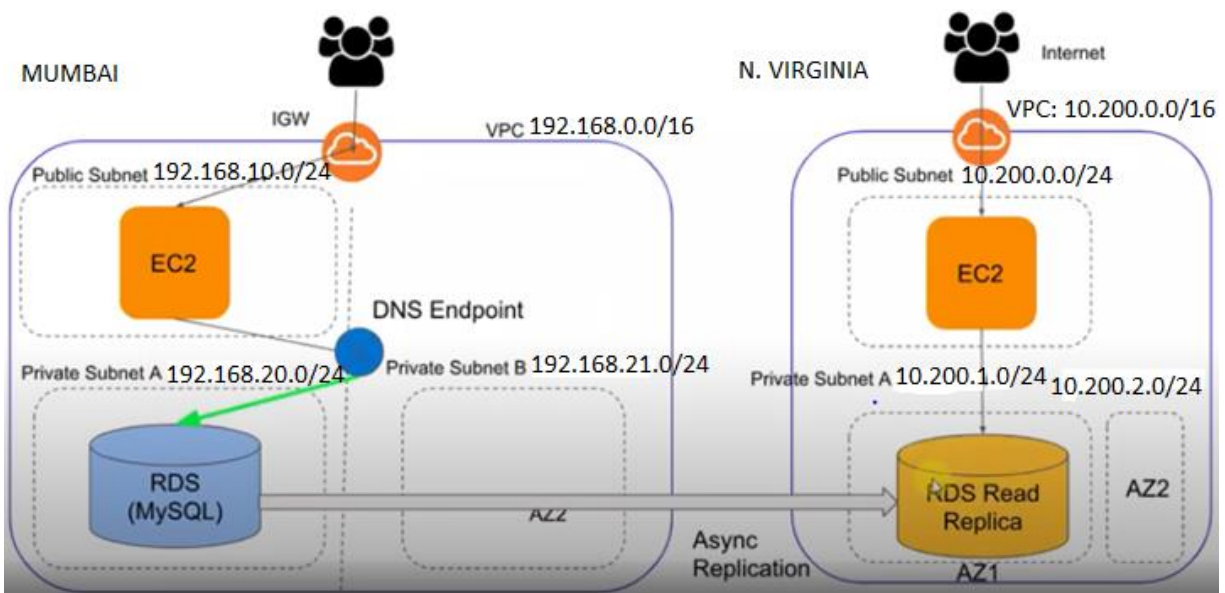


## RDS Multi-Region Replication

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:

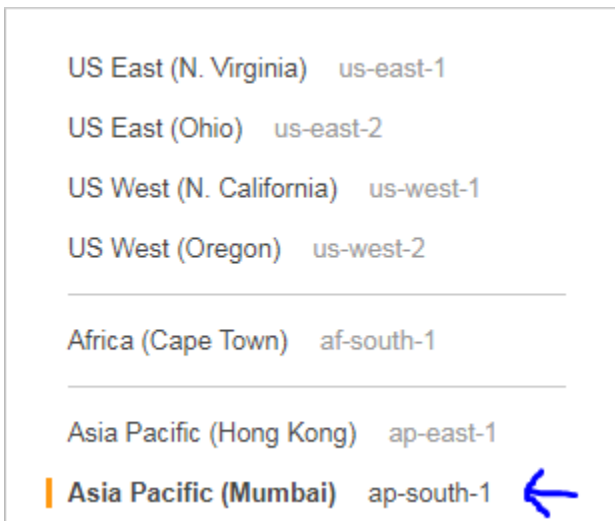
- Task 1: Create a Custom Network Design in another Region
- Task 2: Create a DB Subnet Group
- Task 3: Configure Read Replicas across AWS Region
- Task 4: Launch a Front-End Web Server
- Task 5: Configure Front-End Server with MySQL
- Task 6: Connect the Front-End Servers with RDS DB Instances
- Task 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).

### Create VPC

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range for your VPC. Specify the IPv4 address range as a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16. You can optionally associate an IPv6 CIDR block with the VPC.

Name tag  ⓘ

IPv4 CIDR block\*  ⓘ

IPv6 CIDR block ☒ No IPv6 CIDR Block ⓘ  
☐ Amazon provided IPv6 CIDR block  
☐ IPv6 CIDR owned by me

Tenancy  ⓘ

Click on Create.

