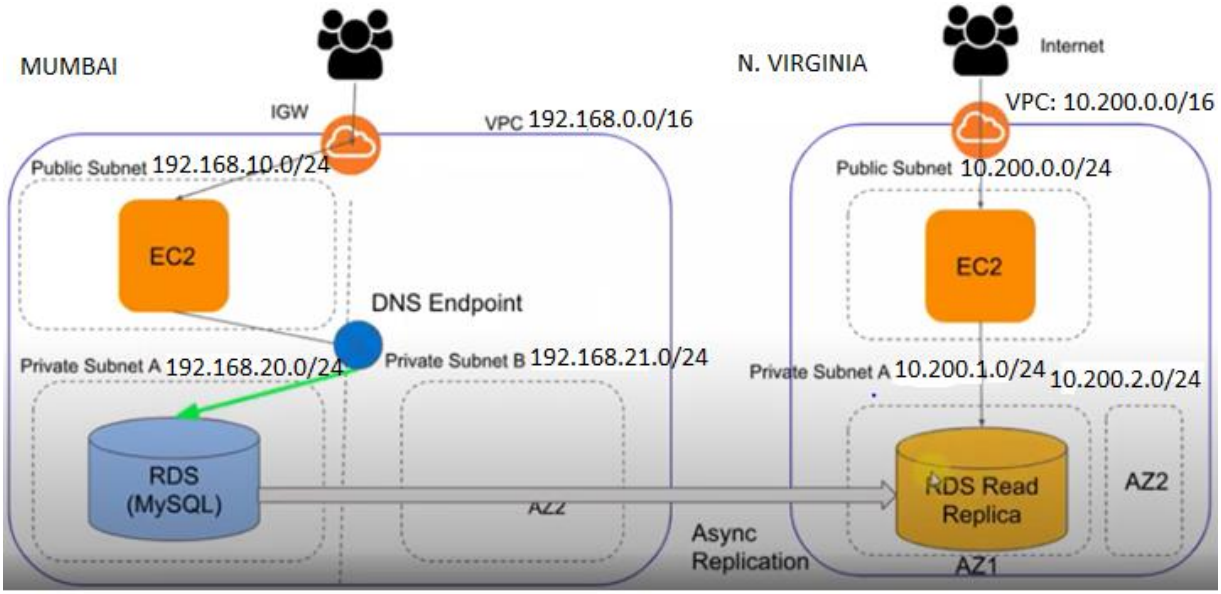
RDS Multi-Region Replication

In the previous lab, we deployed RDS Instances (Master & Read Replica) across the Availability Zones to maintain the High Availability across the Availability Zones and tested the Failover.

In this lab, we are going to use the same architecture but this time we are going to create Read Replicas across another AWS Region to maintain High Availability even more.

Now even if your entire primary region goes down, your application database will still be available and served by read replicas in another AWS region. This is another way to achieve High Availability within RDS Instances.



Below is the list of Tasks:

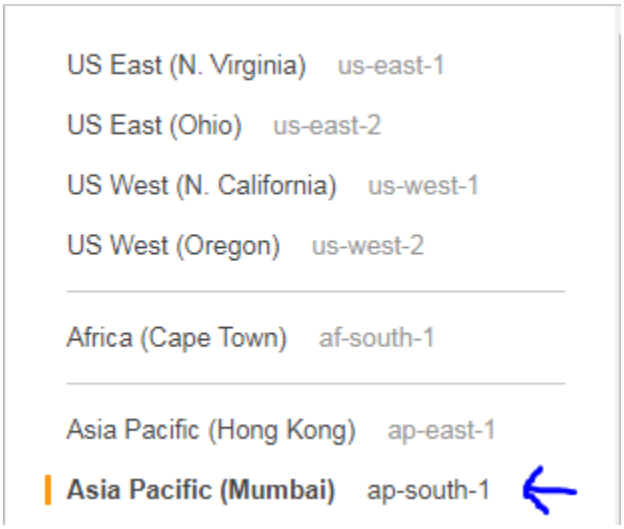
- Task 1: Create a Custom Network Design in another Region
- Task 2: Create a DB Subnet Group
- Task 3: Configure Read Replicas across AWS Region
- Task 4: Launch a Front-End Web Server
- Task 5: Configure Front-End Server with MySQL
- Task 6: Connect the Front-End Servers with RDS DB Instances
- Task 7: MySQL Administration
- Task 8: Test the Replication
- Task 9: Introduction to MySQL Workbench

Task 1: Create a Custom Network Design in another Region

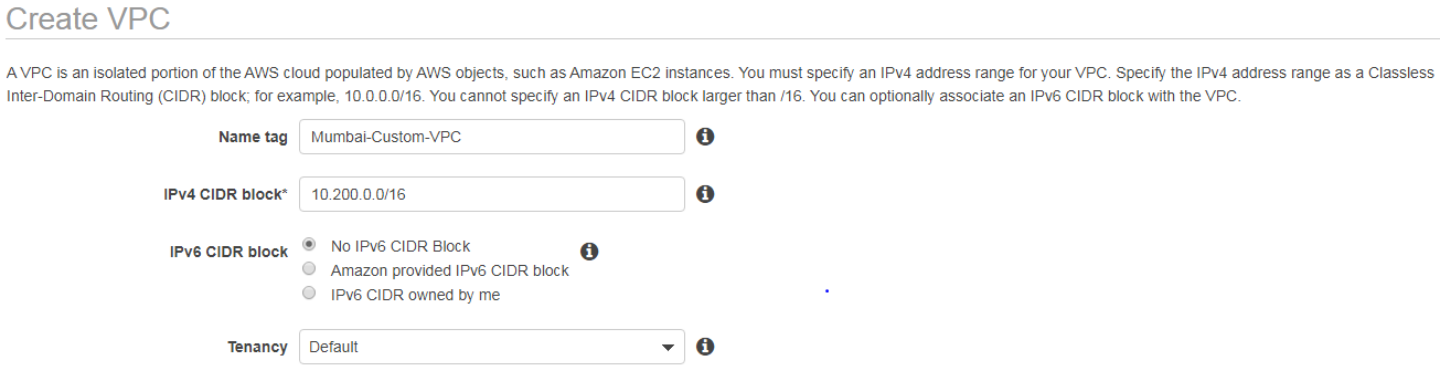
In all the previous labs, we've mainly worked in AWS region-N. Virginia (us-east-1).

In this Lab, we are going to select another Region as Mumbai (ap-south-1).

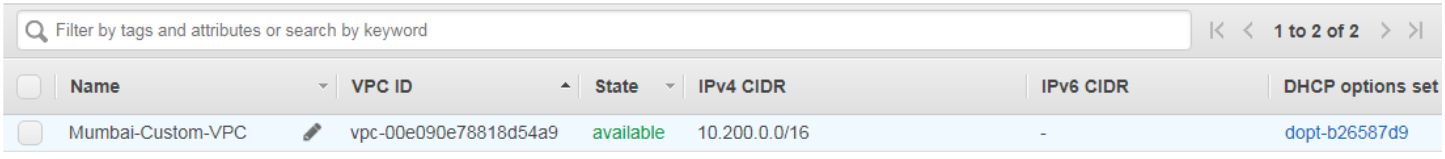
On the top of the AWS Console, select Region as Asia Pacific (Mumbai) (ap-south-1).



Navigate to VPC Service and create a Custom VPC (10.200.0.0/16).



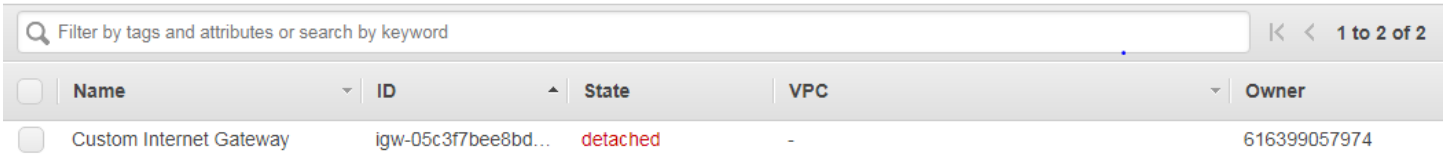
Click on Create.



Now click on Internet Gateway and click on Create Internet Gateway.

Give the Name tag.

Now click on Create.



Select the Internet Gateway which you've created and navigate to Actions and click on Attach to VPC.

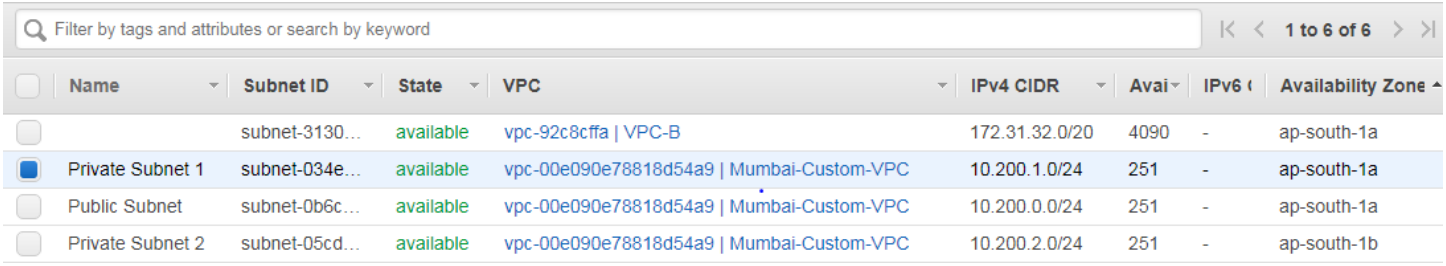
Select the VPC as Custom VPC and click on Attach.

On the left-hand side, now click on Subnets.

Create 1 Public Subnet of the CIDR 10.200.0.0/24 in Availability Zone “ap-south-1a”.

Create 1 Private Subnet of the CIDR 10.200.1.0/24 in Availability Zone “ap-south-1a”.

Create 1 Private Subnet of the CIDR 10.200.2.0/24 in Availability Zone “ap-south-1b”.

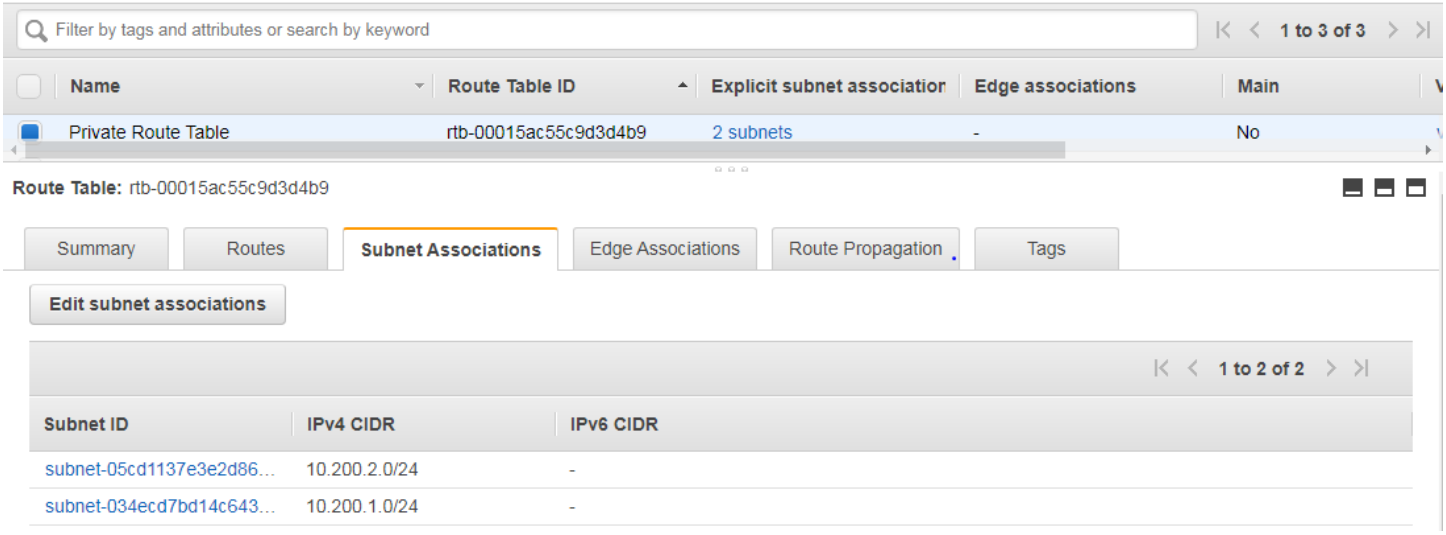


Make sure to place both the Private Subnets into two different Availability Zones.

