Shaikh Ateeb Ahmed
29-08-2025

Amazon RDS

Step 1: Create the VPC & subnets

1. Open the VPC console

1. Sign in to the AWS Management Console.

2. From the Services menu search for **VPC** and click **VPC**.

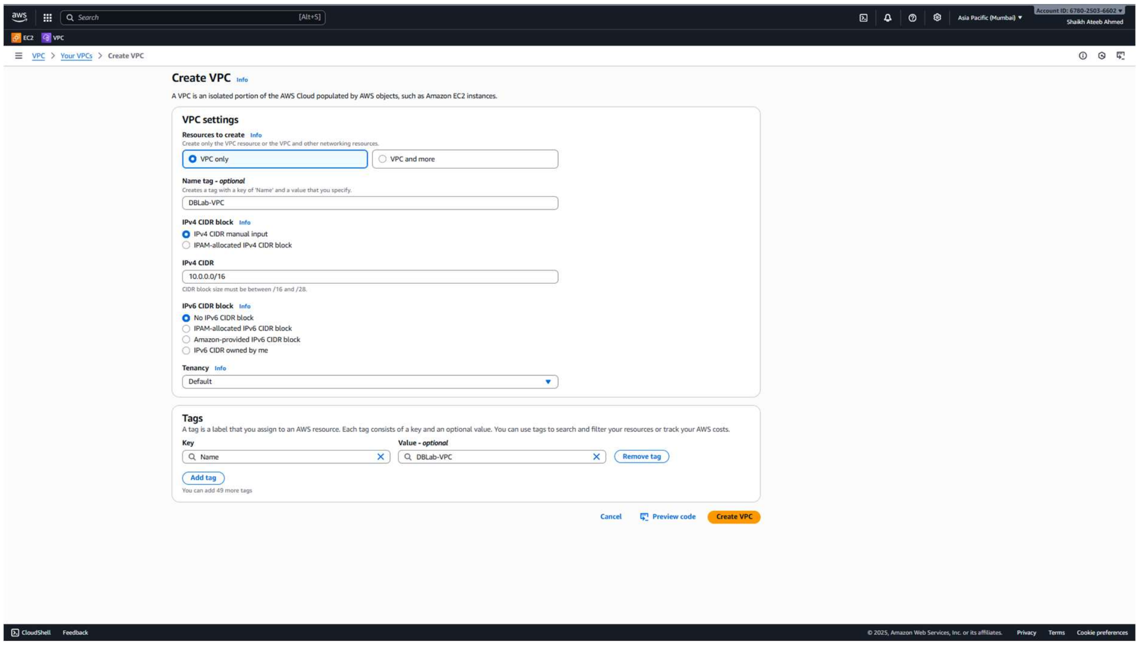
2. Create the VPC

1. In the left menu click **Your VPCs** → **Create VPC** (button).

2. Choose **Create VPC** (wizard with single VPC option).

3. Fill the form:

- **Name tag**: DBLab-VPC
- **IPv4 CIDR block**: 10.0.0.0/16
- **IPv6 CIDR block**: leave as **No IPv6 CIDR block** (unless you need IPv6)
- **Tenancy**: Default



Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
● VPC only ○ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
DBLab-VPC

IPv4 CIDR block Info
● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR
10.0.0.0/16
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
● No IPv6 CIDR block
○ IPAM-allocated IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block
○ IPv6 CIDR owned by me

Tenancy Info
Default

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	DBLab-VPC	Remove tag

Add tag
You can add 49 more tags

Cancel Preview code Create VPC

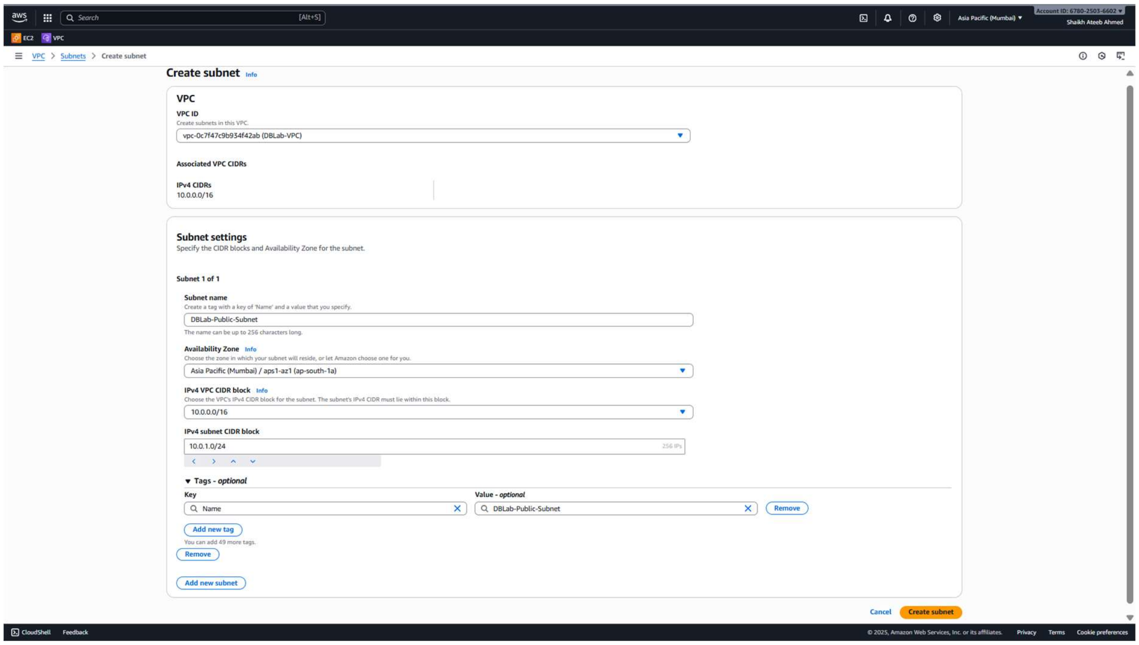
4. Click **Create VPC** (wait for success notification).

3. Create the public subnet

1. Left menu → **Subnets** → **Create subnet**.

2. Configure:

- **Name tag**: DBLab-Public-Subnet
- **VPC**: select DBLab-VPC
- **Availability Zone**: pick an AZ (e.g., us-east-1a) — note which AZ for later.
- **IPv4 CIDR block**: 10.0.1.0/24



Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-0c7f47c9b934f42ab (DBLab-VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
DBLab-Public-Subnet
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
Asia Pacific (Mumbai) / aps1-az1 (ap-south-1a)

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24 256 IPs

▾ Tags - *optional*

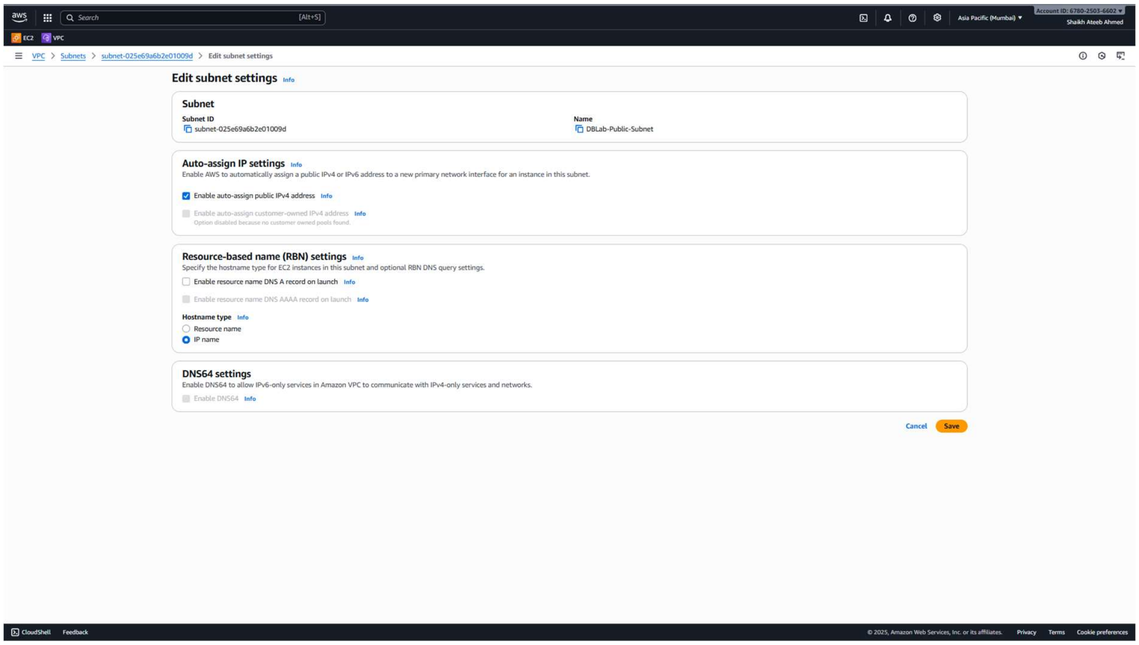
Key	Value - *optional*
Name	DBLab-Public-Subnet

3. Click **Create subnet**.

1. In **Subnets**, select DBLab-Public-Subnet.

2. Actions → **Edit subnet settings** → check **Enable auto-assign public IPv4 address** → **Save**. This ensures instances launched in this subnet can receive public IPs.

Shaikh Ateeb Ahmed
29-08-2025



aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
VPC > Subnets > subnet-025e69a6b2e01009d > Edit subnet settings

Edit subnet settings Info

Subnet

Subnet ID: subnet-025e69a6b2e01009d

Name: DBLab-Public-Subnet

Auto-assign IP settings Info

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

☑ Enable auto-assign public IPv4 address Info

□ Enable auto-assign customer-owned IPv4 address Info
Option disabled because no customer owned pools found.

Resource-based name (RBN) settings Info

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

□ Enable resource name DNS A record on launch Info

□ Enable resource name DNS AAAA record on launch Info

Hostname type Info
○ Resource name
● IP name

DNS64 settings

Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

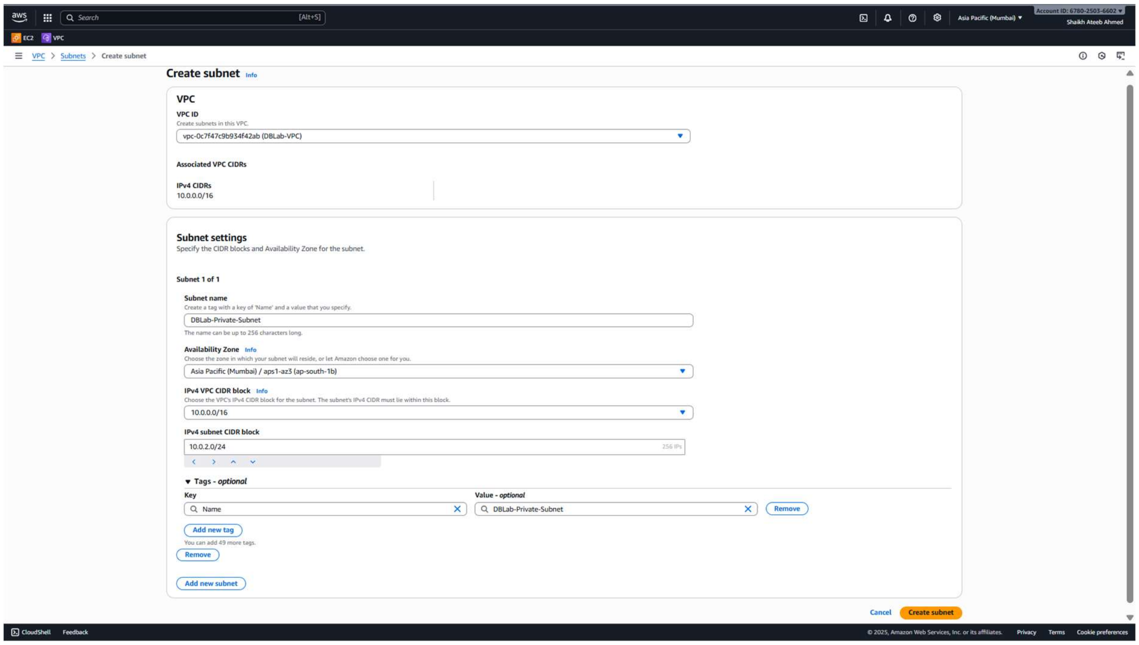
□ Enable DNS64 Info

4. Create the private subnet

1. Left menu → **Subnets** → **Create subnet** again.
2. Configure:

- **Name tag**: DBLab-Private-Subnet
- **VPC**: DBLab-VPC
- **Availability Zone**: either same AZ as public or another (e.g., us-east-1b) — good to separate for resilience.
- **IPv4 CIDR block**: 10.0.2.0/24

Shaikh Ateeb Ahmed
29-08-2025



Create subnet

VPC

VPC ID
Create subnets in this VPC.
vpc-0c7f47c9b934f42ab (DBLab-VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
DBLab-Private-Subnet
The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
Asia Pacific (Mumbai) / aps1-az3 (ap-south-1b)

IPv4 VPC CIDR block
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

Tags - *optional*

Key	Value - *optional*
Name	DBLab-Private-Subnet

You can add 49 more tags.