Filter by tags and attributes or search by keyword							1 to 2 of 2	
<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set		
<input type="checkbox"/>	Mumbai-Custom-VPC	vpc-00e090e78818d54a9	available	10.200.0.0/16	-	dopt-b26587d9		

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

Give the Name tag.

### 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.

Name tag



\* Required

Cancel

Create

Now click on Create.

Filter by tags and attributes or search by keyword						1 to 2 of 2	
<input type="checkbox"/>	Name	ID	State	VPC	Owner		
<input type="checkbox"/>	Custom Internet Gateway	igw-05c3f7bee8bd...	detached	-	616399057974		

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.

### Attach to VPC

Attach an internet gateway to a VPC to enable communication with the internet. Specify the VPC you would like to attach below.

VPC\*



▶ AWS Command Line Interface command

\* Required

Cancel

Attach

<input type="checkbox"/>	Name	ID	State	VPC	Owner
<input checked="" type="checkbox"/>	Custom Internet Gateway	igw-05c3f7bee8bd...	attached	vpc-00e090e78818d54a9   Mumbai-Custom-VPC	616399057974

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”.

Filter by tags and attributes or search by keyword								
1 to 6 of 6								
<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR	Avai	IPv6	Availability Zone
<input type="checkbox"/>		subnet-3130...	available	vpc-92c8cfa   VPC-B	172.31.32.0/20	4090	-	ap-south-1a
<input checked="" type="checkbox"/>	Private Subnet 1	subnet-034e...	available	vpc-00e090e78818d54a9   Mumbai-Custom-VPC	10.200.1.0/24	251	-	ap-south-1a
<input type="checkbox"/>	Public Subnet	subnet-0b6c...	available	vpc-00e090e78818d54a9   Mumbai-Custom-VPC	10.200.0.0/24	251	-	ap-south-1a
<input type="checkbox"/>	Private Subnet 2	subnet-05cd...	available	vpc-00e090e78818d54a9   Mumbai-Custom-VPC	10.200.2.0/24	251	-	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.

Filter by tags and attributes or search by keyword					
1 to 3 of 3					
<input type="checkbox"/>	Name	Route Table ID	Explicit subnet association	Edge associations	Main
<input checked="" type="checkbox"/>	Private Route Table	rtb-00015ac55c9d3d4b9	2 subnets	-	No

**Route Table:** rtb-00015ac55c9d3d4b9

Summary

Routes

**Subnet Associations**

Edge Associations

Route Propagation

Tags

Edit subnet associations

1 to 2 of 2		
Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-05cd1137e3e2d86...	10.200.2.0/24	-
subnet-034ecd7bd14c643...	10.200.1.0/24	-

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.

### Edit routes

Destination	Target	Status	Propagated
10.200.0.0/16	local	active	No
0.0.0.0/0	igw-05c3f7bee8bd72358		No

[Add route](#)

\* Required [Cancel](#) [Save routes](#)

Click on Save Routes.

Public Route Table rtb-0324920d01c7de84d - - No

Route Table: rtb-0324920d01c7de84d

Summary Routes Subnet Associations Edge Associations Route Propagation Tags

[Edit routes](#)

View All routes

Destination	Target	Status	Propagated
10.200.0.0/16	local	active	No
0.0.0.0/0	igw-05c3f7bee8bd72358	active	No

Associate the one Custom Public Subnets to the Route Table.

Public Route Table rtb-0324920d01c7de84d subnet-0b6c4a15fb543fa4c -

Route Table: rtb-0324920d01c7de84d

Summary Routes Subnet Associations Edge Associations Route Propagation Tags

[Edit subnet associations](#)

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-0b6c4a15fb543fa4c   Public Subnet	10.200.0.0/24	-

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.

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
MySQL/Aurora	TCP	3306	Custom	10.200.0.0/24
				e.g. SSH for Admin Desktop

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.

RDS-SG sg-091eb3cb01dc... RDS-SG vpc-00e090e7881... EC2-VPC RDS-SG

Security Group: sg-091eb3cb01dce0c78

Description Inbound Rules Outbound Rules Tags

Edit rules

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
MySQL/Aurora	TCP	3306	10.200.0.0/24	

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.

EC2-SG sg-0ebad45f6765... EC2-SG vpc-00e090e7881... EC2-VPC EC2-SG

Security Group: sg-0ebad45f67653d6b5

Description Inbound Rules Outbound Rules Tags

Edit rules

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
SSH	TCP	22	70.123.124.218/32	

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.