Now click on Route Table and click on Create Route Table.

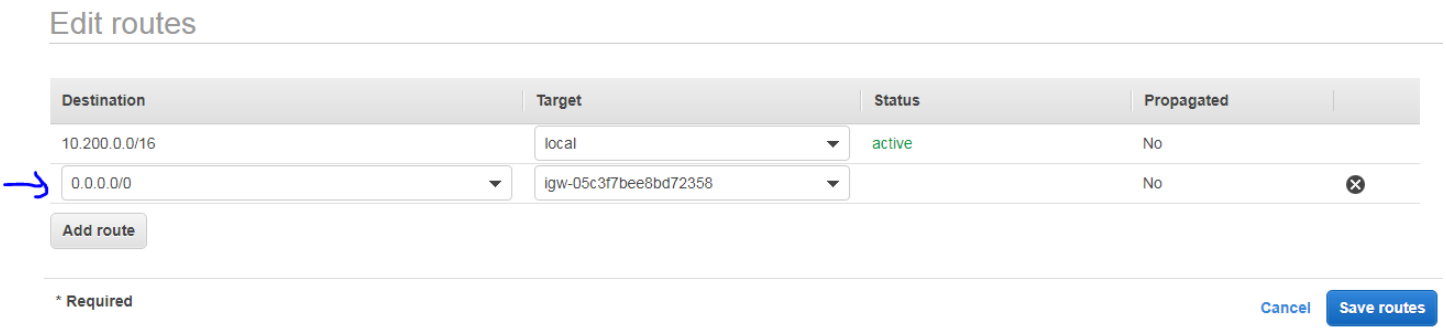
Create a Route Table with name “Private Route Table” and attach it to the Custom VPC.

Associate two Private Subnets to the Route Table.

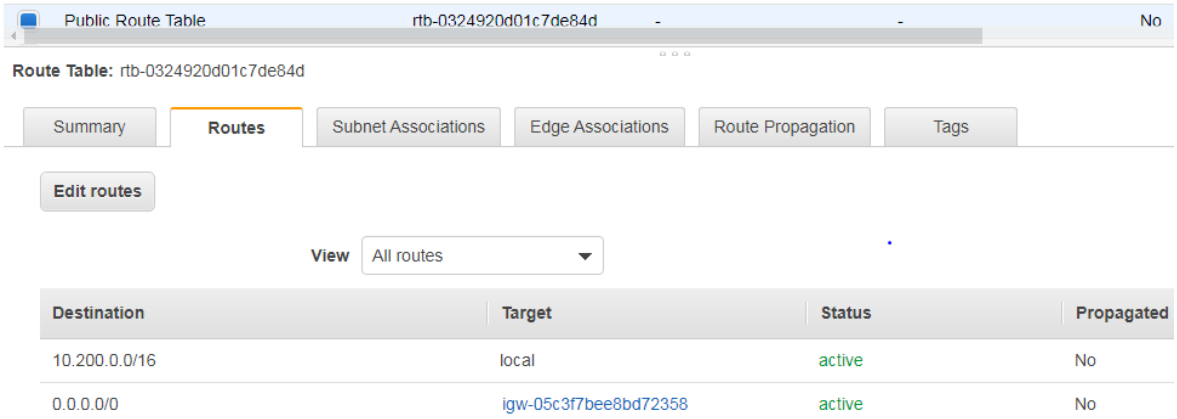


Now create a Route Table with name “Public Route Table” and attach it to the Custom VPC.

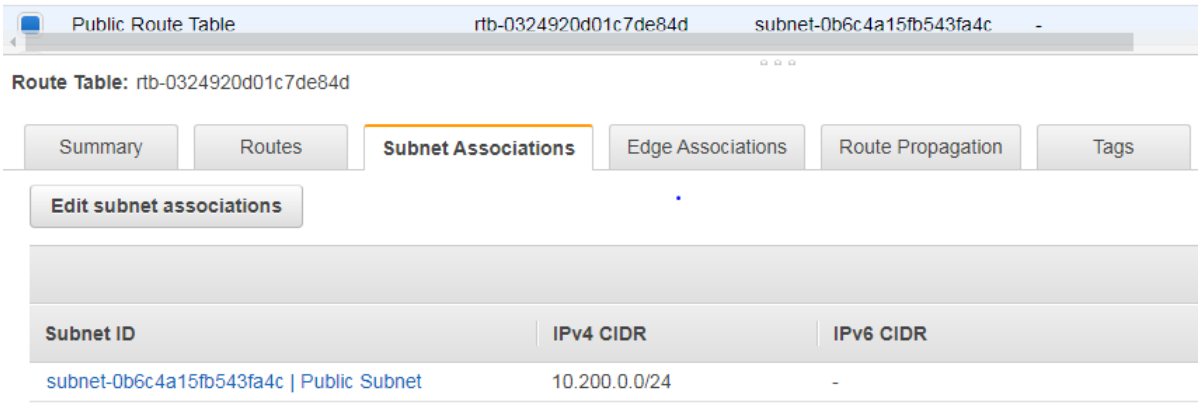
Once created, add a rule to route to the Internet through the Custom Internet Gateway which we created earlier.



Click on Save Routes.



Associate the one Custom Public Subnets to the Route Table.



Your Custom Network Design is now complete.

It is always a best practice to deploy your architecture in Custom VPC for Security purposes.

Now click on Security Groups.

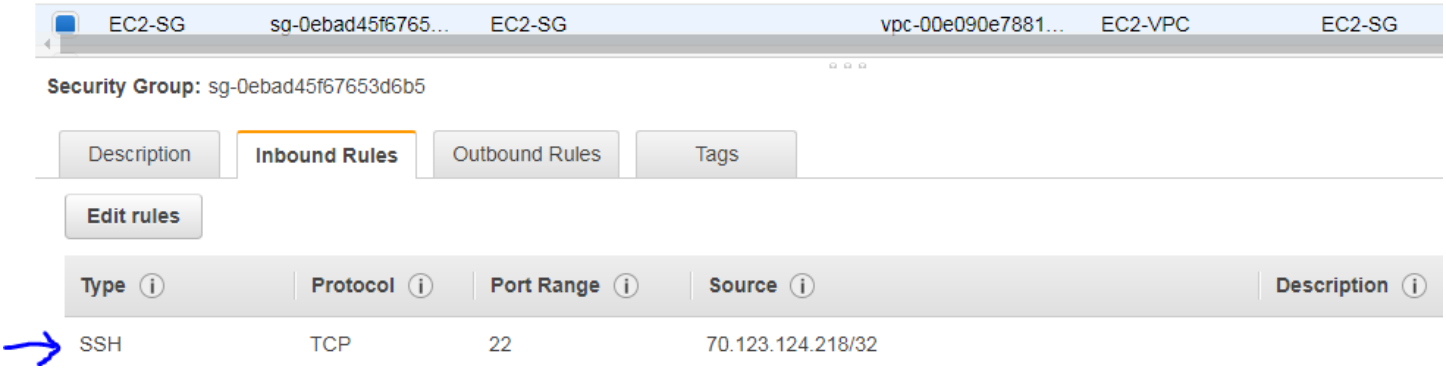
Create a Security Group for the RDS Instances and attach it to the Custom VPC.

Create an Inbound Rule as follows.

It means the RDS Instances in Private Subnets will now only be accessed by the EC2 Instances in Public Subnet (10.200.0.0/24). Keep the Outbound traffic as Default.

Now create a Security Group for EC2 Instances in Public Subnet and attach it to the Custom VPC.

Add the Inbound Rule as follows and keep the Outbound Rule as Default.



It means I'm the only one who can take SSH of EC2 Instances which are launched in the Public Subnet from my Host Machine.

We've configured Security for RDS Instances and EC2 Instances.

Your Custom Network Design is now complete.

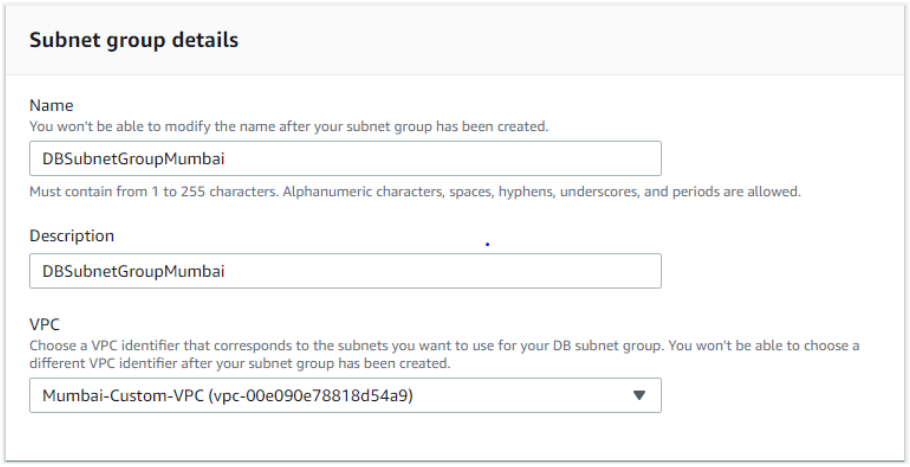
Task 2: Create a DB Subnet Group

Stay back to Asia Pacific (Mumbai) (ap-south-1).

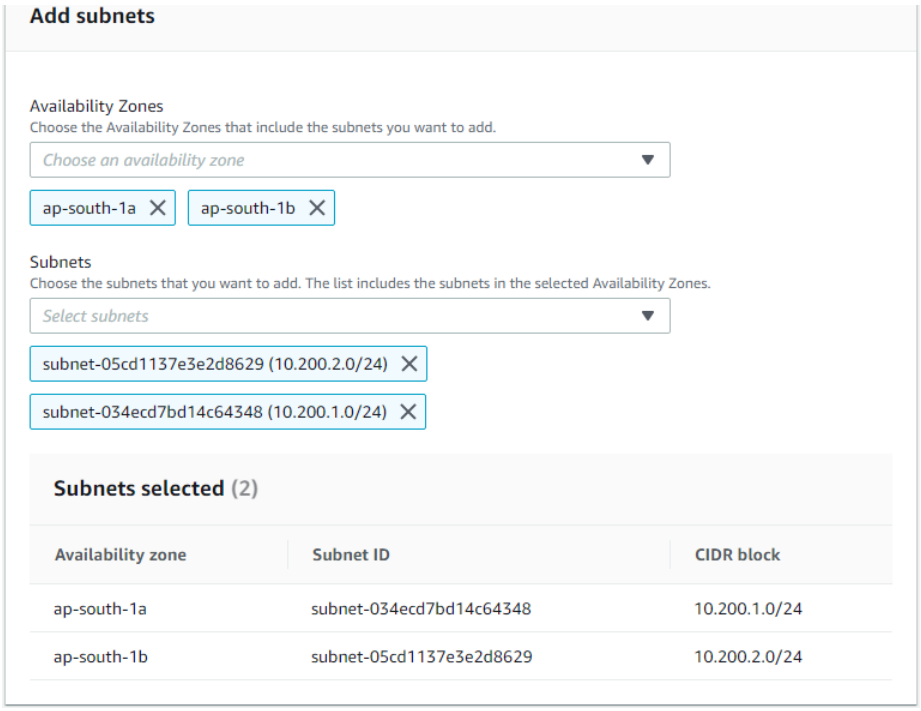
Navigate to Relational Database Service (RDS) and click on Subnet Groups.