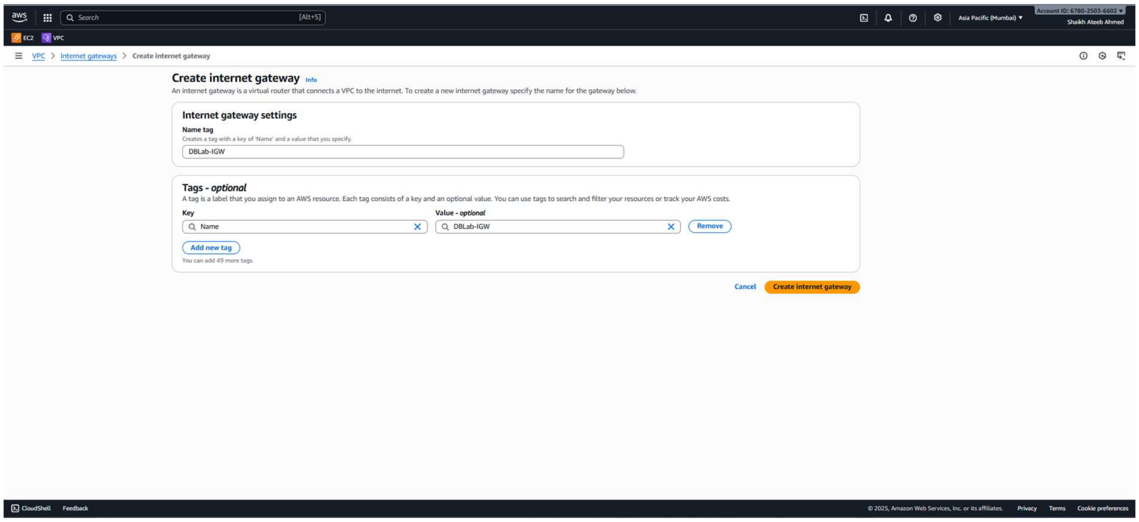
3. Click **Create subnet**.

4. Ensure **auto-assign public IPv4** is **disabled** (default). Do not enable—this keeps it private.

5. Create and attach an Internet Gateway (IGW)

1. Left menu → **Internet Gateways** → **Create internet gateway**.

2. **Name tag**: DBLab-IGW → **Create internet gateway**.



Create internet gateway
An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

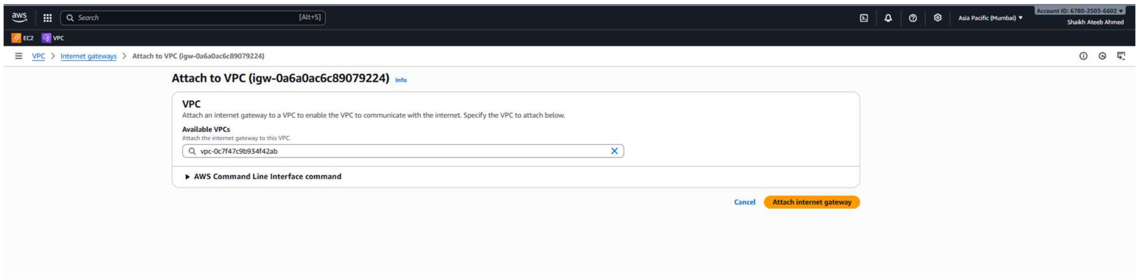
Name tag
Creates a tag with a key of 'Name' and a value that you specify.
DBLab-IGW

Tags - *optional*
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	DBLab-IGW

You can add 49 more tags.

3. Select the new IGW → **Actions** → **Attach to VPC** → choose DBLab-VPC → **Attach internet gateway**.



VPC > Internet gateways > Attach to VPC (igw-0a6a0ac6c89079224)

Attach to VPC (igw-0a6a0ac6c89079224) Info

VPC

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs

Attach the internet gateway to this VPC.

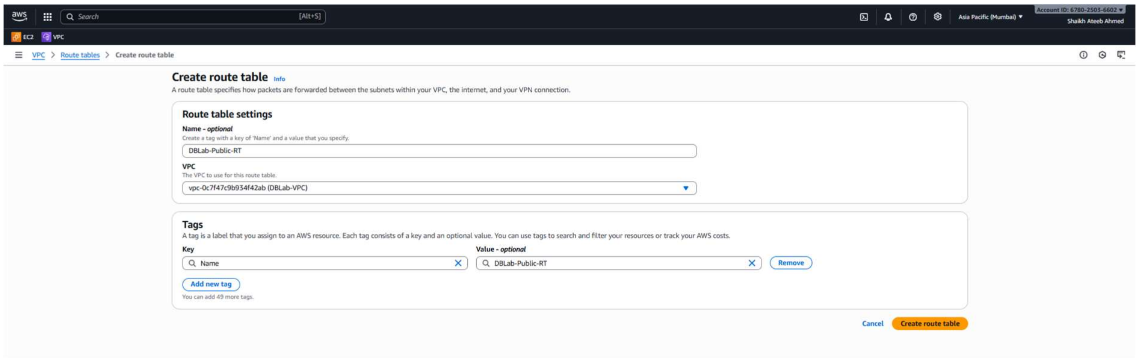
vpc-0c7f47c9b934f42ab

▸ AWS Command Line Interface command

Cancel | Attach internet gateway

6. Create a route table for the public subnet

1. Left menu → **Route tables** → **Create route table**.
2. **Name tag**: DBLab-Public-RT
   VPC: DBLab-VPC → **Create route table**.



VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - *optional*

Create a tag with a key of 'Name' and a value that you specify.

DBLab-Public-RT

VPC

The VPC to use for this route table.

vpc-0c7f47c9b934f42ab (DBLab-VPC)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	DBLab-Public-RT	Remove

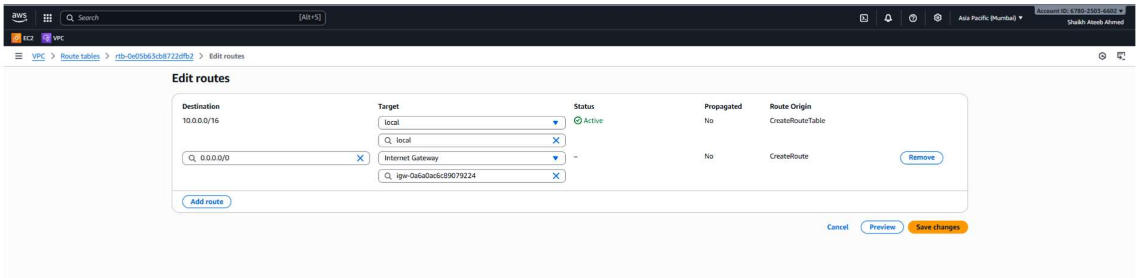
Add new tag

You can add 49 more tags.

Cancel | Create route table

3. Select DBLab-Public-RT in the list → **Routes** tab → **Edit routes** → **Add route**:

- **Destination**: 0.0.0.0/0
- **Target**: select the internet gateway and select your IGW (igw-0a6a0ac6c89079224) → **Save routes**.



VPC > Route tables > rtb-0e05b63cb8722dfb2 > Edit routes

Edit routes

Destination	Target	Status	Propagated	Route Origin	
10.0.0.0/16	local local	Active	No	CreateRouteTable	
0.0.0.0/0	Internet Gateway igw-0a6a0ac6c89079224	-	No	CreateRoute	Remove

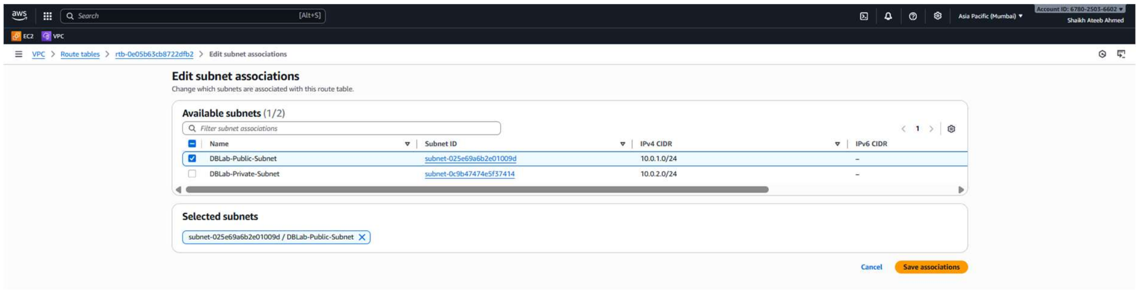
Add route

Cancel | Preview | Save changes

7. Associate the public subnet with the public route table

1. Still on the DBLab-Public-RT → **Subnet associations** tab → **Edit subnet associations**.
2. Check DBLab-Public-Subnet → **Save**.

Shaikh Ateeb Ahmed
29-08-2025



VPC > Route tables > rtb-0e05b63cb8722dfb2 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)

Filter subnet associations

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
☑	DBLab-Public-Subnet	subnet-025e69a6b2e01009d	10.0.1.0/24	–
□	DBLab-Private-Subnet	subnet-0c9b47474e5f37414	10.0.2.0/24	–

Selected subnets

subnet-025e69a6b2e01009d / DBLab-Public-Subnet

Result: any instance in DBLab-Public-Subnet with a public IP will have internet access through the IGW.

Step 2: Launch a Bastion EC2 Instance

1. Sign in to the AWS Console → Services → **EC2** → **Instances** → **Launch instances**.
2. **Choose an AMI**

   o Select **Amazon Linux 2 AMI (HVM)**.

3. **Choose an instance type**

   o Pick **t3.micro** (Free tier eligible) → **Next: Configure instance details**.

4. **Configure instance details**

   o **Network (VPC)**: choose DBLab-VPC (the VPC you created).

   o **Subnet**: choose DBLab-Public-Subnet.

   o **Auto-assign Public IP**: **Enable** (or Ensure "Enable" is selected).

   o Leave other defaults (unless you need IAM role, user data, etc.) → **Next: Add storage**.

5. **Add storage**

   o Default 8 GB is fine for a bastion → **Next: Add tags**.

6. **Add tags**

   o Name = DBLab-Bastion (helps identify instance) → **Next: Configure security group**.

7. **Configure Security Group**

   o Choose **Create a new security group** or select an existing Bastion-SG.