**Subnet group details**

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

**Description**

**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.

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

**Add subnets**

**Availability Zones**  
Choose the Availability Zones that include the subnets you want to add.  
  

ap-south-1a

ap-south-1b

**Subnets**  
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.  
  

subnet-05cd1137e3e2d8629 (10.200.2.0/24)

subnet-034ecd7bd14c64348 (10.200.1.0/24)

**Subnets selected (2)**

Availability zone	Subnet ID	CIDR block
ap-south-1a	subnet-034ecd7bd14c64348	10.200.1.0/24
ap-south-1b	subnet-05cd1137e3e2d8629	10.200.2.0/24

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.

The screenshot shows the AWS Management Console interface for a specific DB instance. The breadcrumb navigation indicates the path: RDS > Databases > dbinstanceidentifier. The instance name 'dbinstanceidentifier' is displayed prominently. To the right of the name are 'Modify' and 'Actions' buttons. Below this is a 'Summary' section containing a table with the following data:

DB identifier dbinstanceidentifier	CPU 2.30%	Info Available	Class db.t2.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ us-east-1b

Now click on Actions and click on Create Read Replica.

This screenshot shows the same console view as the previous one, but with the 'Actions' dropdown menu open. The menu lists several options: Stop, Reboot, Delete, Create read replica (highlighted with a blue arrow), Create Aurora read replica, Promote, Take snapshot, Restore to point in time, and Migrate snapshot. Below the summary table, there are tabs for 'Connectivity & security', 'Monitoring', 'Logs & events', 'Configuration', and 'Maintenance & backups'.



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.

**Network & Security**

**Destination region**  
The region in which the replica will be launched

Asia Pacific (Mumbai) ▼

**Destination DB subnet group**

dbsubnetgroupmumbai ▼

**Availability zone**  
The EC2 Availability Zone that the database instance will be created in.

ap-south-1a ▼

**Publicly accessible**

☐ Yes  
EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

☒ No  
DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

**VPC security groups**

Choose VPC security groups ▼

RDS-SG X

Give the DB Instance Identifier Name of your choice.

**Settings**

**Read replica source**  
Source DB instance Identifier

dbinstanceidentifier ▼

**DB instance identifier**  
DB instance identifier. This is the unique key that identifies a DB instance. This parameter is stored as a lowercase string (e.g. mydbinstance).

readreplica

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.

**Instance specifications**

**DB instance class**  
Contains the compute and memory capacity of the DB instance.

db.t2.micro — 1 vCPU, 1 GiB RAM ▼

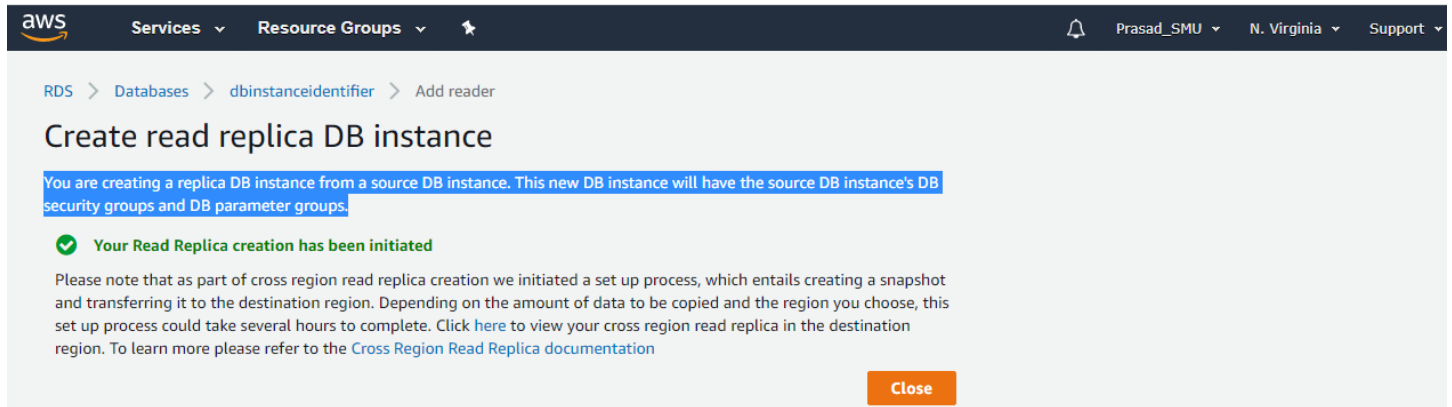
→ **Multi-AZ deployment**  
Specifies if the DB instance should have a standby deployed in another availability zone.