Create a Subnet Group as follows.

Give Subnet Group Name and choose Custom VPC.



Select both the Availability Zones and choose Private Subnet 1 and Private Subnet 2.



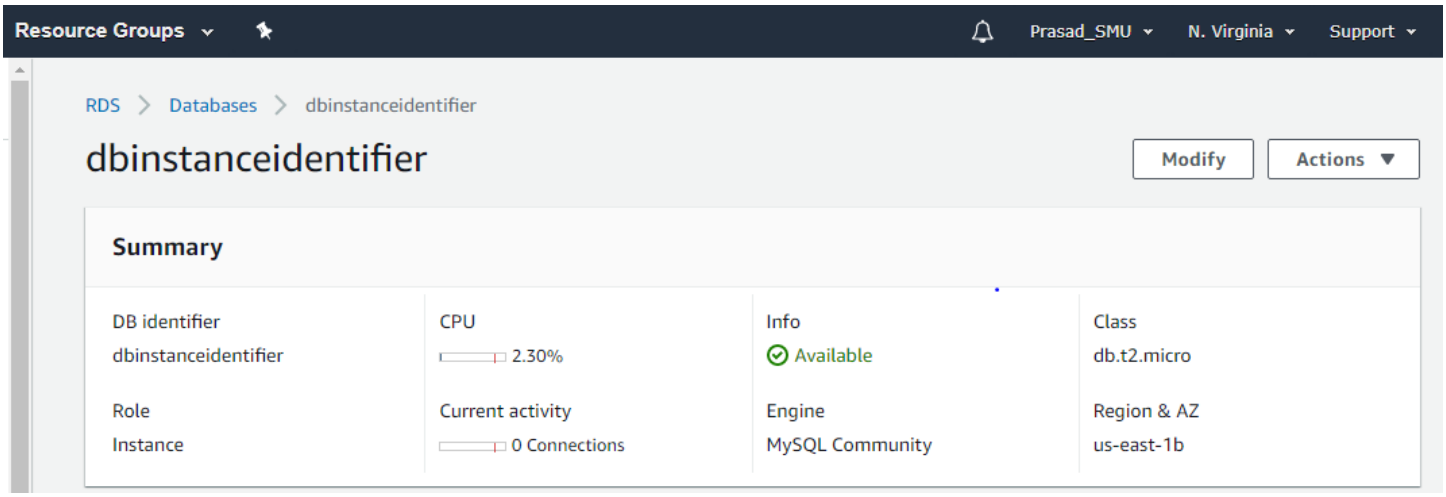
Now click on Create.

Task 3: Configure Read Replicas across AWS Region

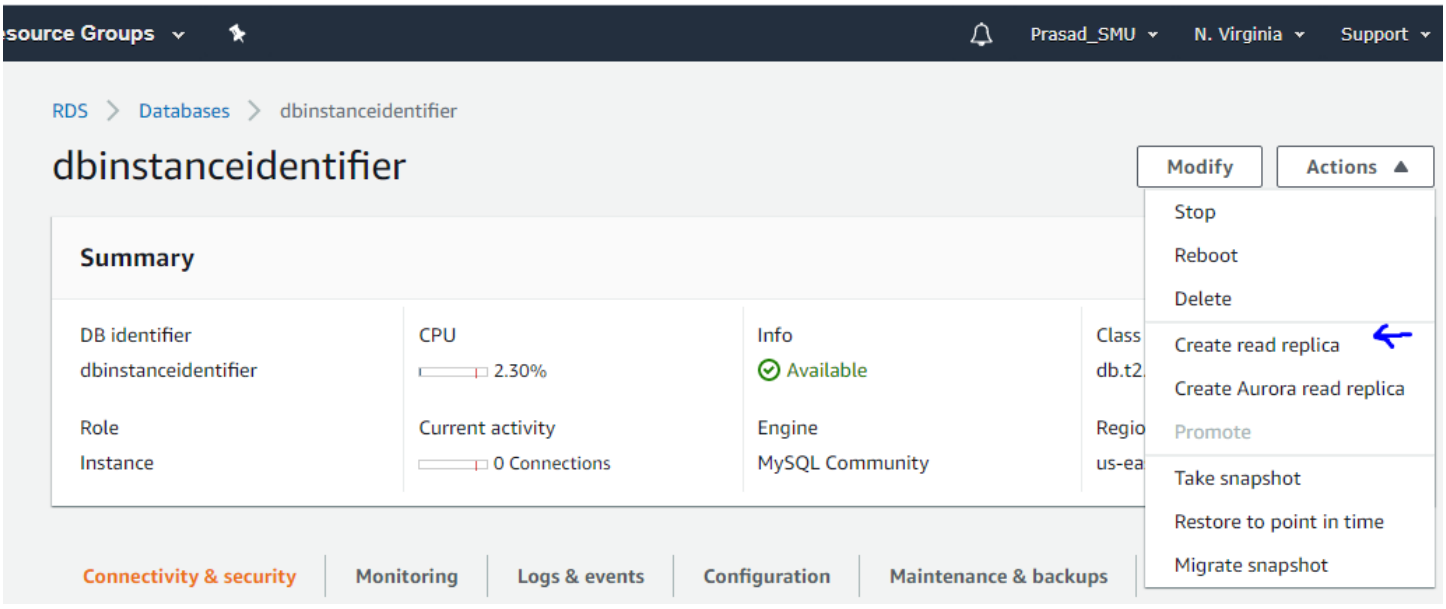
Come back to US East (N. Virginia) (us-east-1) region, wherein we have our original architecture.

Navigate to the Relational Database Service (RDS).

Under Databases, click on your DB Instance.



Now click on Actions and click on Create Read Replica.



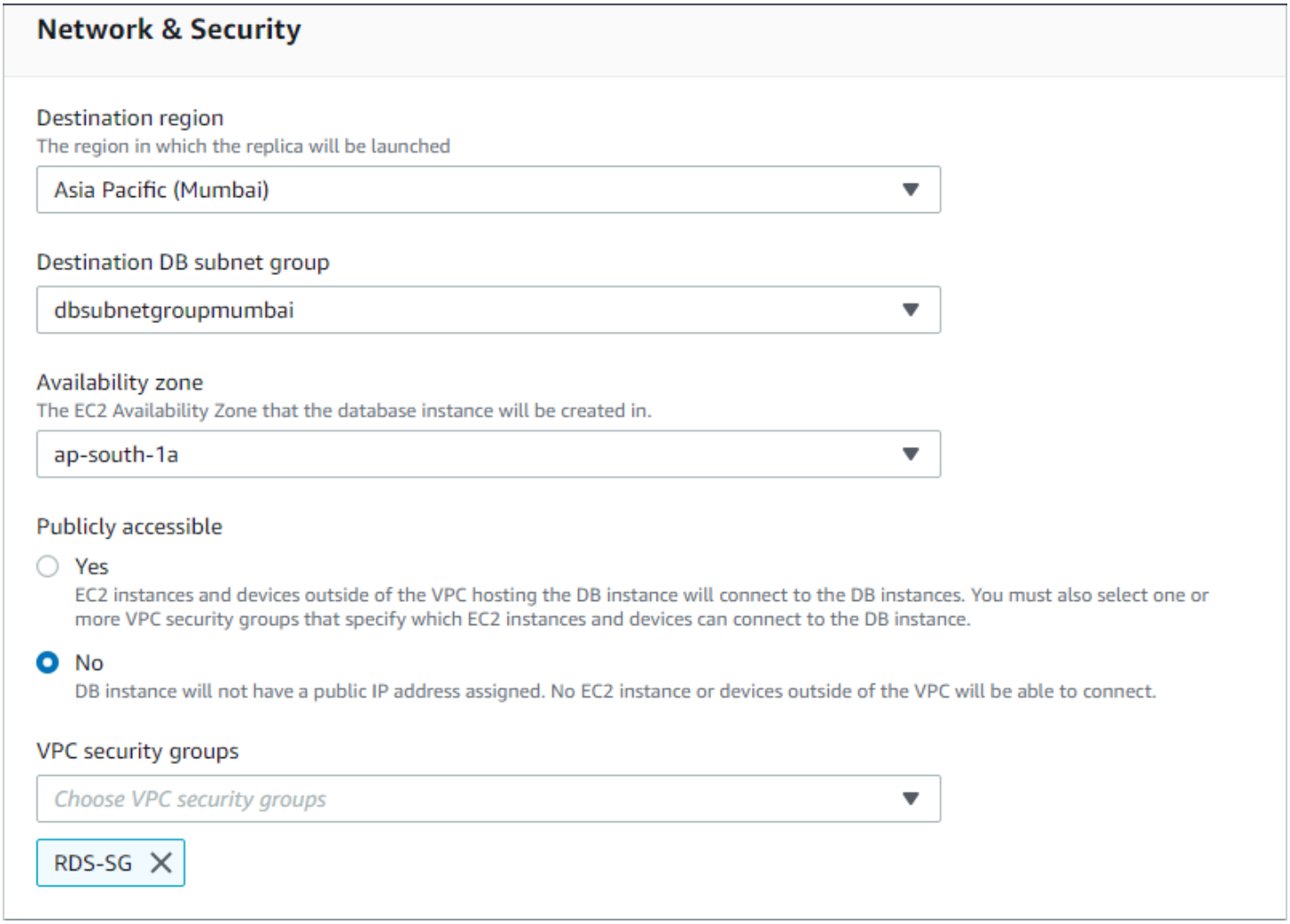
Under Network settings, select Destination Region as Asia Pacific (Mumbai).

Click on Destination DB Subnet Group, it will automatically pop the DB Subnet that you've created in Mumbai region.

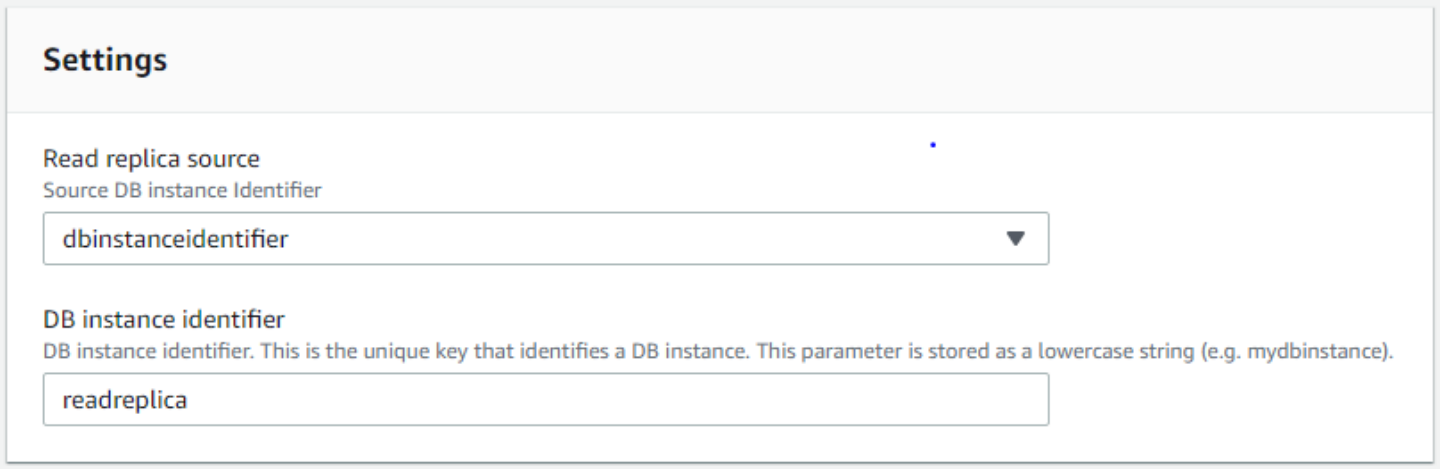
Select the AZ preference as ap-south-1a. It means DB Instance Read Replica will be launched in Mumbai Region's AZ (ap-south-1a) and in Private Subnet 1 (10.200.1.0/24).