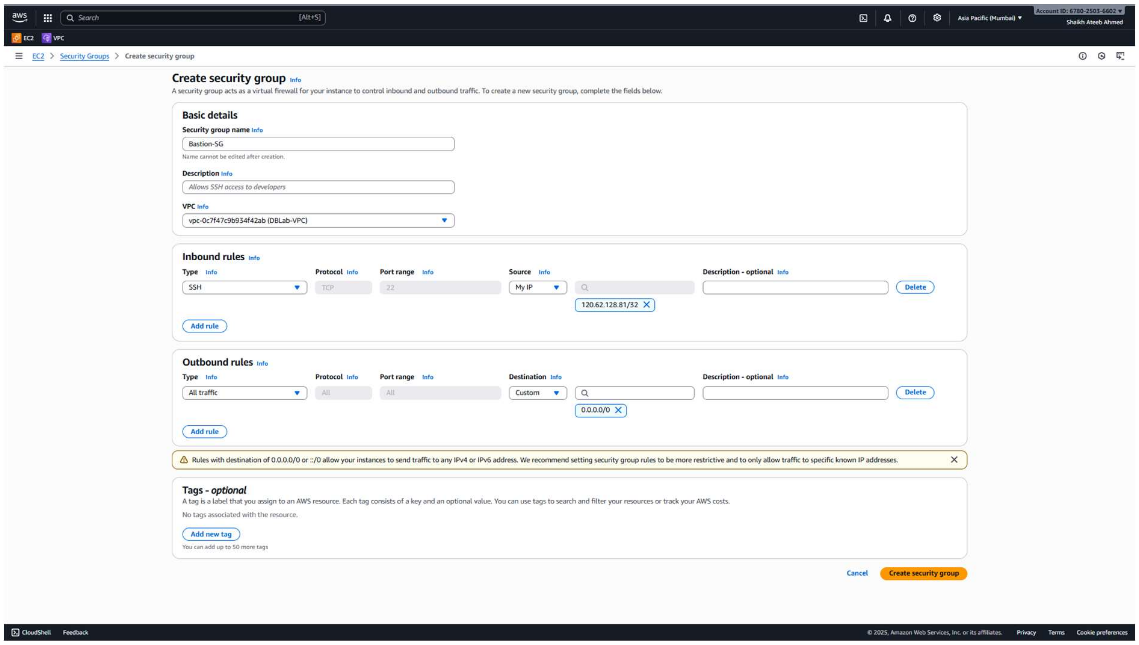
   o Inbound rule:

   ▪ Type: **SSH**

   ▪ Protocol: TCP

- Port range: **22**
- Source: **My IP** (recommended). This will put your current public IP (e.g., 203.0.113.5/32) — do **not** use 0.0.0.0/0.

- Outbound: leave default (allow all) → **Review and Launch**.



Create security group

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name: Bastion-SG

Name cannot be edited after creation.

Description: Allows SSH access to developers

VPC: vpc-0c7f47c9b934f42ab (DBLab-VPC)

Inbound rules

Type	Protocol	Port range	Source		Description - optional
SSH	TCP	22	My IP	120.62.128.81/32	

Outbound rules

Type	Protocol	Port range	Destination		Description - optional
All traffic	All	All	Custom	0.0.0.0/0	

⚠ Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.

Tags - *optional*

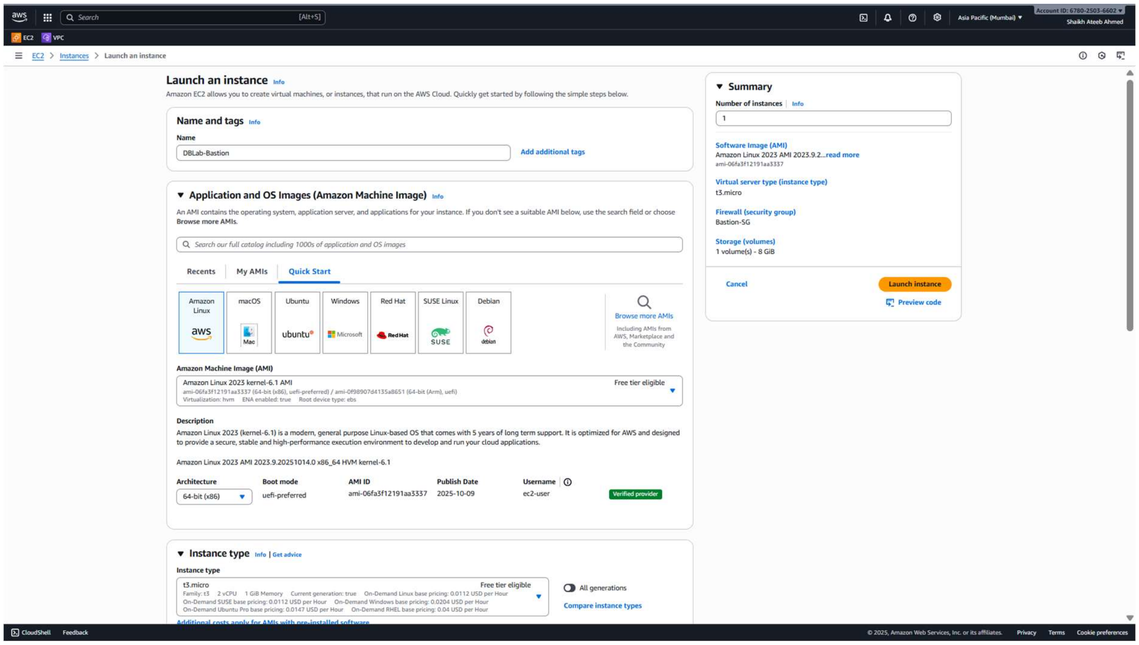
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

You can add up to 50 more tags

8. **Key pair**

- Choose an existing key pair you own.



Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: DBLab-Bastion

Application and OS Images (Amazon Machine Image)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose Browse more AMIs.

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI — Free tier eligible
ami-06fa3f12191aa3337 (64-bit (x86), uefi-preferred) / ami-0f98907d4135a8651 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.9.20251014.0 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	Publish Date	Username
64-bit (x86)	uefi-preferred	ami-06fa3f12191aa3337	2025-10-09	ec2-user

Instance type

Instance type: t3.micro — Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand Linux base pricing: 0.0112 USD per Hour
On-Demand SUSE base pricing: 0.0112 USD per Hour On-Demand Windows base pricing: 0.0204 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0147 USD per Hour On-Demand RHEL base pricing: 0.04 USD per Hour

Additional costs apply for AMIs with pre-installed software

Summary

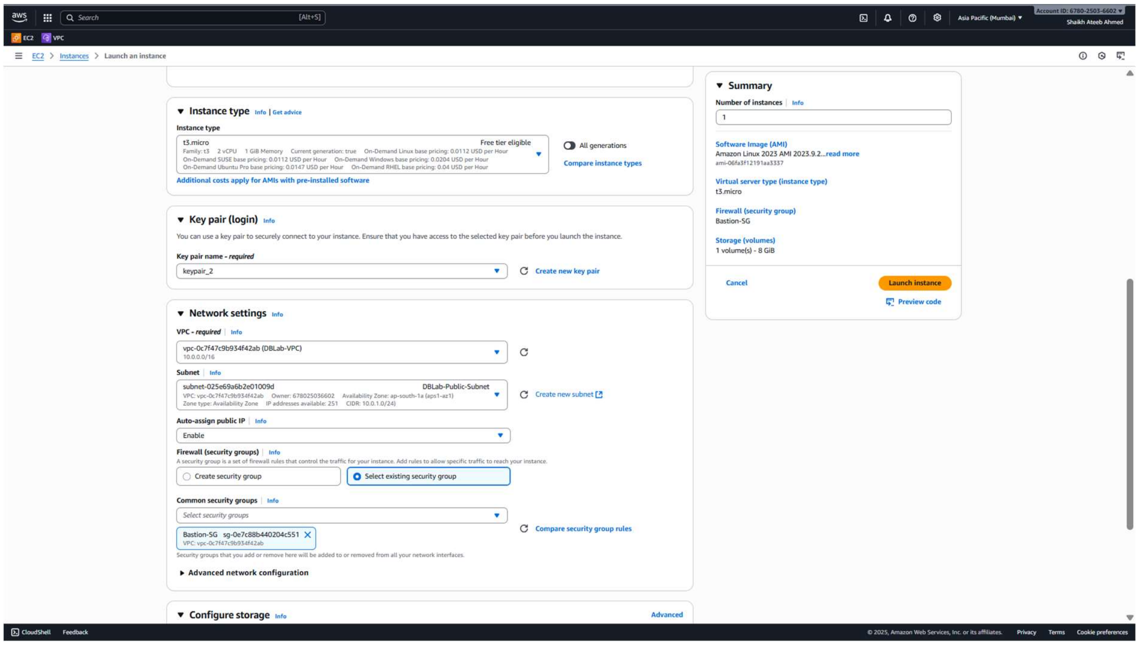
Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.9.2...read more
ami-06fa3f12191aa3337

Virtual server type (instance type): t3.micro

Firewall (security group): Bastion-SG

Storage (volumes): 1 volume(s) - 8 GiB



▼ Instance type

Instance type
t3.micro Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand Linux base pricing: 0.0112 USD per Hour
On-Demand SUSE base pricing: 0.0112 USD per Hour On-Demand Windows base pricing: 0.0204 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0147 USD per Hour On-Demand RHEL base pricing: 0.04 USD per Hour

▼ Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*
keypair_2

▼ Network settings

VPC - *required*
vpc-0c7f47c9b934f42ab (DBLab-VPC)
10.0.0.0/16

Subnet
subnet-025e69a6b2e01009d DBLab-Public-Subnet
VPC: vpc-0c7f47c9b934f42ab Owner: 678025036602 Availability Zone: ap-south-1a (aps1-az1)
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24)

Auto-assign public IP
Enable

Firewall (security groups)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
○ Create security group ● Select existing security group

Common security groups
Select security groups
Bastion-SG sg-0e7c88b440204c551
VPC: vpc-0c7f47c9b934f42ab
Security groups that you add or remove here will be added to or removed from all your network interfaces.

▶ Advanced network configuration

▼ Configure storage

▼ Summary

Number of instances
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.9.2...read more
ami-06fa3f12191aa3337

Virtual server type (instance type)
t3.micro

Firewall (security group)
Bastion-SG

Storage (volumes)
1 volume(s) - 8 GiB

- Click **Launch instances**.
- Click **View Instances** to return to the Instances screen and wait until the instance state is **running** and status checks are OK.

9. Note the instance's **Public IPv4** or **Public DNS** (you'll use it to SSH).

SSH from Windows using PuTTY (if you use .pem → .ppk)

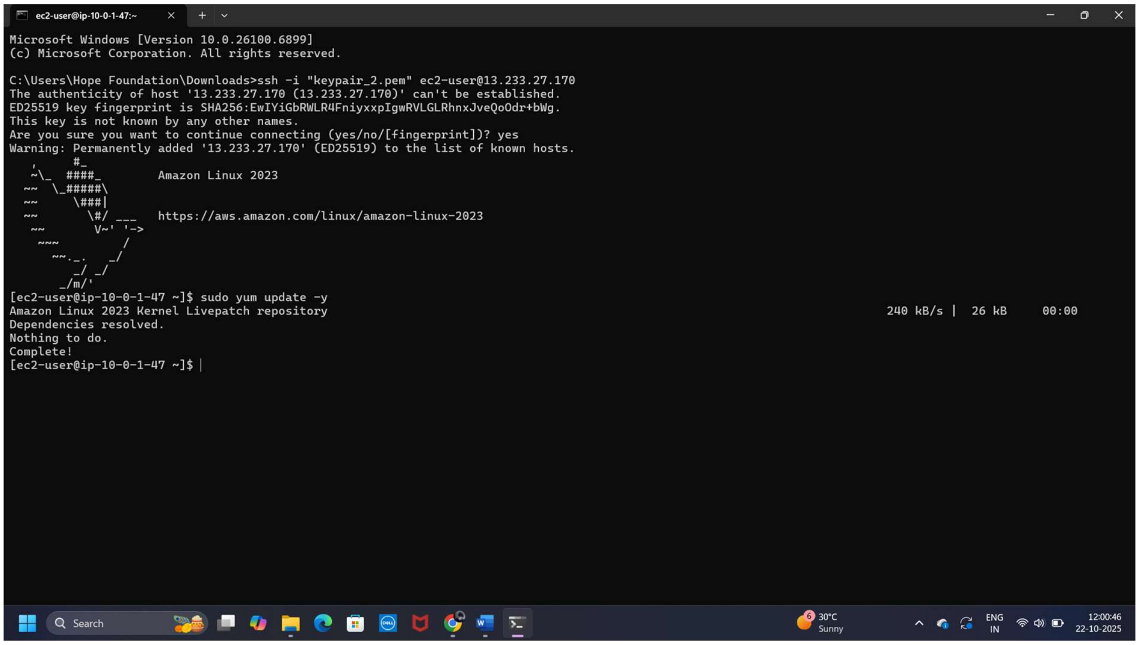
Install MySQL and PostgreSQL clients (on bastion host)

After SSH'ing into the bastion, run the commands for your AMI.