☐ Yes

☒ No

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.

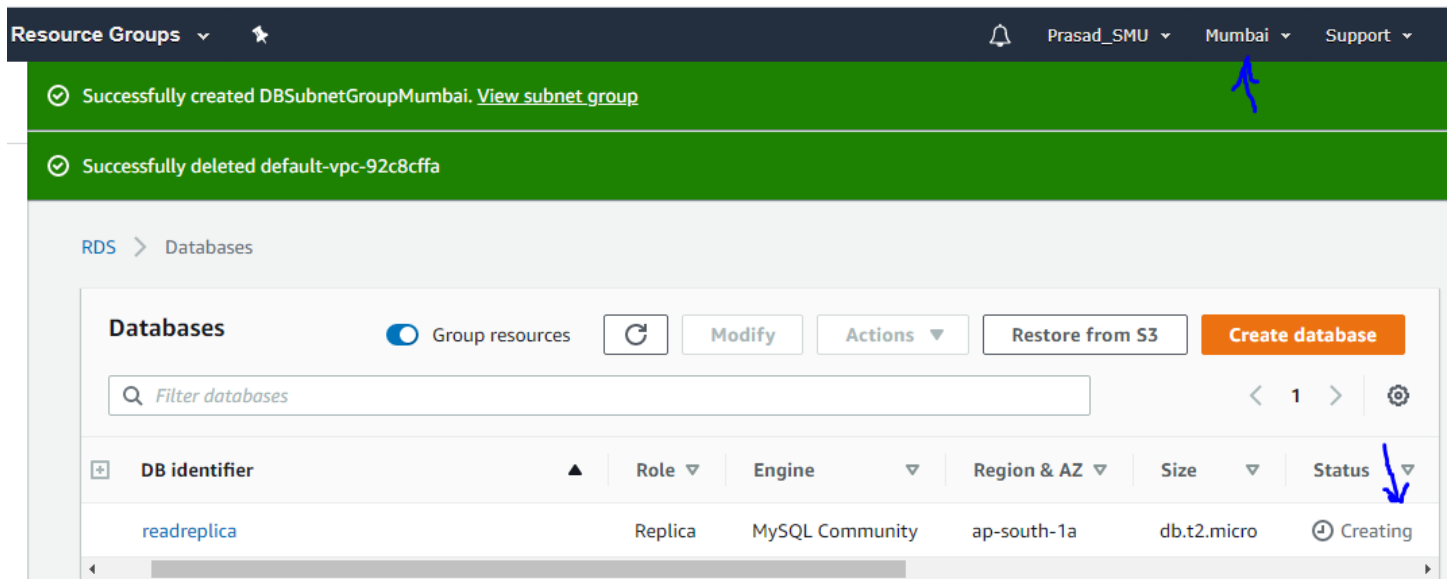
Finally, review all the configurations and click on Create Read Replica.



The screenshot shows the AWS console interface. At the top, the navigation bar includes the AWS logo, 'Services', 'Resource Groups', and a user profile 'Prasad\_SMU' in the 'N. Virginia' region. Below the navigation bar, the breadcrumb trail is 'RDS > Databases > dbinstanceidentifier > Add reader'. The main heading is 'Create read replica DB instance'. A blue highlighted box contains the text: 'You are creating a replica DB instance from a source DB instance. This new DB instance will have the source DB instance's DB security groups and DB parameter groups.' Below this, a green checkmark icon is followed by the text 'Your Read Replica creation has been initiated'. A paragraph of text explains the cross-region read replica creation process, mentioning snapshot creation and data transfer. A 'Close' button is located at the bottom right of the notification area.

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.



The screenshot shows the AWS console interface. At the top, the navigation bar includes the 'Resource Groups' dropdown, a user profile 'Prasad\_SMU', and the 'Mumbai' region. Below the navigation bar, there are two green success messages: 'Successfully created DBSubnetGroupMumbai. View subnet group' and 'Successfully deleted default-vpc-92c8cffa'. The main content area is titled 'Databases' and includes a 'Group resources' toggle, a 'Filter databases' search bar, and buttons for 'Modify', 'Actions', 'Restore from S3', and 'Create database'. A table lists the databases. The first row is 'readreplica' with a status of 'Creating'. A blue arrow points to the 'Status' column header, and another blue arrow points to the 'Creating' status of the 'readreplica'.