Since we are launching our DB Instance Read Replica in Private Subnets, there is no sense of making the Read Replica publicly available, select NO.

Remove the Default Security Group and add a Security Group that we've created for DB Instances.

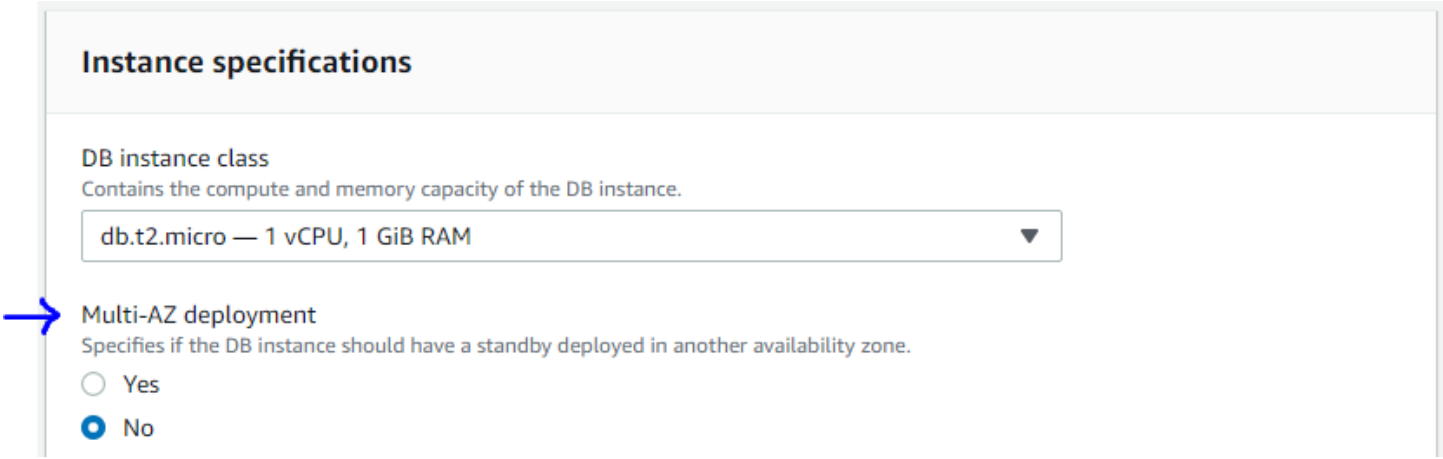


Give the DB Instance Identifier Name of your choice.



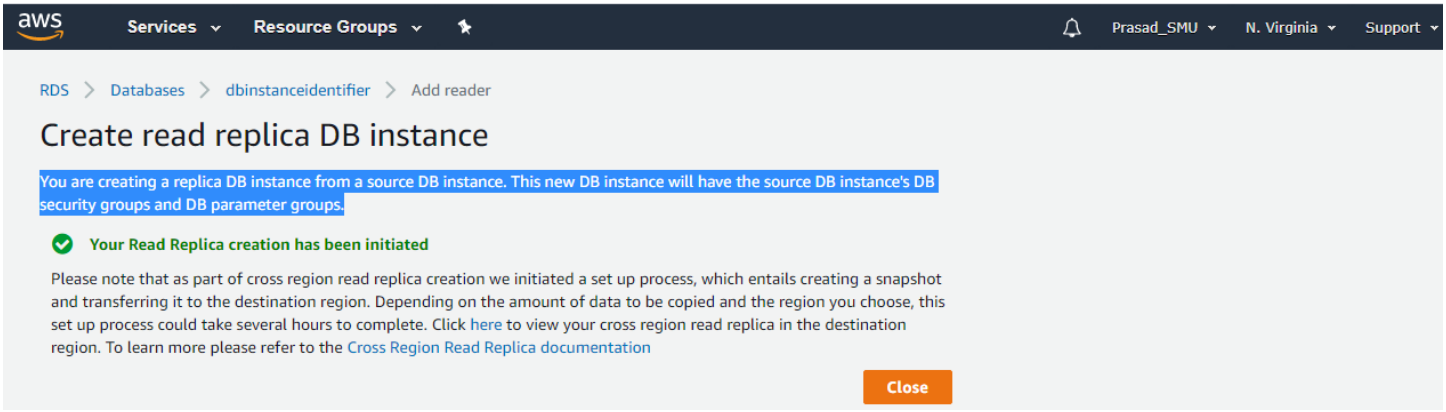
Under Instance Specifications, you'll notice that the option for Multi-AZ deployment.

If we click on YES, then it will launch read replicas in Private Subnet 1 and Private Subnet 2 which are different Availability Zones in Mumbai (ap-south-1) region. This makes your architecture even more Highly Available.



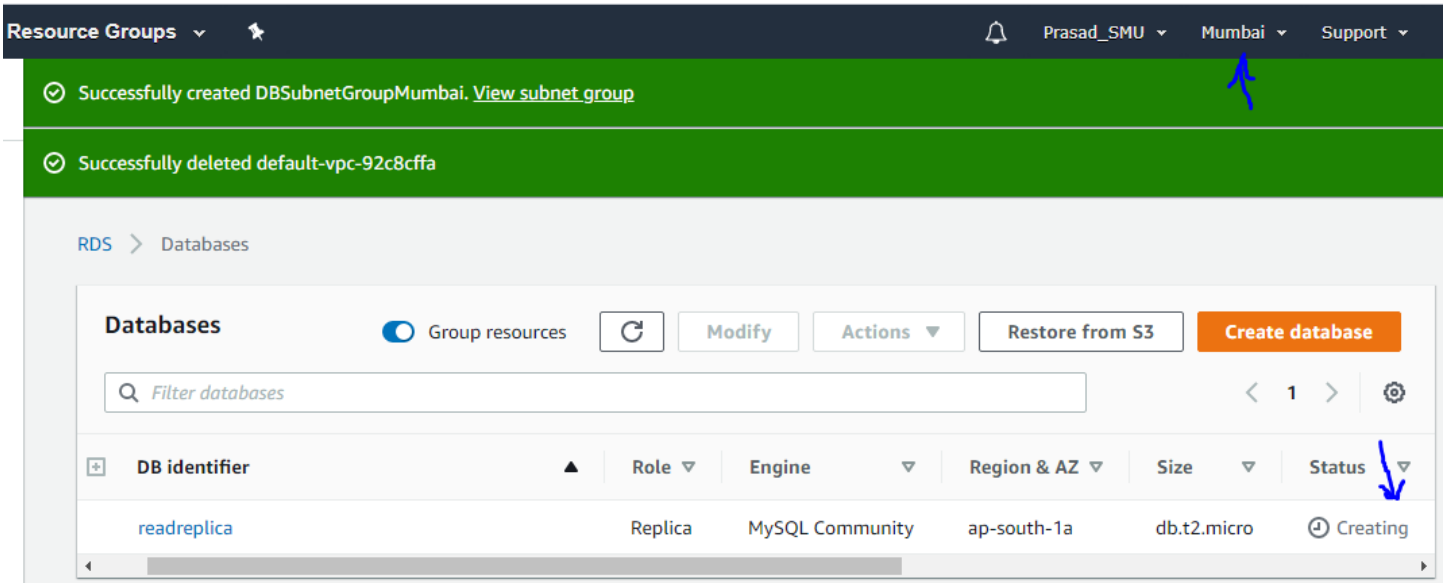
For this purpose of the Lab, we are not going to deploy Multi-AZ deployment in Mumbai (ap-south-1) region, hence we'll select NO.

Finally, review all the configurations and click on Create Read Replica.



Come back to Mumbai (ap-south-1) region and navigate to the Relational Database Service (RDS) and click on Databases.

You'll notice that new Read Replica is getting created.



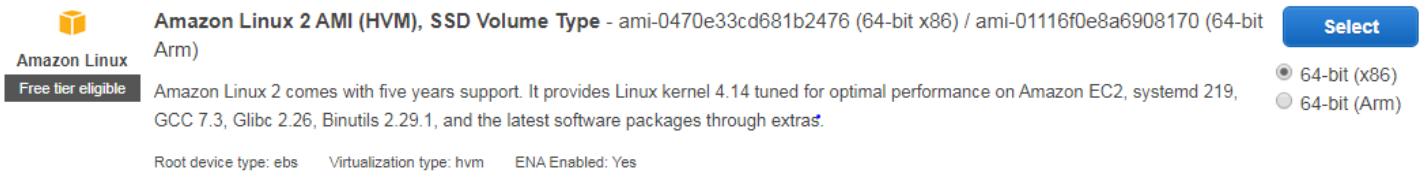
Read Replica will be Available in 5-10 minutes.

Task 4: Launch a Front-End Web Server

Switch to the Mumbai (ap-south-1) region.

Navigate to the EC2 Service and Launch a Linux EC2 Instance.

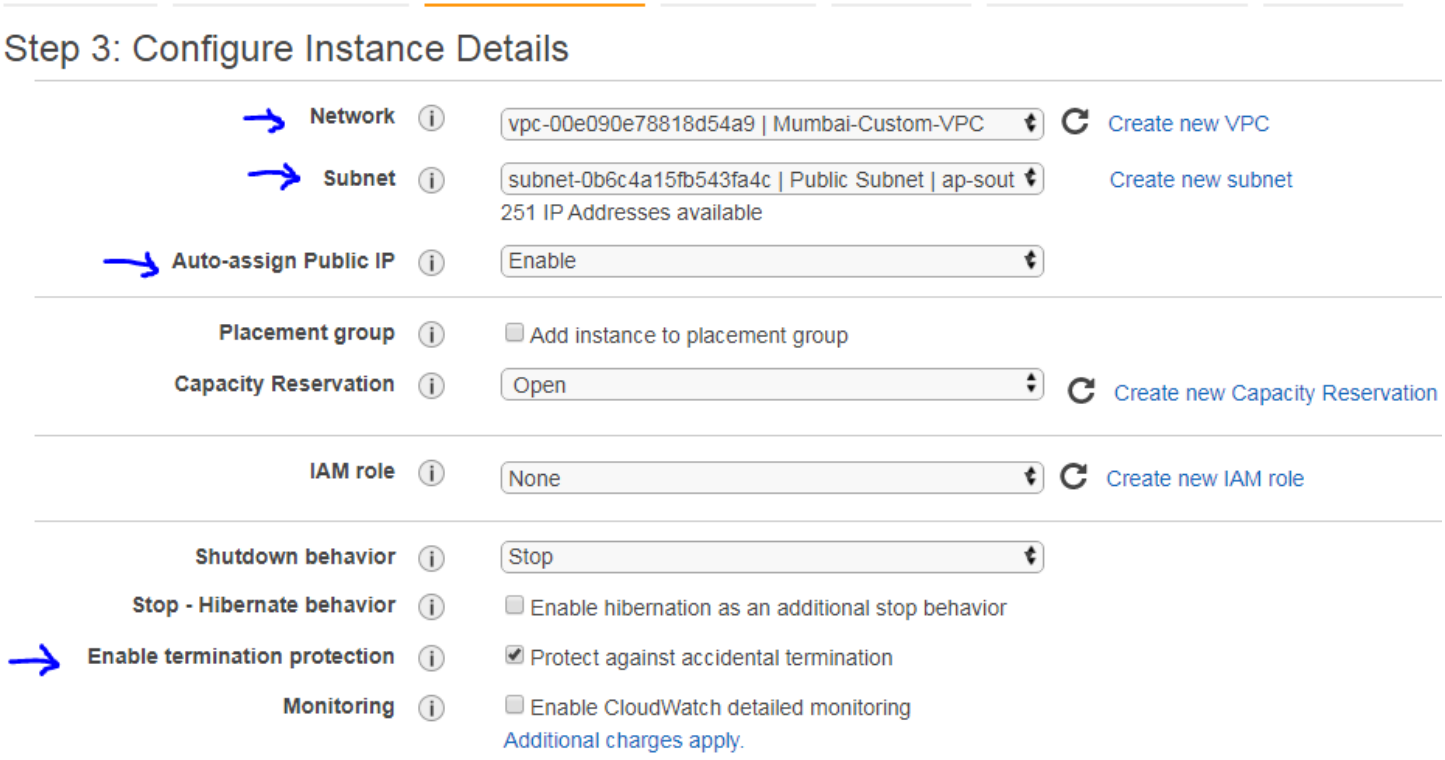
Select the Instance AMI as **Amazon Linux 2 AMI (HVM), SSD Volume Type.**