Amazon Linux 2 (yum)

# update OS and install clients

sudo yum update -y



```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hope Foundation\Downloads>ssh -i "keypair_2.pem" ec2-user@13.233.27.170
The authenticity of host '13.233.27.170 (13.233.27.170)' can't be established.
ED25519 key fingerprint is SHA256:EwIYiGbRWLR4FniyxxpIgwRVLGLRhnxJveQoOdr+bWg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.233.27.170' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-1-47 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository                                                      240 kB/s |  26 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-1-47 ~]$ |
```

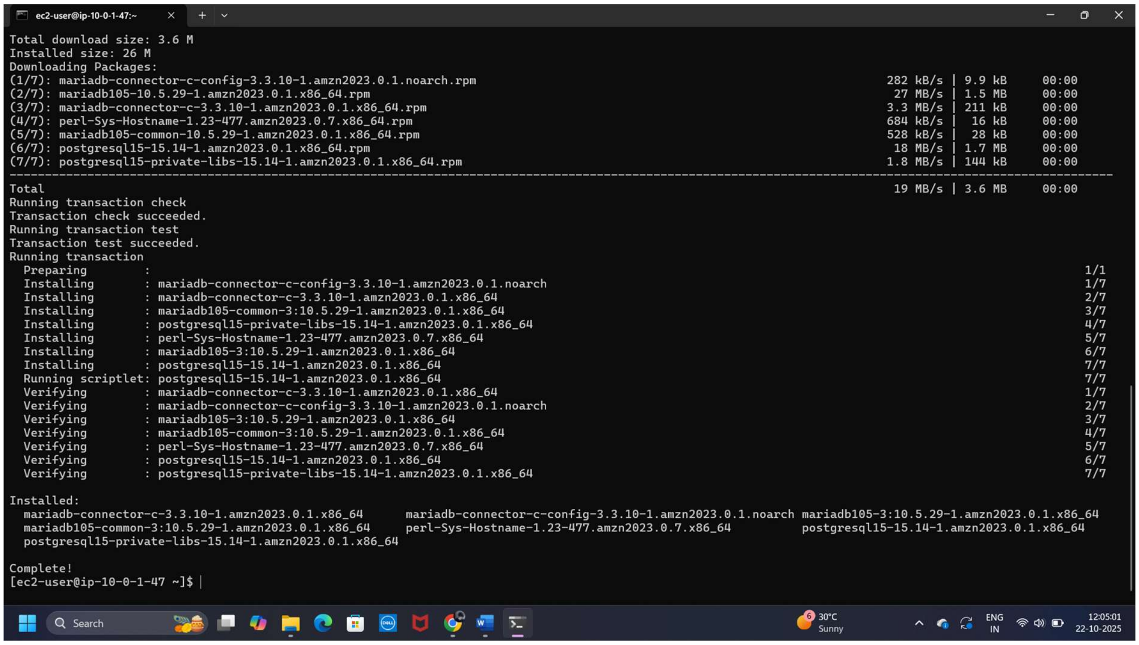
# Update package index

sudo yum update -y

sudo dnf install -y mariadb105 postgresql15

This will install:

- mysql client (via MariaDB 10.5 client)
- psql client (PostgreSQL 15)



```
Total download size: 3.6 M
Installed size: 26 M
Downloading Packages:
(1/7): mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch.rpm                                 282 kB/s | 9.9 kB     00:00
(2/7): mariadb105-10.5.29-1.amzn2023.0.1.x86_64.rpm                                                 27 MB/s | 1.5 MB     00:00
(3/7): mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64.rpm                                        3.3 MB/s | 211 kB     00:00
(4/7): perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64.rpm                                          684 kB/s |  16 kB     00:00
(5/7): mariadb105-common-10.5.29-1.amzn2023.0.1.x86_64.rpm                                         528 kB/s |  28 kB     00:00
(6/7): postgresql15-15.14-1.amzn2023.0.1.x86_64.rpm                                                 18 MB/s | 1.7 MB     00:00
(7/7): postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64.rpm                                   1.8 MB/s | 144 kB     00:00
---------------------------------------------------------------------------------------------------------------------------------
Total                                                                                               19 MB/s | 3.6 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                                          1/1
  Installing       : mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch                                                  1/7
  Installing       : mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64                                                         2/7
  Installing       : mariadb105-common-3:10.5.29-1.amzn2023.0.1.x86_64                                                        3/7
  Installing       : postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64                                                    4/7
  Installing       : perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64                                                           5/7
  Installing       : mariadb105-3:10.5.29-1.amzn2023.0.1.x86_64                                                               6/7
  Installing       : postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                 7/7
  Running scriptlet: postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                 7/7
  Verifying        : mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64                                                         1/7
  Verifying        : mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch                                                  2/7
  Verifying        : mariadb105-3:10.5.29-1.amzn2023.0.1.x86_64                                                               3/7
  Verifying        : mariadb105-common-3:10.5.29-1.amzn2023.0.1.x86_64                                                        4/7
  Verifying        : perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64                                                           5/7
  Verifying        : postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                 6/7
  Verifying        : postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64                                                    7/7

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          postgresql15-15.14-1.amzn2023.0.1.x86_64
  postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-10-0-1-47 ~]$ |
```

Once installed, confirm both clients work:

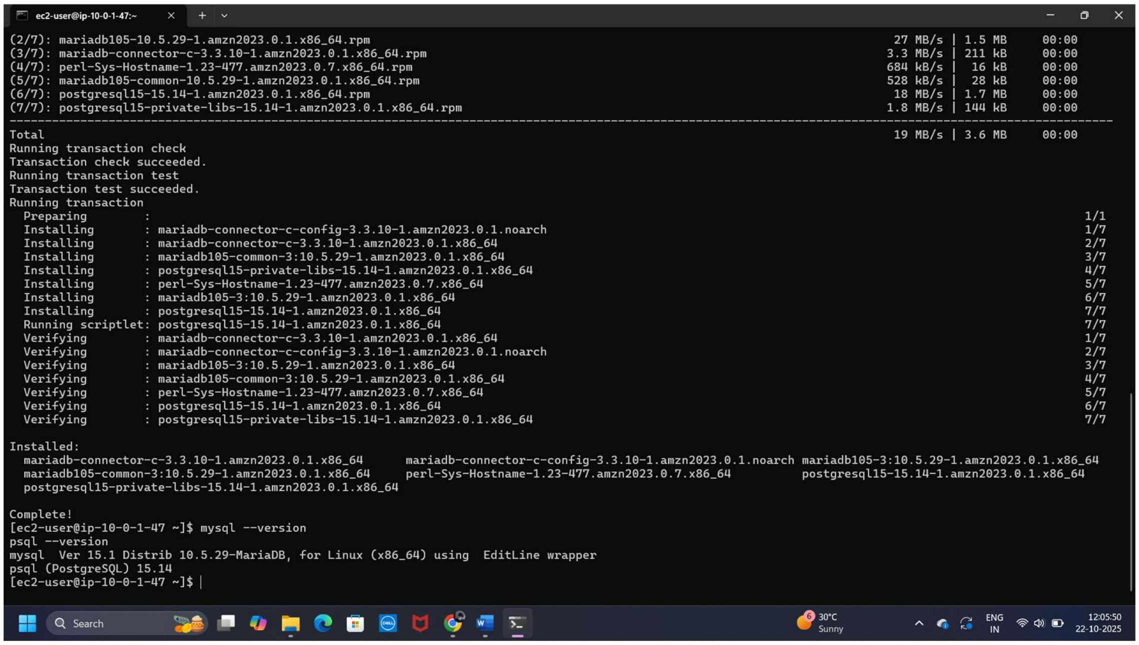
mysql --version

psql --version

Expected output examples:

mysql Ver 15.1 Distrib 10.5.16-MariaDB, for Linux (x86_64)

psql (PostgreSQL) 15.5



```
(2/7): mariadb105-10.5.29-1.amzn2023.0.1.x86_64.rpm                                                                          27 MB/s | 1.5 MB     00:00
(3/7): mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64.rpm                                                                 3.3 MB/s | 211 kB     00:00
(4/7): perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64.rpm                                                                   684 kB/s |  16 kB     00:00
(5/7): mariadb105-common-10.5.29-1.amzn2023.0.1.x86_64.rpm                                                                  528 kB/s |  28 kB     00:00
(6/7): postgresql15-15.14-1.amzn2023.0.1.x86_64.rpm                                                                          18 MB/s | 1.7 MB     00:00
(7/7): postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64.rpm                                                            1.8 MB/s | 144 kB     00:00
------------------------------------------------------------------------------------------------------------------------------------------------------------
Total                                                                                                                        19 MB/s | 3.6 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                                                                    1/1
  Installing       : mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch                                                                            1/7
  Installing       : mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64                                                                                   2/7
  Installing       : mariadb105-common-3:10.5.29-1.amzn2023.0.1.x86_64                                                                                  3/7
  Installing       : postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64                                                                              4/7
  Installing       : perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64                                                                                     5/7
  Installing       : mariadb105-3:10.5.29-1.amzn2023.0.1.x86_64                                                                                         6/7
  Installing       : postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                                           7/7
  Running scriptlet: postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                                           7/7
  Verifying        : mariadb-connector-c-3.3.10-1.amzn2023.0.1.x86_64                                                                                   1/7
  Verifying        : mariadb-connector-c-config-3.3.10-1.amzn2023.0.1.noarch                                                                            2/7
  Verifying        : mariadb105-3:10.5.29-1.amzn2023.0.1.x86_64                                                                                         3/7
  Verifying        : mariadb105-common-3:10.5.29-1.amzn2023.0.1.x86_64                                                                                  4/7
  Verifying        : perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64                                                                                     5/7
  Verifying        : postgresql15-15.14-1.amzn2023.0.1.x86_64                                                                                           6/7
  Verifying        : postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64                                                                              7/7

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          postgresql15-15.14-1.amzn2023.0.1.x86_64
  postgresql15-private-libs-15.14-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-10-0-1-47 ~]$ mysql --version
psql --version
mysql  Ver 15.1 Distrib 10.5.29-MariaDB, for Linux (x86_64) using  EditLine wrapper
psql (PostgreSQL) 15.14
[ec2-user@ip-10-0-1-47 ~]$ |
```