DB identifier	Role	Engine	Region & AZ	Size	Status
readreplica	Replica	MySQL Community	ap-south-1a	db.t2.micro	Creating


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**.

  
Amazon Linux  
Free tier eligible

**Amazon Linux 2 AMI (HVM), SSD Volume Type** - ami-0470e33cd681b2476 (64-bit x86) / ami-01116f0e8a6908170 (64-bit Arm)  
Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.  
Root device type: ebs    Virtualization type: hvm    ENA Enabled: Yes

[Select](#)  
☒ 64-bit (x86)  
☐ 64-bit (Arm)

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.

### Step 3: Configure Instance Details

→ Network	<input type="text" value="vpc-00e090e78818d54a9   Mumbai-Custom-VPC"/>	<a href="#">Create new VPC</a>
→ Subnet	<input type="text" value="subnet-0b6c4a15fb543fa4c   Public Subnet   ap-sout"/> 251 IP Addresses available	<a href="#">Create new subnet</a>
→ Auto-assign Public IP	<input type="text" value="Enable"/>	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	<input type="text" value="Open"/>	<a href="#">Create new Capacity Reservation</a>
IAM role	<input type="text" value="None"/>	<a href="#">Create new IAM role</a>
Shutdown behavior	<input type="text" value="Stop"/>	
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior	
→ Enable termination protection	<input checked="" type="checkbox"/> Protect against accidental termination	
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring <a href="#">Additional charges apply.</a>	

Select the Custom Security Group that we've created.

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☐ Create a new security group  
☒ Select an existing security group

Security Group ID	Name	Description	Actions
<input type="radio"/> sg-01a8cb8a0ea2cbbec	default	default VPC security group	<a href="#">Copy to new</a>
<input checked="" type="radio"/> sg-0ebad45f67653d6b5	EC2-SG	EC2-SG	<a href="#">Copy to new</a>
<input type="radio"/> sg-091eb3cb01dce0c78	RDS-SG	RDS-SG	<a href="#">Copy to new</a>

Inbound rules for sg-0ebad45f67653d6b5 (Selected security groups: sg-0ebad45f67653d6b5)

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	70.123.124.218/32	

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.

Mumbai-Front End Server

i-0b09aec945f449db3

t2.micro

ap-south-1a

running

2/2 checks ...

None

Instance: **i-0b09aec945f449db3 (Mumbai-Front End Server)** Public IP: **3.6.89.78**

Description

Status Checks

Monitoring

Tags

Instance ID

i-0b09aec945f449db3

Public DNS (IPv4)

-

Instance state

running

IPv4 Public IP

3.6.89.78

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

Resource Groups

Prasad\_SMU Mumbai Support

RDS > Databases > readreplica

readreplica Modify Actions

Summary

DB identifier

readreplica

CPU

1.53%

Role

Replica

Current activity

0 Connections

Info

Available

Engine

MySQL Community

Class

db.t2.micro

Region & AZ

ap-south-1a

## Task 5: Configure Front-End Server with MySQL

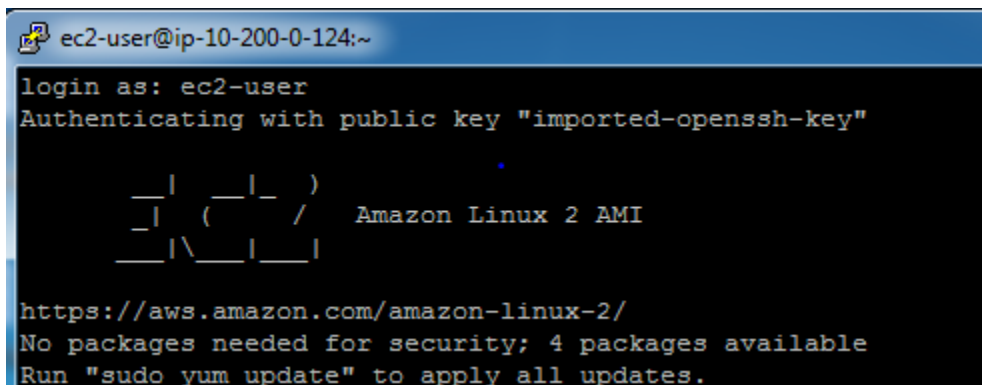
First, we'll connect the Front-End server with DB Instances in Mumbai (ap-south-1) region.