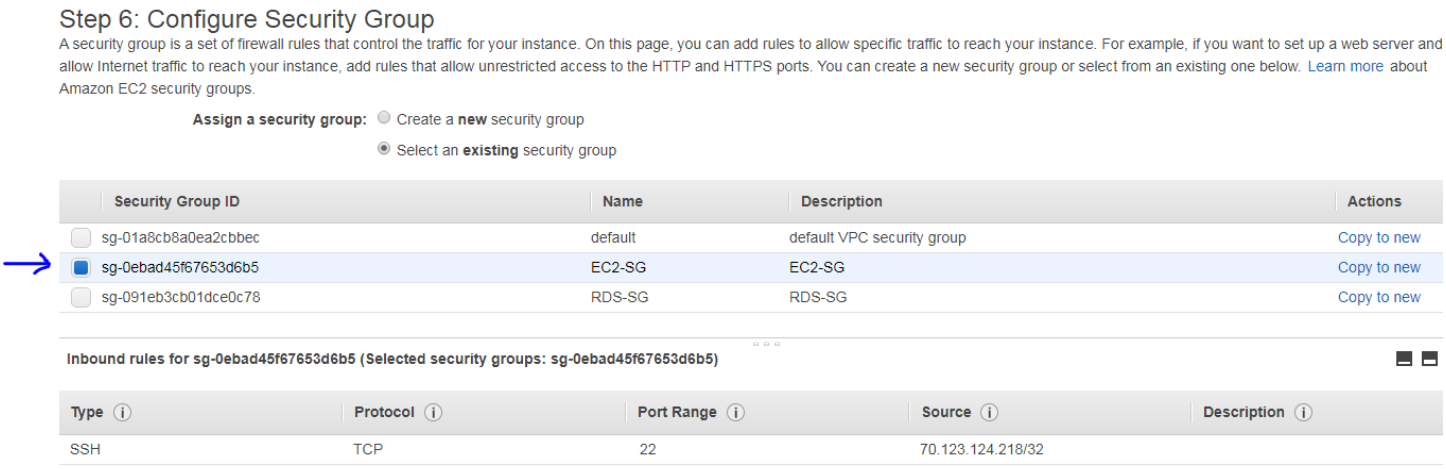
Configure the Instance as follows.

Specify the Network as Custom VPC and select Subnet as Public Subnet.

Also select the Auto-assign Public IP as Enable and Enable Termination Protection.



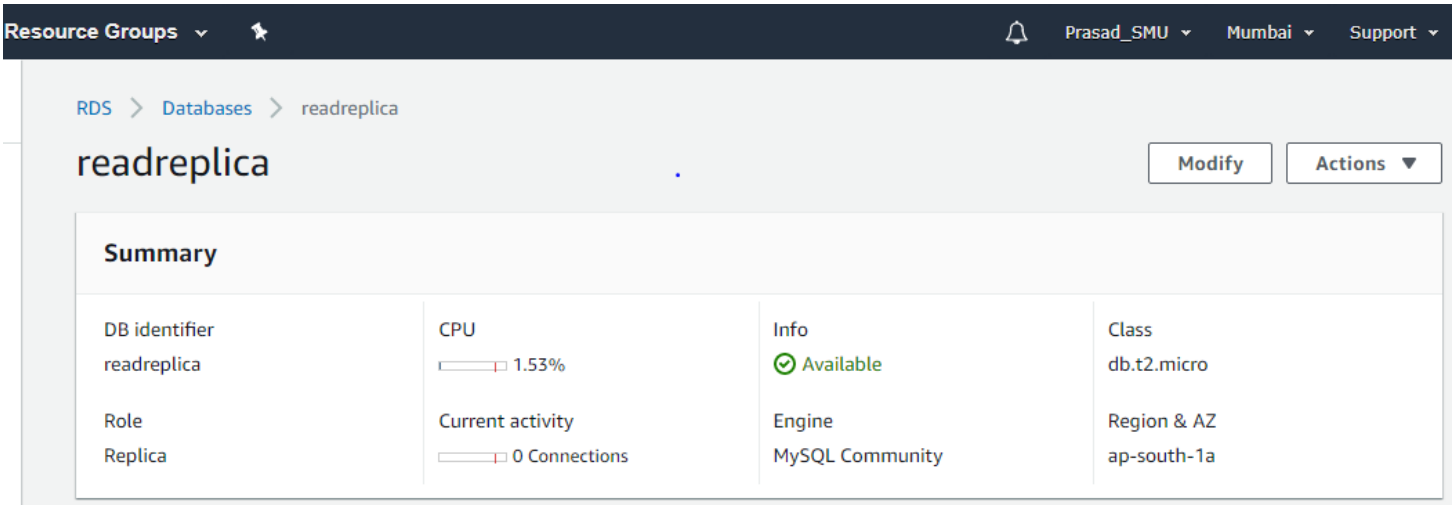
Select the Custom Security Group that we've created.



Download a new key pair for this Instance and save it in the appropriate location on your Desktop and finally click on Launch.

Now the EC2 Instance is now AVAILABLE in the Mumbai (ap-south-1) region.

RDS Read Replica is now AVAILABLE in the Mumbai (ap-south-1) region.



Task 5: Configure Front-End Server with MySQL

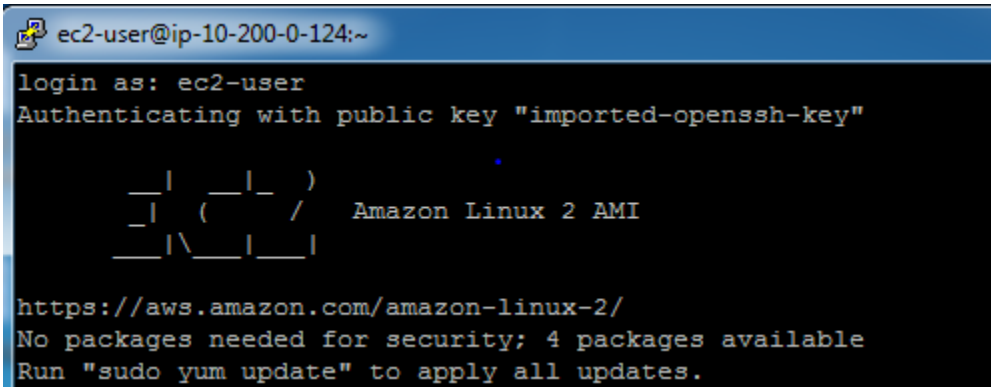
First, we'll connect the Front-End server with DB Instances in Mumbai (ap-south-1) region.

First, using the PuTTYgen, convert the downloaded. pem key to. ppk format.

- Open puttygen.exe
- In the PuTTY Key Generator panel, choose File > Load private key.
- At the bottom of the Load private key panel, click on the drop-down menu that displays *PuTTY Private Key Files (.ppk) and choose All Files**.
- Still in same panel, browse to the directory where you downloaded the .pem file (for example the Downloads directory).
- Select. pem and click Open.
- A PuTTYgen Notice screen should display, indicating that the key was successfully imported. Click OK.
- Click Save private key, then click Yes to save it without a passphrase.
- Give it the filename and click Save.
- Click the X at the top right of the PuTTY Key Generator to close it.

To connect to your Linux EC2 Instance, follow the below steps.

- Open PuTTY software.
- Give the Hostname as the Public IP Address of the Linux EC2 Instance.
- Click on Connections, then click on SSH, then click on Auth.
- Browse the .ppk file and hit Open.
- For certificates validation, click on YES.
- Provide the username as "ec2-user".



You're Linux EC2 Instance has been successfully launched.

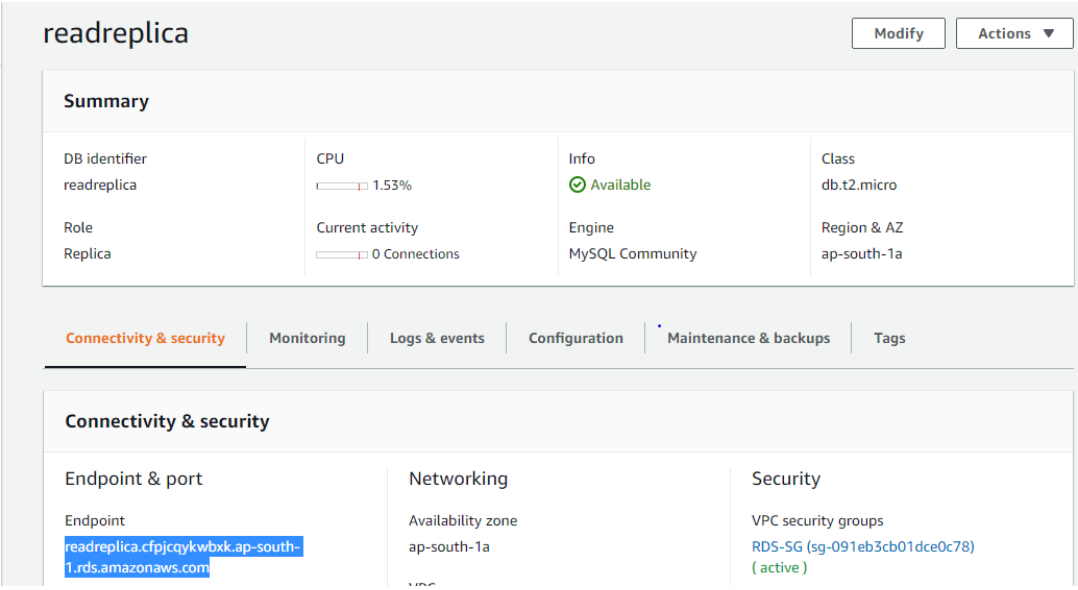
Now install the MySQL client on the Linux EC2 Instance.