Step 3: Create Amazon RDS Instance

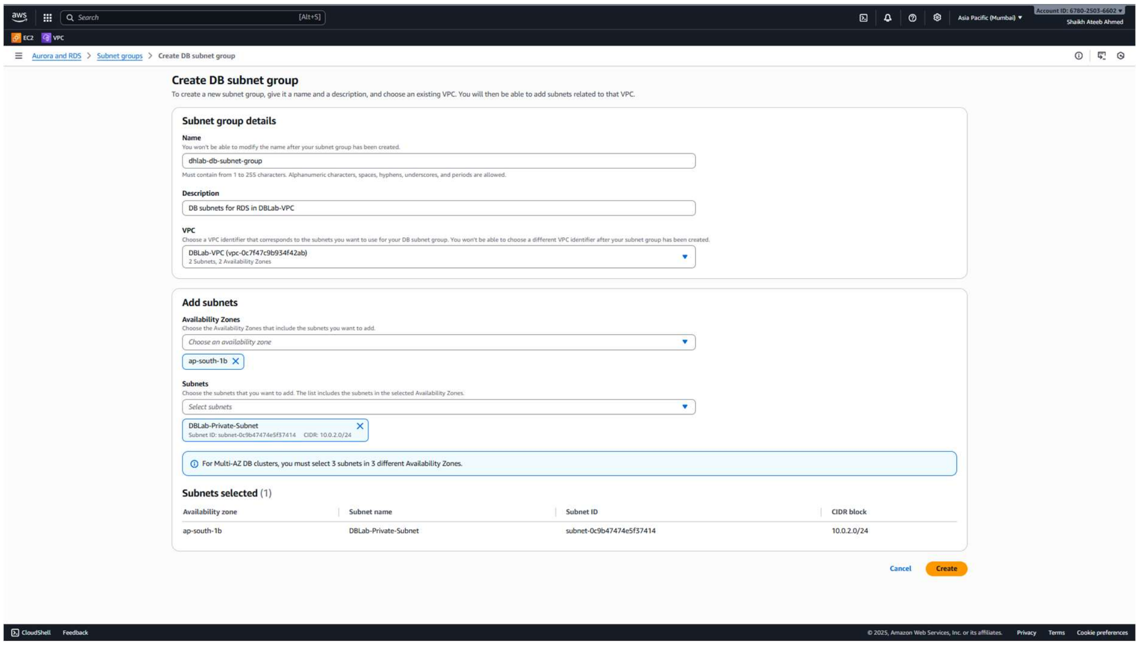
Create a DB Subnet Group

RDS instances in private subnets must use a DB subnet group.

1. Open **AWS Console → RDS**.
2. In the left menu, click **Subnet groups** (or **DB subnet groups**) → **Create DB subnet group**.
3. Fill the form:

- **Name**: dhlab-db-subnet-group
- **Description**: DB subnets for RDS in DBLab-VPC
- **VPC**: select DBLab-VPC

4. **Add subnets**: choose the **private subnet(s)** you created — DBLab-Private-Subnet (for multi-AZ add private subnets in different AZs).



aws
Search [Alt+S]
EC2
VPC
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
Aurora and RDS > Subnet groups > Create DB subnet group

Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.
dhlab-db-subnet-group
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description
DB subnets for RDS in DBLab-VPC

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.
DBLab-VPC (vpc-0c7f47c9b934f42ab)
2 Subnets, 2 Availability Zones

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone
ap-south-1b ✕

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets
DBLab-Private-Subnet ✕
Subnet ID: subnet-0c9b47474e5f37414 CIDR: 10.0.2.0/24

ⓘ For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

Subnets selected (1)

Availability zone	Subnet name	Subnet ID	CIDR block
ap-south-1b	DBLab-Private-Subnet	subnet-0c9b47474e5f37414	10.0.2.0/24

5. Click **Create DB subnet group**.

6. Confirm it appears in the list and its status is **available**.

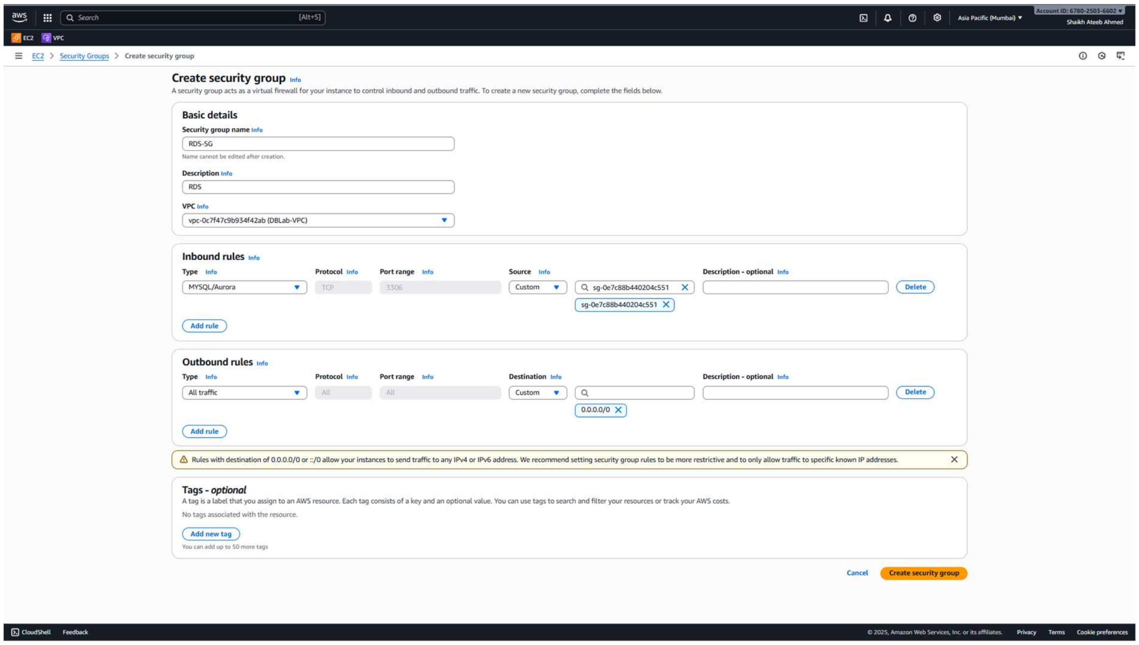
Create an RDS Security Group

You already planned RDS-SG to allow DB access only from Bastion. If not created, do it now in EC2 → Security Groups.

1. **EC2 Console → Network & Security → Security Groups → Create security group**.

- **Name**: RDS-SG
- **VPC**: DBLab-VPC
- **Inbound rule**:
  - Type: **MySQL/Aurora** (or **Postgres** if PostgreSQL)
  - Protocol: TCP
  - Port: **3306** (MySQL)
  - Source: **Custom** → choose the **Bastion-SG** (not a CIDR). This references the bastion security group so only instances using that SG can connect.
- Outbound: leave default (allow all).

Shaikh Ateeb Ahmed
29-08-2025



Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name: RDS-SG

Name cannot be edited after creation.

Description: RDS

VPC: vpc-0c7f47c9b934f42ab (DBLab-VPC)

Inbound rules

Type	Protocol	Port range	Source		Description - optional
MYSQL/Aurora	TCP	3306	Custom	sg-0e7c88b440204c551 sg-0e7c88b440204c551	

Outbound rules

Type	Protocol	Port range	Destination		Description - optional
All traffic	All	All	Custom	0.0.0.0/0	

⚠ Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

You can add up to 50 more tags

2. Create and note the security group ID.

Create the RDS Database (MySQL example)

1. In **RDS Console → Databases** → **Create database**.
2. Choose **Standard Create** (gives full options).
3. **Engine**: select **MySQL** (or PostgreSQL if you prefer).
4. **Templates**: choose **Free tier** (if eligible) — this preselects small sizes.
5. **Settings**:

- **DB instance identifier**: rds-mysql-lab
- **Master username**: e.g., admin (store this)
- **Master password**: enter a strong password and confirm (ateeb6867)

6. **DB instance size**:

- **DB instance class**: select db.t3.micro (Free tier / small lab)

7. **Storage**:

- **Storage type**: General Purpose (SSD)
- **Allocated storage**: 20 GiB

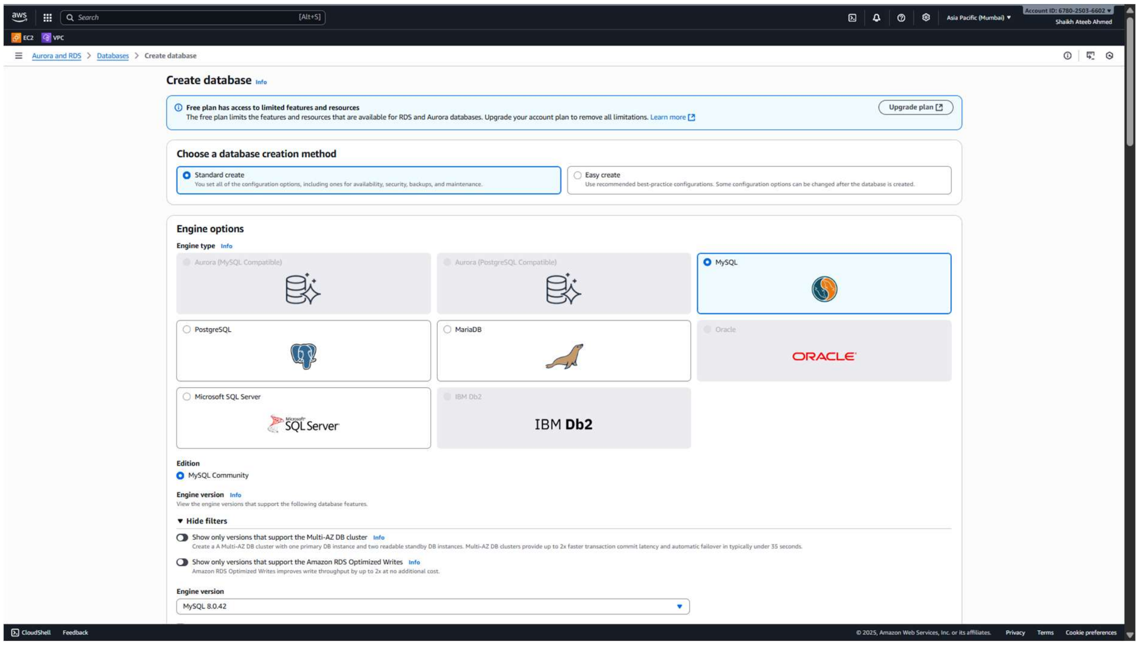
8. **Connectivity** (this is critical):

- **Virtual private cloud (VPC)**: select DBLab-VPC

- **Subnet group**: select dhlab-db-subnet-group (the DB Subnet Group you created)
- **Public access**: **No** (Important — keep DB private)
- **VPC security groups**: choose **RDS-SG** (the SG that allows traffic only from the Bastion-SG)
- **Availability zone**: leave as No preference (or choose an AZ). For production, consider Multi-AZ.
- **Connectivity method**: defaults are fine for a lab.

9. **Database authentication**:

- Choose **Password authentication** (default). (You can enable IAM DB authentication if desired.)


aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
Aurora and RDS > Databases > Create 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

● Standard create
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.

Engine options

Engine type Info

○ Aurora (MySQL Compatible)
○ Aurora (PostgreSQL Compatible)
● MySQL
○ PostgreSQL
○ MariaDB
○ Oracle
○ Microsoft SQL Server
○ IBM Db2

Edition
● MySQL Community

Engine version Info
View the engine versions that support the following database features.

