_schema |
| my_app_db |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)

mysql> use my_app_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> use my_app_db;
Database changed
mysql> show tables;
+-----+
| Tables in my_app_db |
+-----+
| contacts |
+-----+
1 row in set (0.00 sec)

mysql> select * from contacts;
+-----+
| id | username |
+-----+
| 1 | hasrathr123@gmail.com |
| 2 | hasratraja.shaiikh@adypu.edu.in |
+-----+
2 rows in set (0.00 sec)

mysql>
```

X