Command: Sudo yum install mysql

```
[ec2-user@ip-10-200-0-124 ~]$ sudo yum install mysql
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 2.4 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package mariadb.x86_64 1:5.5.64-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package        Arch          Version                   Repository         Size
================================================================================
Installing:
 mariadb        x86_64        1:5.5.64-1.amzn2          amzn2-core        9.0 M

Transaction Summary
================================================================================
Install  1 Package

Total download size: 9.0 M
Installed size: 49 M
Is this ok [y/d/N]: y
Downloading packages:
mariadb-5.5.64-1.amzn2.x86_64.rpm                        | 9.0 MB   00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : 1:mariadb-5.5.64-1.amzn2.x86_64                              1/1
  Verifying  : 1:mariadb-5.5.64-1.amzn2.x86_64                              1/1

Installed:
  mariadb.x86_64 1:5.5.64-1.amzn2

Complete!
[ec2-user@ip-10-200-0-124 ~]$
```

Task 6: Connect the Front-End Server with RDS DB Instances

Now to connect the EC2 Instance and DB Instance in Mumbai region, first copy the RDS Instance Endpoint.



Go back to PuTTY session and run the below command to connect your EC2 Instance to the RDS Database.

Command:

mysql -h readreplica.cfpjcqykwbxk.ap-south-1.rds.amazonaws.com -u master -P 3306 -p

Give the password that you've set.

Congratulations, you've now connected your EC2 Instance with DB Instance (Read Replica) in Mumbai Region.

```
[ec2-user@ip-10-200-0-124 ~]$ mysql -h readreplica.cfpjcqykwbxk.ap-south-1.rds.amazonaws.com -u master -P 3306 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 5.7.22 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)]>
```

Task 7: MySQL Administration

Now check the existing Databases. Run the following Command.

Command: show databases;

```
MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| innodb             |
| mydb               |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0.00 sec)

MySQL [(none)]>
```

Let's get into the "mydb" database and verify the tables that we created on the RDS Instance in N. Virginia Region.

To get into the get into the "mydb" database, run the following command.

Command: use mydb;

```
MySQL [(none)]> use mydb;
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 [mydb]>
```

Let's now verify the Table. Run the below command.

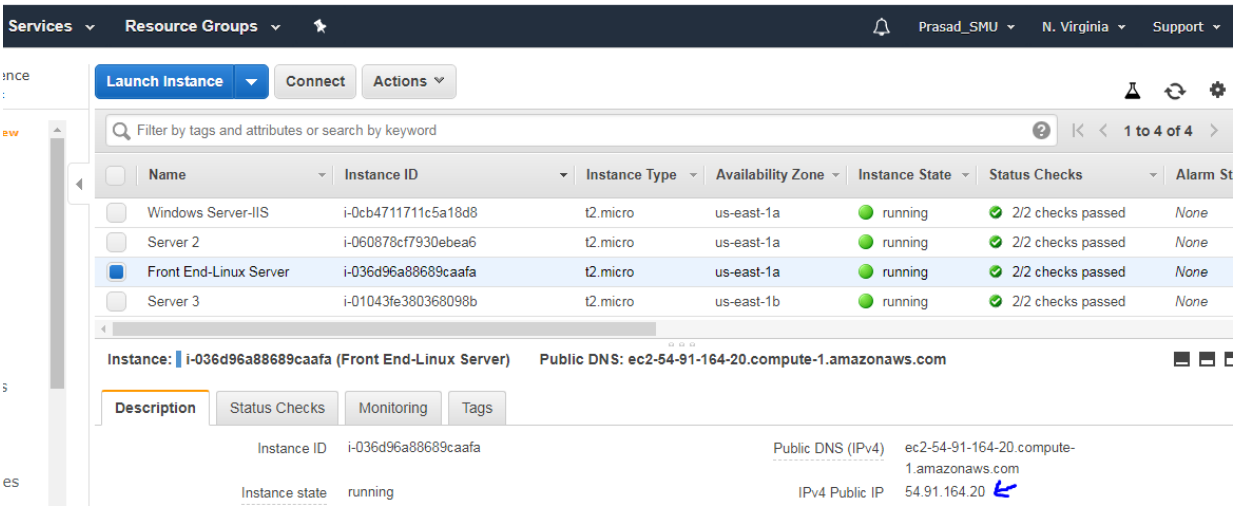
Command: select * from prasadtraining;

```
MySQL [mydb]>
MySQL [mydb]> select * from prasadtraining;
+----------------+-------+------------+
| NAME           | BATCH | PHONE      |
+----------------+-------+------------+
| PRASAD BHAVSAR |     1 | 1234567899 |
| SCOTT KINGSLEY |     2 | 9876543210 |
+----------------+-------+------------+
2 rows in set (0.01 sec)

MySQL [mydb]>
```

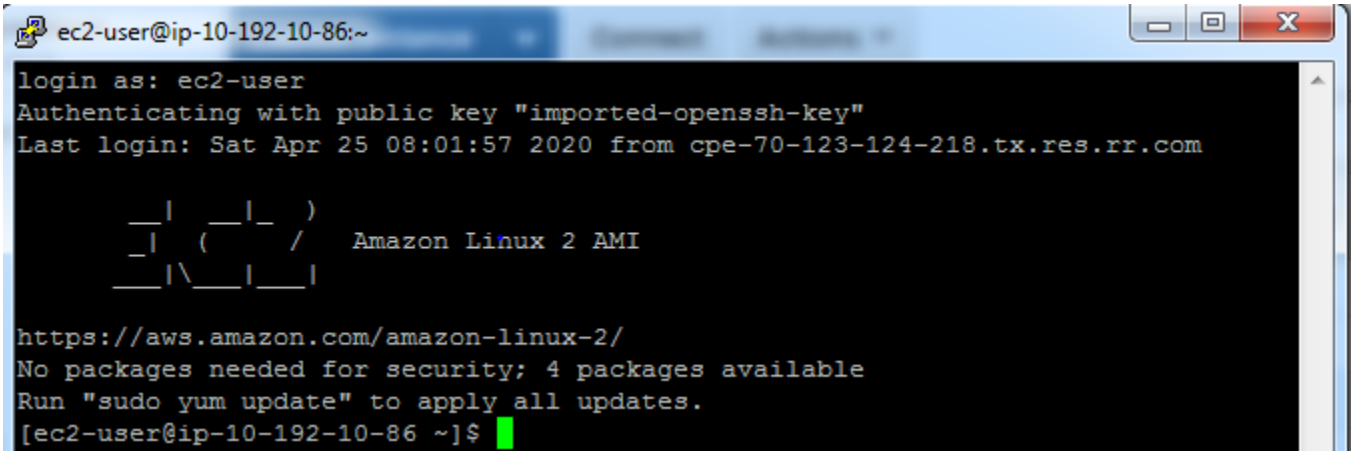
We can see that whatever tables that we had created in “mydb” database on RDS Instance in N. Virginia Region are available on the Read Replica in Mumbai region.

Task 8: Test the Replication

Similarly, lets connect back to the EC2 Instance in N. Virginia region.

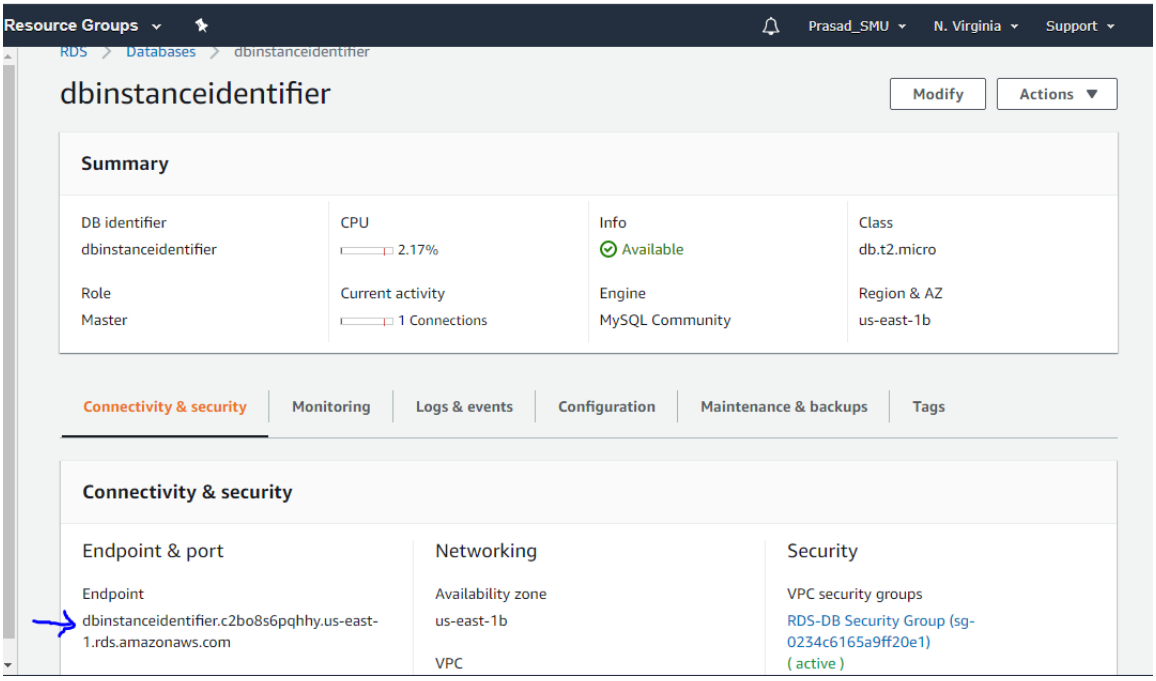


Use the .ppk file that you had converted in the previous lab, and take a PuTTY session.



Connect this EC2 Instance again to the DB Instance running in the N. Virginia Region.

Copy the RDS Instance Endpoint.



Comeback to the PuTTY session of the N. Virginia's Front-End server and run the below command.

Command:

mysql -h dbinstanceidentifier.c2bo8s6pqhhy.us-east-1.rds.amazonaws.com -u master -P 3306 -p

Give the password that you've set.

```
[ec2-user@ip-10-192-10-86 ~]$ mysql -h dbinstanceidentifier.c2bo8s6pqhhy.us-east
-1.rds.amazonaws.com -u master -P 3306 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 186
Server version: 5.7.22-log 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)]>
```

That's it, you've now connected DB Instance to Front End EC2 Instance.

Now check the existing Databases. Run the following Command.

Command: show databases;

```
MySQL [(none)]>  show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| innodb             |
| mydb               |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0.00 sec)

MySQL [(none)]>
```

To get into the get into the "mydb" database, run the following command.

Command: use mydb;

```
MySQL [(none)]> use mydb;
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 [mydb]>
```

Enter the values in the table as follows.