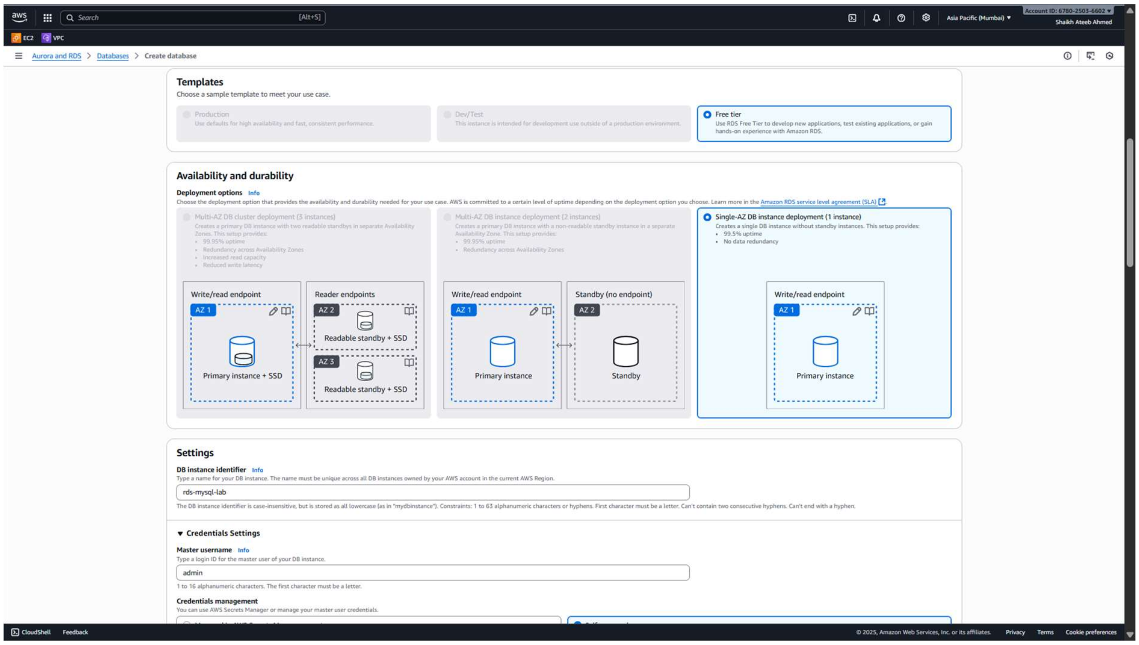
▼ Hide filters

Show only versions that support the Multi-AZ DB cluster Info
Create a A Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.

Show only versions that support the Amazon RDS Optimized Writes Info
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Engine version
MySQL 8.0.42 ▾

Shaikh Ateeb Ahmed
29-08-2025



Templates
Choose a sample template to meet your use case.

○ Production
Use defaults for high availability and fast, consistent performance.

○ Dev/Test
This instance is intended for development use outside of a production environment.

● Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

Availability and durability

Deployment options Info
Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the Amazon RDS service level agreement (SLA)

○ Multi-AZ DB cluster deployment (3 instances)
Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:
- 99.95% uptime
- Redundancy across Availability Zones
- Increased read capacity
- Reduced write latency

○ Multi-AZ DB instance deployment (2 instances)
Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:
- 99.95% uptime
- Redundancy across Availability Zones

● Single-AZ DB instance deployment (1 instance)
Creates a single DB instance without standby instances. This setup provides:
- 99.5% uptime
- No data redundancy

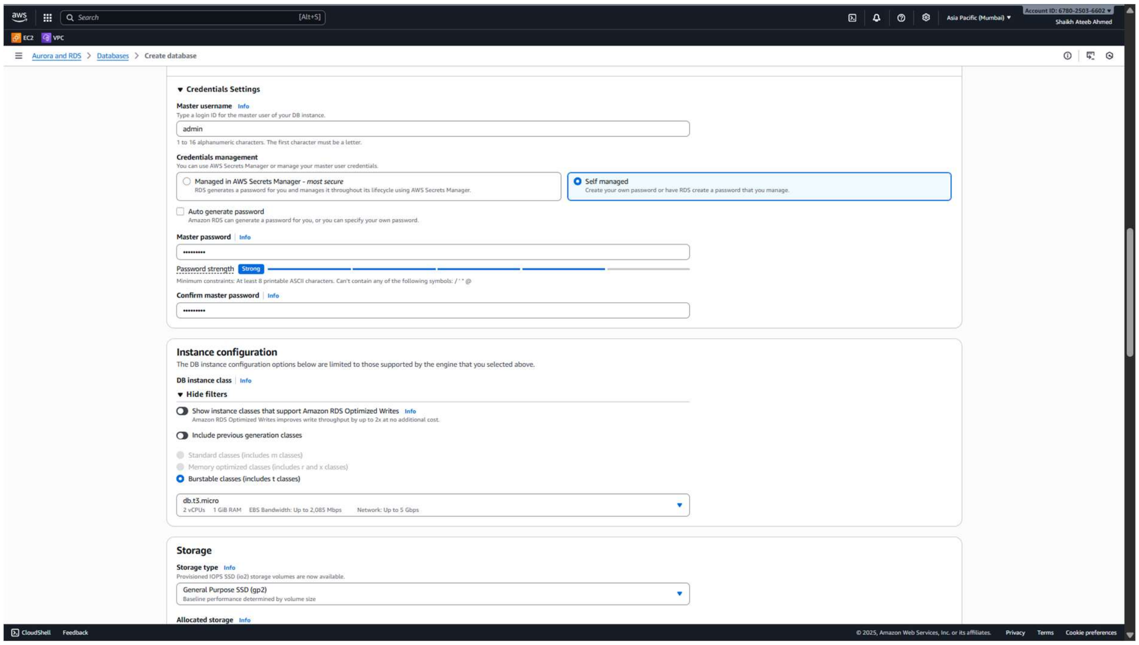
Settings

DB instance identifier Info
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
rds-mysql-lab
The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username Info
Type a login ID for the master user of your DB instance.
admin
1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.


aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
Aurora and RDS > Databases > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

▼ **Credentials Settings**

Master username Info
Type a login ID for the master user of your DB instance.
admin
1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

○ Managed in AWS Secrets Manager - *most secure*
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

● Self managed
Create your own password or have RDS create a password that you manage.

□ Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password | Info
•••••••••
Password strength Strong
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password | Info
•••••••••

Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class | Info

▼ **Hide filters**

○ Show instance classes that support Amazon RDS Optimized Writes Info
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

○ Include previous generation classes

○ Standard classes (includes m classes)
○ Memory optimized classes (includes r and x classes)
● Burstable classes (includes t classes)

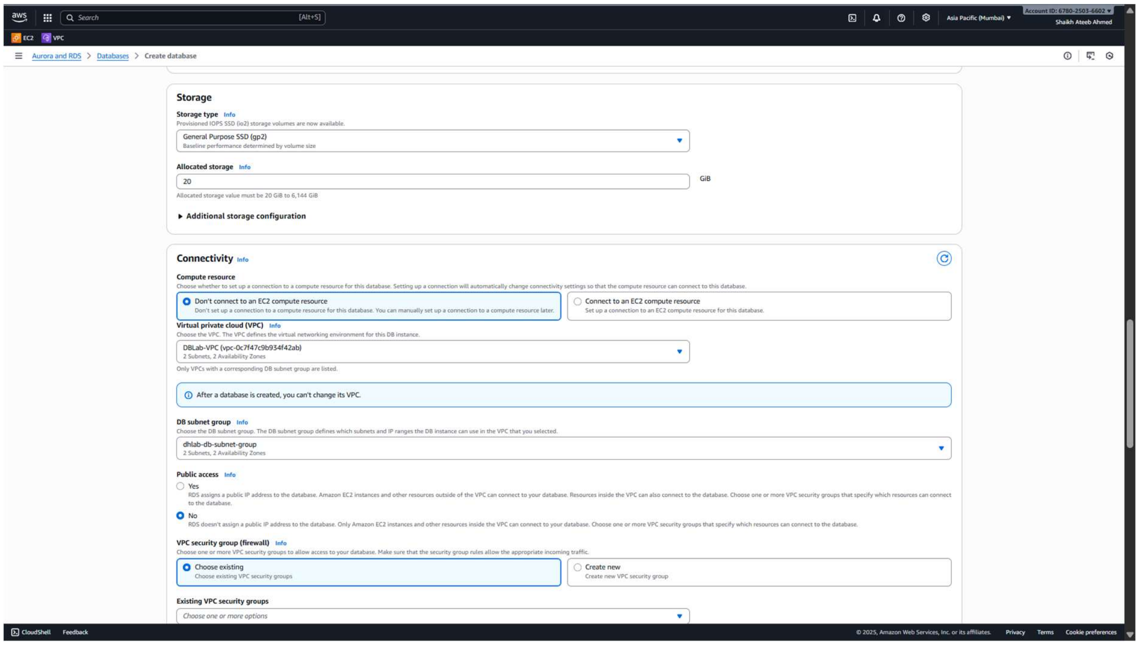
db.t3.micro
2 vCPUs 1 GiB RAM EBS Bandwidth: Up to 2,085 Mbps Network: Up to 5 Gbps

Storage

Storage type Info
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage Info

Shaikh Ateeb Ahmed
29-08-2025


aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
Aurora and RDS > Databases > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Storage

Storage type Info
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage Info
20 GiB
Allocated storage value must be 20 GiB to 6,144 GiB

▸ **Additional storage configuration**

Connectivity Info

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

● Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

○ Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.
DBLab-VPC (vpc-0c7f47c9b934f42ab)
2 Subnets, 2 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.
dhlab-db-subnet-group
2 Subnets, 2 Availability Zones

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.

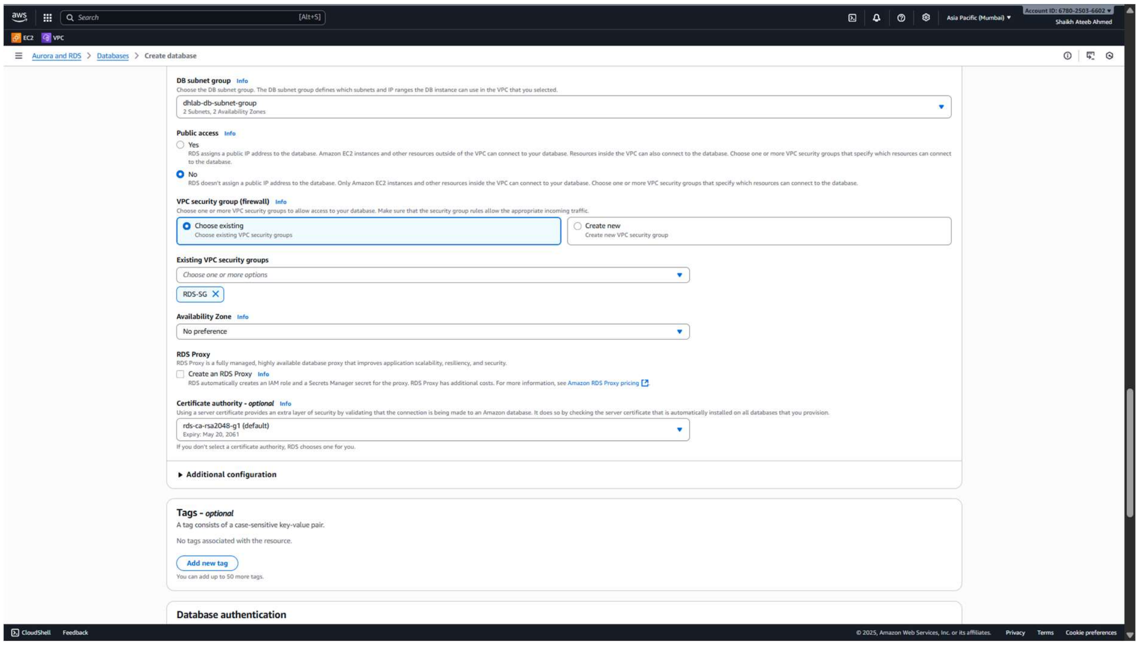
● No
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

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

Existing VPC security groups
Choose one or more options


aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
Aurora and RDS > Databases > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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.
dhlab-db-subnet-group
2 Subnets, 2 Availability Zones

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

Existing VPC security groups
Choose one or more options

RDS-SG ✕

Availability Zone Info
No preference

RDS Proxy
RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

□ Create an RDS Proxy Info
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see Amazon RDS Proxy pricing.

Certificate authority - *optional* Info
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.
rds-ca-rsa2048-g1 (default)
Expiry: May 20, 2061
If you don't select a certificate authority, RDS chooses one for you.