Commands:

insert into prasadtraining values ('RANDY BLYTHE',3,'1122334455');

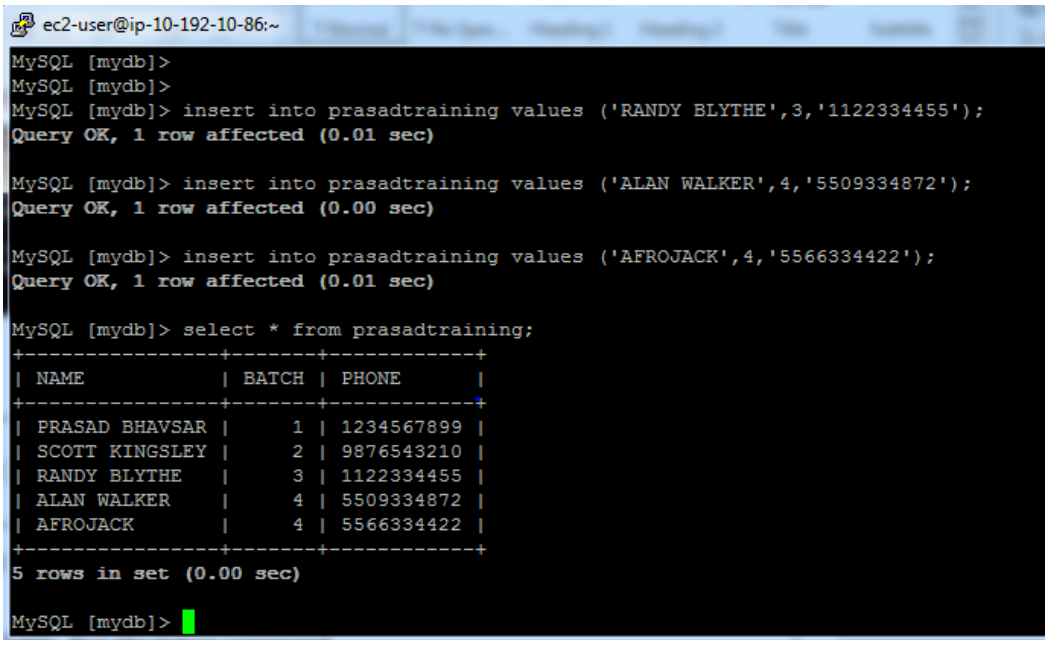
insert into prasadtraining values ('ALAN WALKER',4,'5509334872');

insert into prasadtraining values ('AFROJACK',4,'5566334422');

```
MySQL [mydb]> insert into prasadtraining values ('RANDY BLYTHE',3,'1122334455');
Query OK, 1 row affected (0.01 sec)

MySQL [mydb]> insert into prasadtraining values ('ALAN WALKER',4,'5509334872');
Query OK, 1 row affected (0.00 sec)

MySQL [mydb]> insert into prasadtraining values ('AFROJACK',4,'5566334422');
Query OK, 1 row affected (0.01 sec)

MySQL [mydb]>
```

Let's now verify the Table. Run the below command.

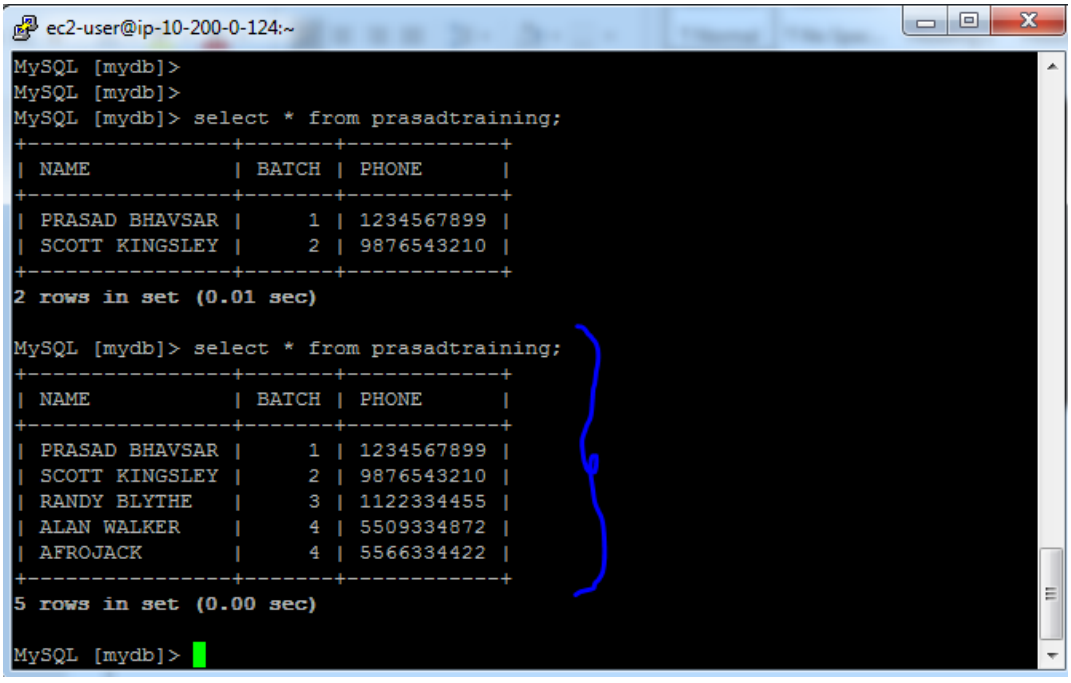
Command: select * from prasadtraining;



Comeback to the PuTTY session of the Mumbai's Front-End server.

Verify the Table. Run the below command.

Command: select * from prasadtraining;

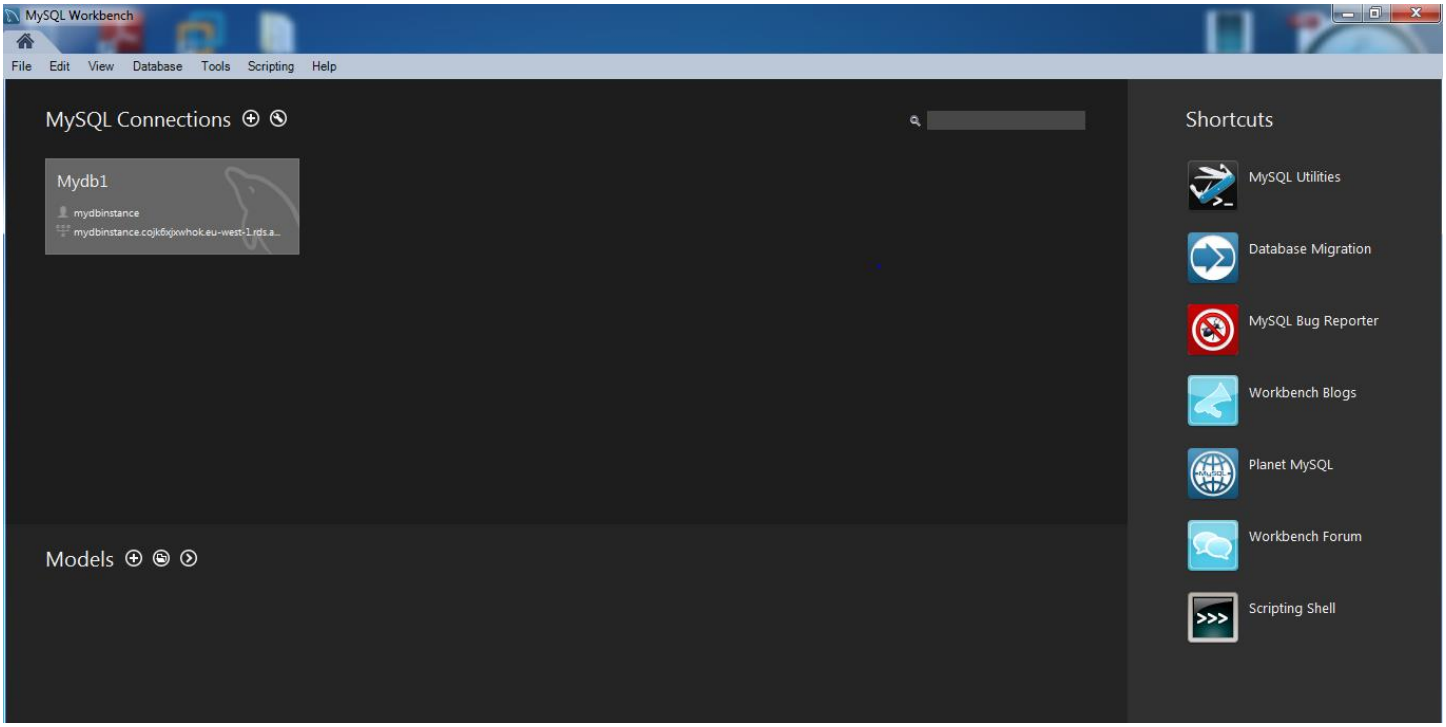


All the Tables are also Available.

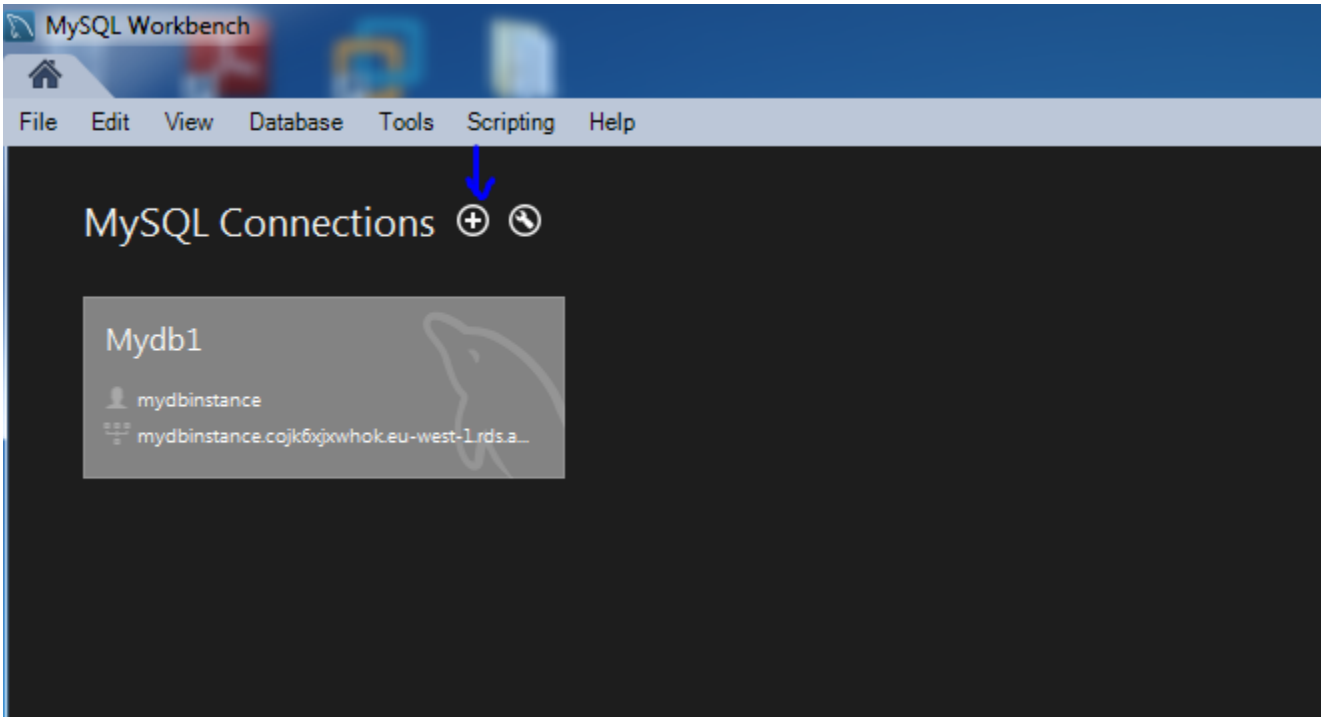
In real world scenario, Data Replication happens asynchronously. It means all the tables gets available immediately on the Read Replica in another Region. Since, in our case the table size very small, the replication happened immediately.