▸ **Additional configuration**

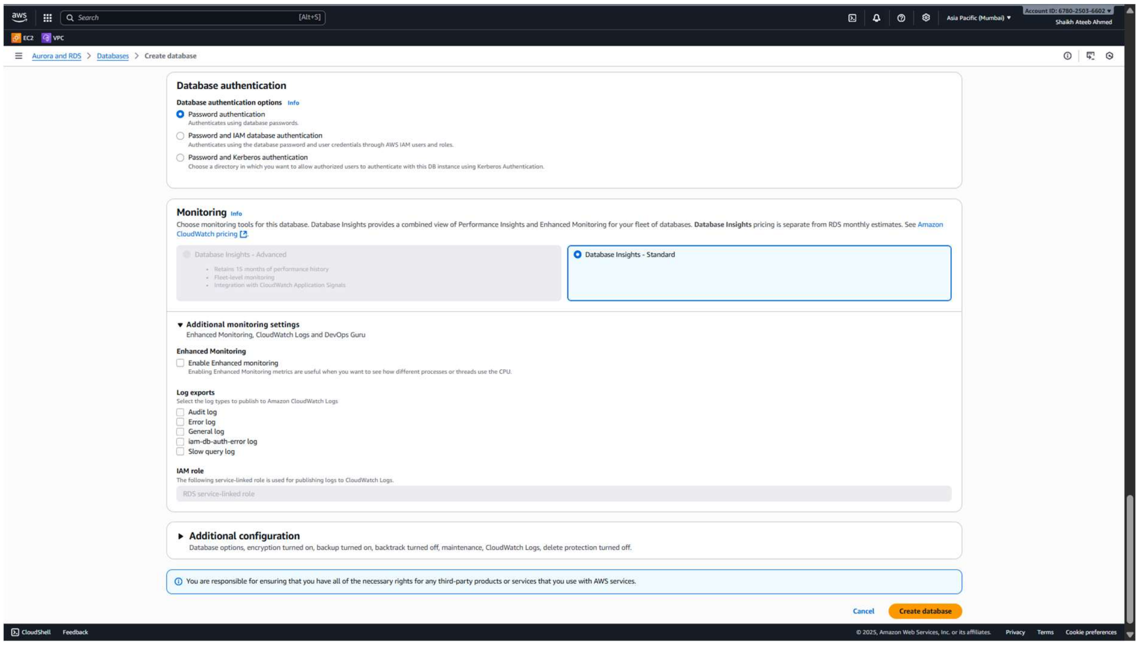
Tags - *optional*
A tag consists of a case-sensitive key-value pair.

No tags associated with the resource.

Add new tag
You can add up to 50 more tags.

Database authentication

Shaikh Ateeb Ahmed
29-08-2025


aws
Search [Alt+S]
Asia Pacific (Mumbai) ▾
Account ID: 6780-2503-6602 ▾
Shaikh Ateeb Ahmed
EC2
VPC
Aurora and RDS > Databases > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Database authentication

Database authentication options Info

● Password authentication
Authenticates using database passwords.

○ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

○ Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Monitoring Info

Choose monitoring tools for this database. Database Insights provides a combined view of Performance Insights and Enhanced Monitoring for your fleet of databases. **Database Insights** pricing is separate from RDS monthly estimates. See Amazon CloudWatch pricing

○ Database Insights - Advanced
- Retains 15 months of performance history
- Fleet-level monitoring
- Integration with CloudWatch Application Signals

● Database Insights - Standard

▼ Additional monitoring settings
Enhanced Monitoring, CloudWatch Logs and DevOps Guru

Enhanced Monitoring

□ Enable Enhanced monitoring
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

Log exports
Select the log types to publish to Amazon CloudWatch Logs

□ Audit log
□ Error log
□ General log
□ iam-db-auth-error log
□ Slow query log

IAM role
The following service-linked role is used for publishing logs to CloudWatch Logs.
RDS service-linked role

▶ Additional configuration
Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned off.

ⓘ You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Cancel | Create database

Locate the RDS Endpoint & Status

1. In **RDS Console → Databases**, click rds-mysql-lab.

2. Wait until **Status** is **Available**.

3. In the **Connectivity & security** tab, note:

- **Endpoint** (host): e.g., rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com

- **Port**: 3306 (MySQL)

- **VPC** and **Subnet group** used.

You will use the **endpoint** and **master username/password** to connect from the bastion.

Step 4: Connect from Bastion Host

Connect to Bastion EC2 from Windows Using PowerShell (simplest)

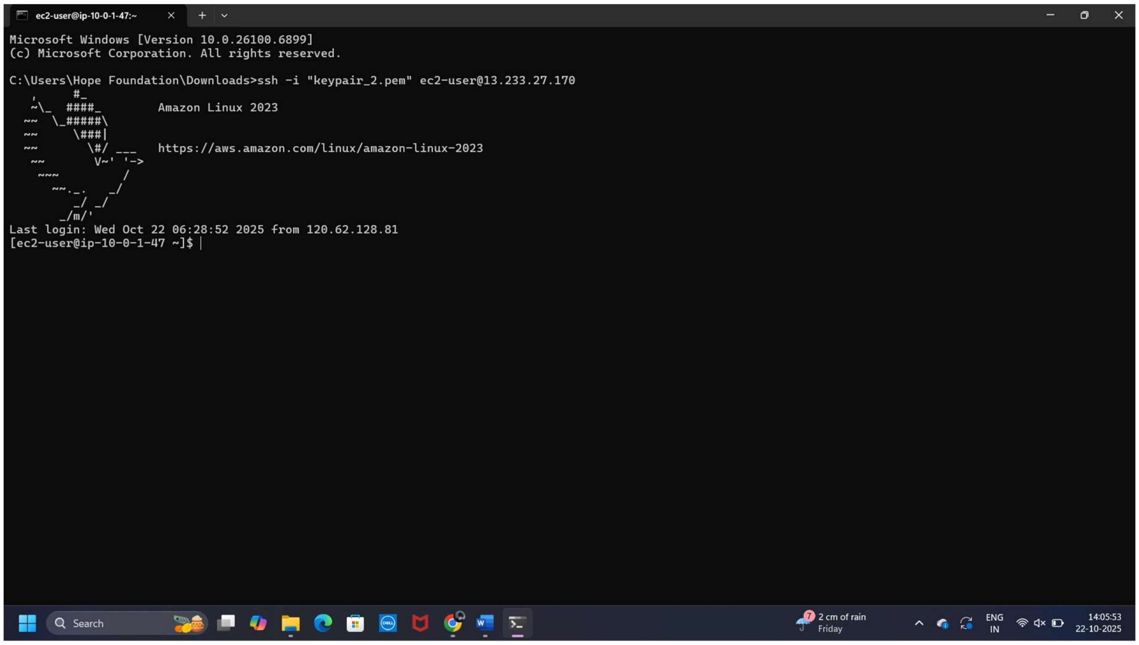
1. Open **PowerShell** and run:

```
ssh -i "mykey.pem" ec2-user@<EC2-Public-IP>
```

Example:

```
ssh -i "mykey.pem" ec2-user@3.109.45.22
```

Shaikh Ateeb Ahmed
29-08-2025

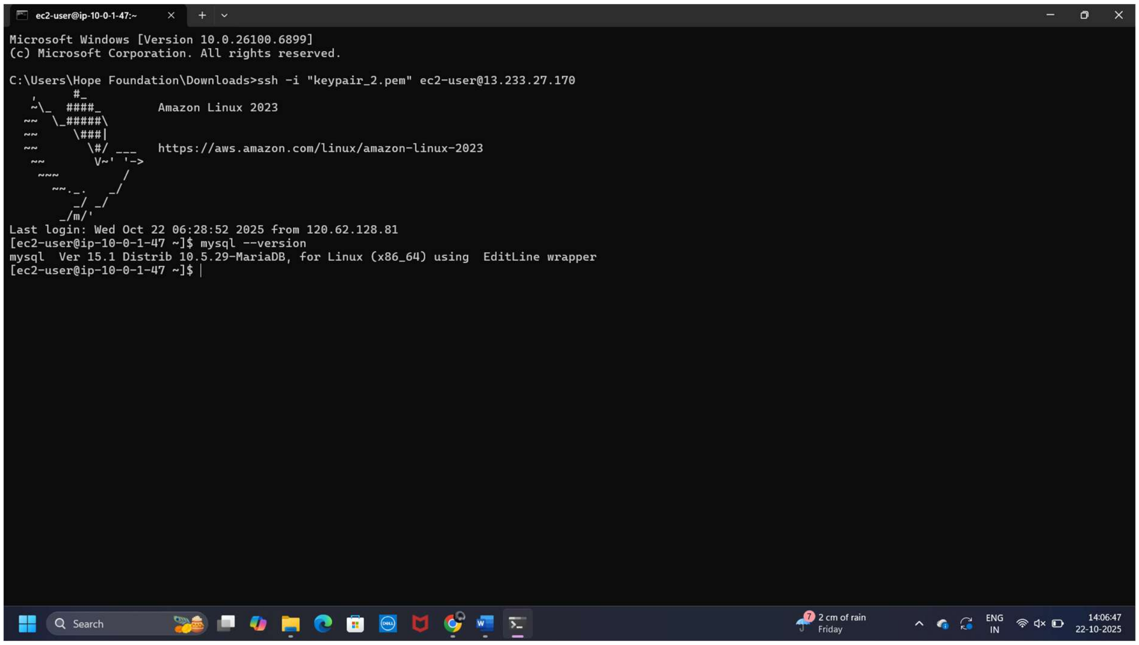


```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hope Foundation\Downloads>ssh -i "keypair_2.pem" ec2-user@13.233.27.170
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Wed Oct 22 06:28:52 2025 from 120.62.128.81
[ec2-user@ip-10-0-1-47 ~]$
```

Install MySQL client on the Bastion (Amazon Linux 2)

Verify installation:

mysql --version



```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hope Foundation\Downloads>ssh -i "keypair_2.pem" ec2-user@13.233.27.170
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Wed Oct 22 06:28:52 2025 from 120.62.128.81
[ec2-user@ip-10-0-1-47 ~]$ mysql --version
mysql  Ver 15.1 Distrib 10.5.29-MariaDB, for Linux (x86_64) using  EditLine wrapper
[ec2-user@ip-10-0-1-47 ~]$
```

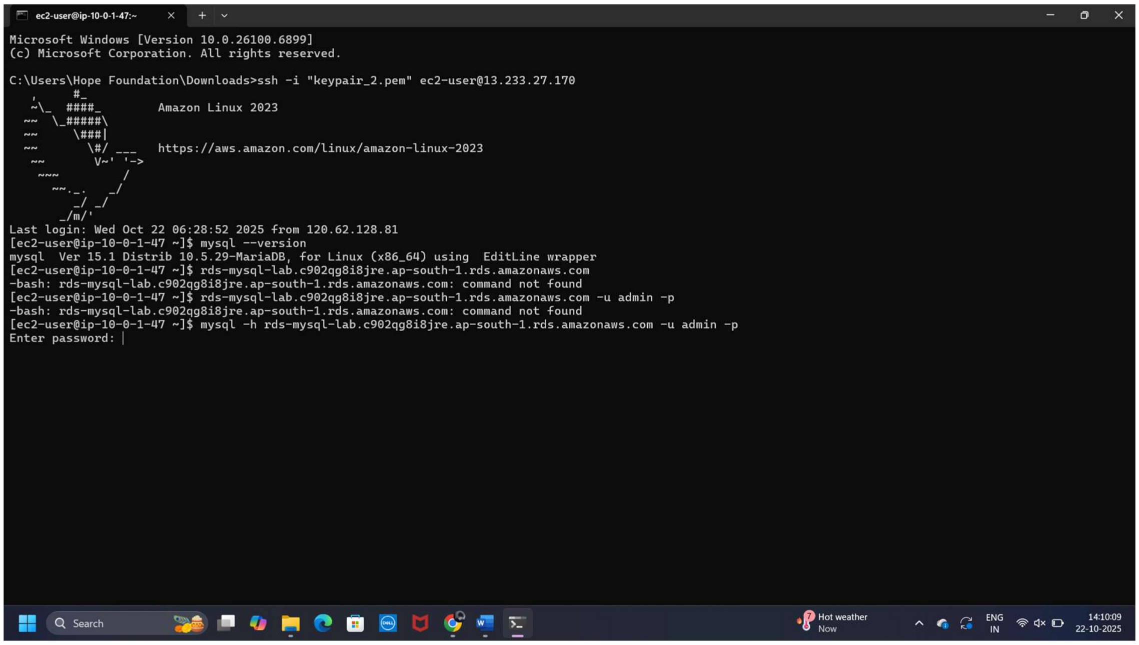
Connect to RDS MySQL from Bastion

Now connect securely using your RDS endpoint:

mysql -h <RDS-ENDPOINT> -u <USERNAME> -p

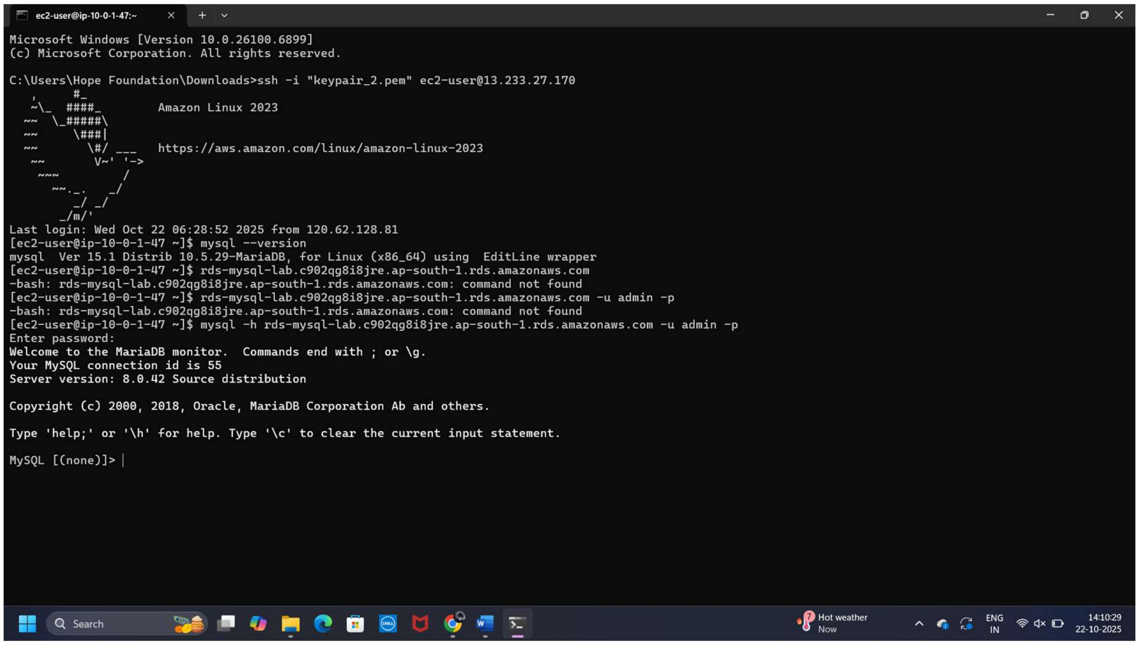
Example:

mysql -h rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p



```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hope Foundation\Downloads>ssh -i "keypair_2.pem" ec2-user@13.233.27.170
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Wed Oct 22 06:28:52 2025 from 120.62.128.81
[ec2-user@ip-10-0-1-47 ~]$ mysql --version
mysql  Ver 15.1 Distrib 10.5.29-MariaDB, for Linux (x86_64) using  EditLine wrapper
[ec2-user@ip-10-0-1-47 ~]$ rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com
-bash: rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com: command not found
[ec2-user@ip-10-0-1-47 ~]$ rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p
-bash: rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com: command not found
[ec2-user@ip-10-0-1-47 ~]$ mysql -h rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p
Enter password: |
```

Then enter your **RDS password** when prompted. (Pass: ateeb6867)



```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hope Foundation\Downloads>ssh -i "keypair_2.pem" ec2-user@13.233.27.170
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Wed Oct 22 06:28:52 2025 from 120.62.128.81
[ec2-user@ip-10-0-1-47 ~]$ mysql --version
mysql  Ver 15.1 Distrib 10.5.29-MariaDB, for Linux (x86_64) using  EditLine wrapper
[ec2-user@ip-10-0-1-47 ~]$ rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com
-bash: rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com: command not found
[ec2-user@ip-10-0-1-47 ~]$ rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p
-bash: rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com: command not found
[ec2-user@ip-10-0-1-47 ~]$ mysql -h rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 8.0.42 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)]> |
```

Test with SQL Queries

Once inside the MySQL shell, run:

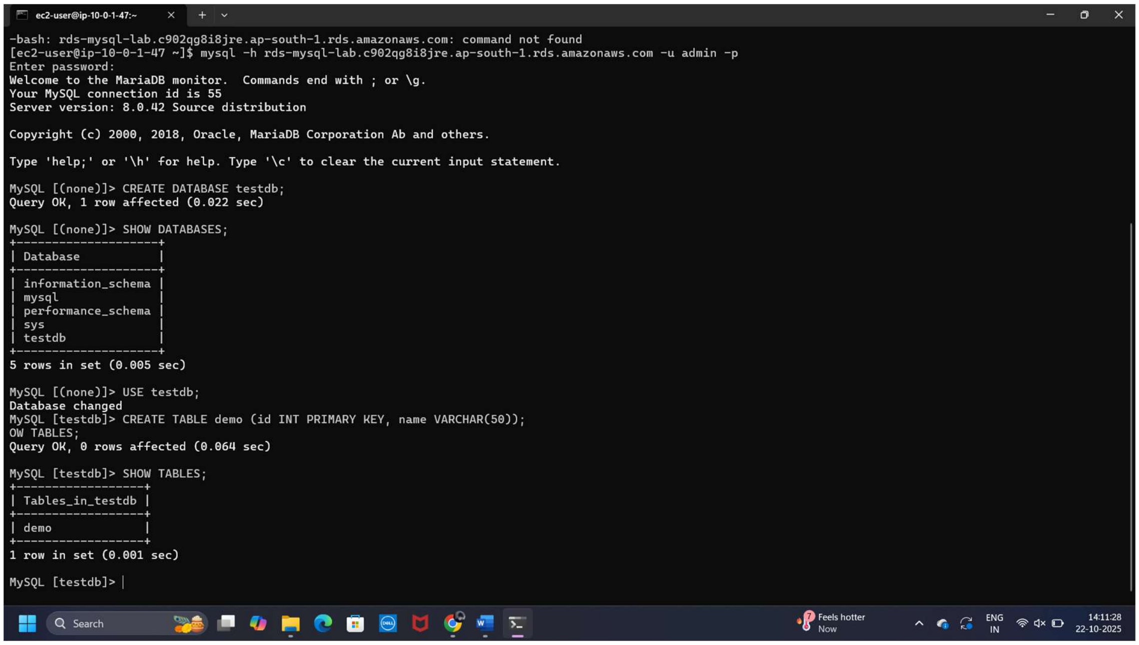
CREATE DATABASE testdb;

Shaikh Ateeb Ahmed
29-08-2025

```
SHOW DATABASES;

USE testdb;

CREATE TABLE demo (id INT PRIMARY KEY, name VARCHAR(50));

SHOW TABLES;
```

Expected output:

```
Database changed

mysql> show tables;

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

| Tables_in_testdb |

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

| demo             |

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



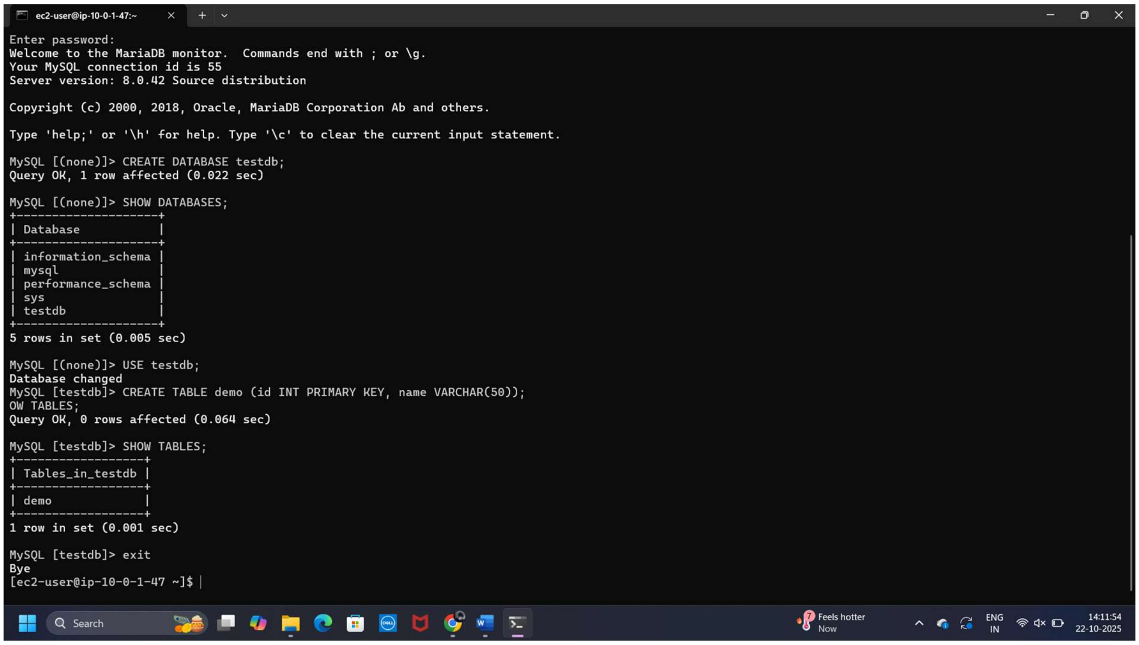
```
-bash: rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com: command not found
[ec2-user@ip-10-0-1-47 ~]$ mysql -h rds-mysql-lab.c902qg8i8jre.ap-south-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 8.0.42 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)]> CREATE DATABASE testdb;
Query OK, 1 row affected (0.022 sec)

MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| testdb             |
+--------------------+
5 rows in set (0.005 sec)

MySQL [(none)]> USE testdb;
Database changed
MySQL [testdb]> CREATE TABLE demo (id INT PRIMARY KEY, name VARCHAR(50));
OW TABLES;
Query OK, 0 rows affected (0.064 sec)

MySQL [testdb]> SHOW TABLES;
+------------------+
| Tables_in_testdb |
+------------------+
| demo             |
+------------------+
1 row in set (0.001 sec)

MySQL [testdb]> |
```

Exit

Type:

exit;

Shaikh Ateeb Ahmed
29-08-2025



```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 8.0.42 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)]> CREATE DATABASE testdb;
Query OK, 1 row affected (0.022 sec)

MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| testdb             |
+--------------------+
5 rows in set (0.005 sec)

MySQL [(none)]> USE testdb;
Database changed
MySQL [testdb]> CREATE TABLE demo (id INT PRIMARY KEY, name VARCHAR(50));
OW TABLES;
Query OK, 0 rows affected (0.064 sec)

MySQL [testdb]> SHOW TABLES;
+------------------+
| Tables_in_testdb |
+------------------+
| demo             |
+------------------+
1 row in set (0.001 sec)

MySQL [testdb]> exit
Bye
[ec2-user@ip-10-0-1-47 ~]$ |
```