Task 9: Introduction to MySQL Workbench

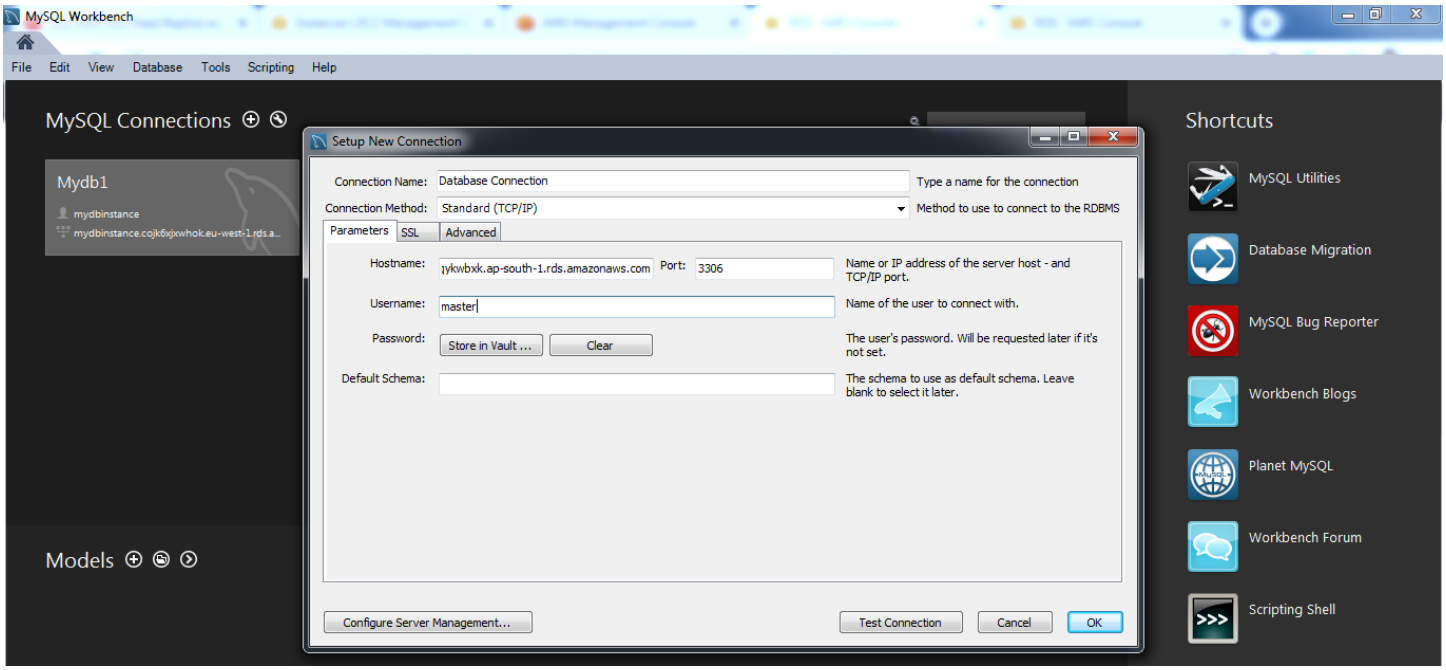
If your DB Instance is in Public Subnets and publicly accessible, then you can use MySQL Workbench software to create tables from your machine.



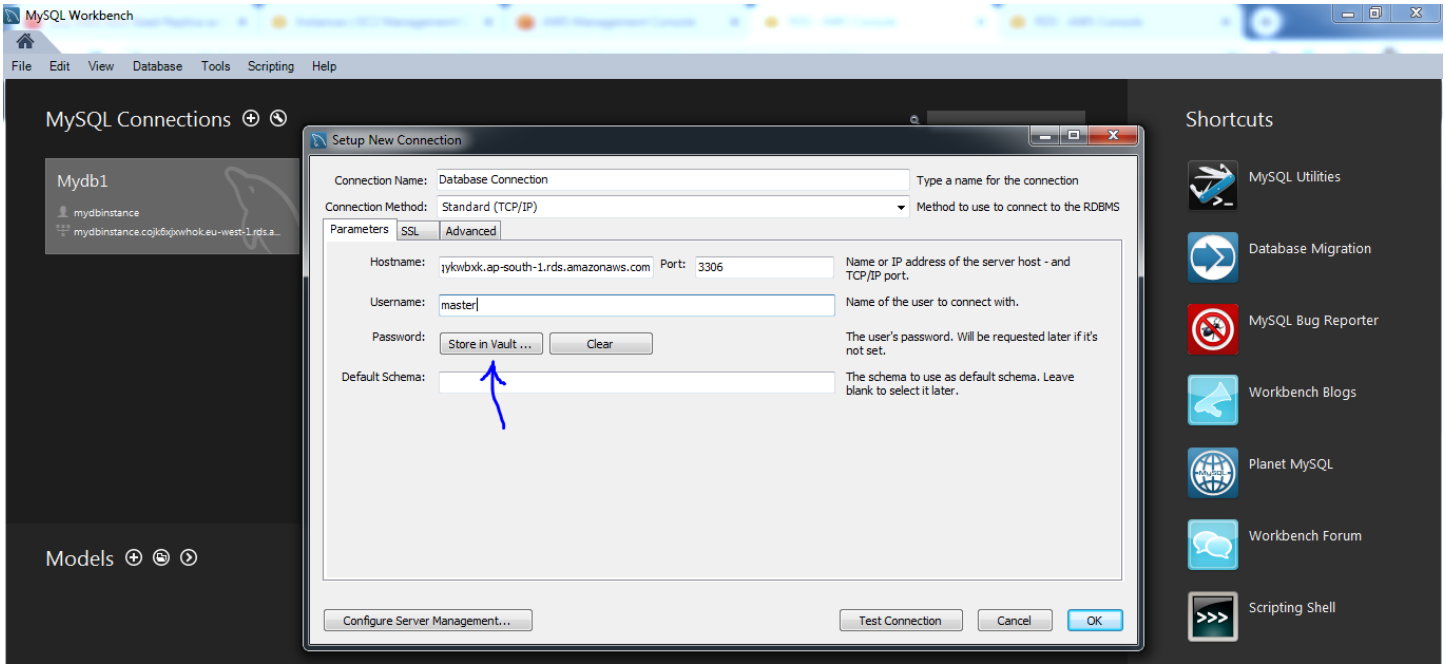
Click on the Plus sign.



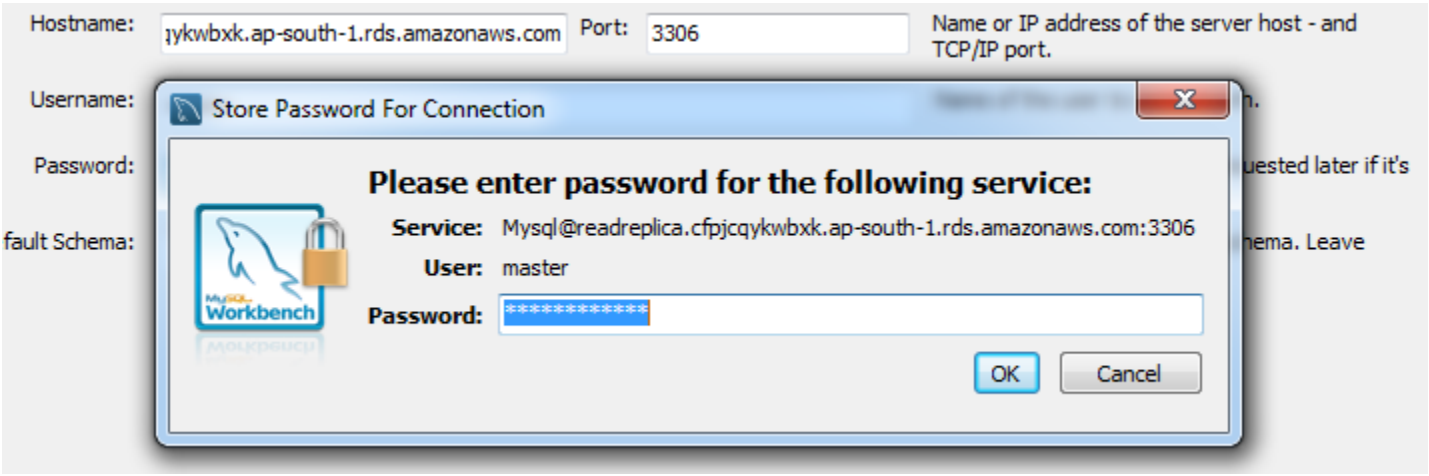
Give the Connection Name of your choice. Specify the Endpoint as Endpoint of the RDS Instance. Give the RDS Instance Username.



Then click on Store in Vault.

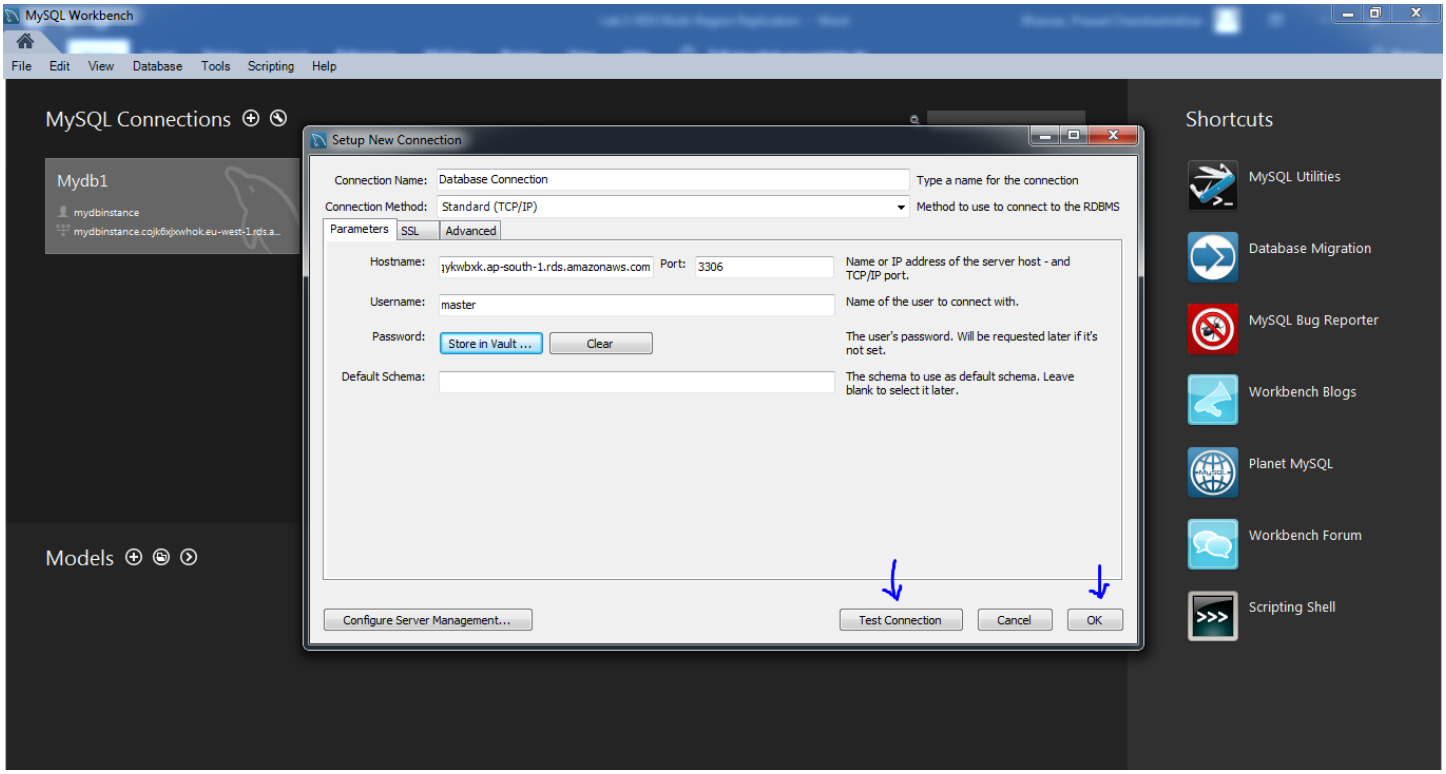


Give the RDS Instance Password and click OK.



Finally, click on Test Connection.

If the Connection is Successful, click on Ok.



Since our all the DB Instances are running in the Private Subnets and are not publicly available, we are not proceeding with the Testing the Connection.

This completes the lab on RDS Multi-Region Replication.

If you have any questions, contact me on pbhavsar@smu.edu.