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".



```
ec2-user@ip-10-200-0-124:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
  _ | _ | _ )  
  _ | ( _ /  Amazon Linux 2 AMI  
  _ | \ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
No packages needed for security; 4 packages available  
Run "sudo yum update" to apply all updates.
```

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.

The screenshot shows the AWS Management Console interface for an RDS Read Replica instance named 'readreplica'. The instance is in the 'ap-south-1' region. The 'Summary' tab is selected, showing the following details:

DB identifier	CPU	Info	Class
readreplica	1.53%	Available	db.t2.micro
Role	Current activity	Engine	Region & AZ
Replica	0 Connections	MySQL Community	ap-south-1a

Below the summary, the 'Connectivity & security' tab is selected, showing the following details:

Endpoint & port	Networking	Security
Endpoint readreplica.cfpjcqykwbxk.ap-south-1.rds.amazonaws.com	Availability zone ap-south-1a	VPC security groups RDS-SG (sg-091eb3cb01dce0c78) (active)

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 "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.

The screenshot shows the AWS Management Console interface. At the top, there are navigation tabs for Services, Resource Groups, and a search bar. Below this, there's a section for 'Instances' with a 'Launch Instance' button and a 'Connect' button. A table lists several instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm St
Windows Server-IIS	i-0cb4711711c5a18d8	t2.micro	us-east-1a	running	2/2 checks passed	None
Server 2	i-060878cf7930ebea6	t2.micro	us-east-1a	running	2/2 checks passed	None
Front End-Linux Server	i-036d96a88689caafa	t2.micro	us-east-1a	running	2/2 checks passed	None
Server 3	i-01043fe380368098b	t2.micro	us-east-1b	running	2/2 checks passed	None

Below the table, the details for the selected instance 'Front End-Linux Server' (Instance ID: i-036d96a88689caafa) are shown. The Public DNS is ec2-54-91-164-20.compute-1.amazonaws.com. The instance state is running. The IPv4 Public IP is 54.91.164.20.

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

```

ec2-user@ip-10-192-10-86:~
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Sat Apr 25 08:01:57 2020 from cpe-70-123-124-218.tx.res.rr.com

  _ | _ | _ )
  _ | ( _ | /   Amazon Linux 2 AMI
  _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 4 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-192-10-86 ~]$
  
```

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

Copy the RDS Instance Endpoint.

The screenshot shows the AWS Management Console for the RDS instance 'dbinstanceidentifier'. The instance is in the 'N. Virginia' region. The 'Summary' tab is selected, showing the following details:

Summary			
DB identifier dbinstanceidentifier	CPU 2.17%	Info Available	Class db.t2.micro
Role Master	Current activity 1 Connections	Engine MySQL Community	Region & AZ us-east-1b

Below the summary, the 'Connectivity & security' tab is selected, showing the following details:

Connectivity & security		
Endpoint & port Endpoint dbinstanceidentifier.c2bo8s6pqhhy.us-east-1.rds.amazonaws.com	Networking Availability zone us-east-1b VPC	Security VPC security groups RDS-DB Security Group (sg-0234c6165a9ff20e1) (active)

A blue arrow points to the endpoint URL: dbinstanceidentifier.c2bo8s6pqhhy.us-east-1.rds.amazonaws.com.

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 "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;

```
ec2-user@ip-10-192-10-86:~  
MySQL [mydb]>  
MySQL [mydb]>  
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]> select * from prasadtraining;  
+-----+-----+-----+  
| NAME          | BATCH | PHONE      |  
+-----+-----+-----+  
| PRASAD BHAVSAR | 1     | 1234567899 |  
| SCOTT KINGSLEY | 2     | 9876543210 |  
| RANDY BLYTHE   | 3     | 1122334455 |  
| ALAN WALKER    | 4     | 5509334872 |  
| AFROJACK       | 4     | 5566334422 |  
+-----+-----+-----+  
5 rows in set (0.00 sec)  
  
MySQL [mydb]>
```

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

Verify the Table. Run the below command.

**Command:** select \* from prasadtraining;

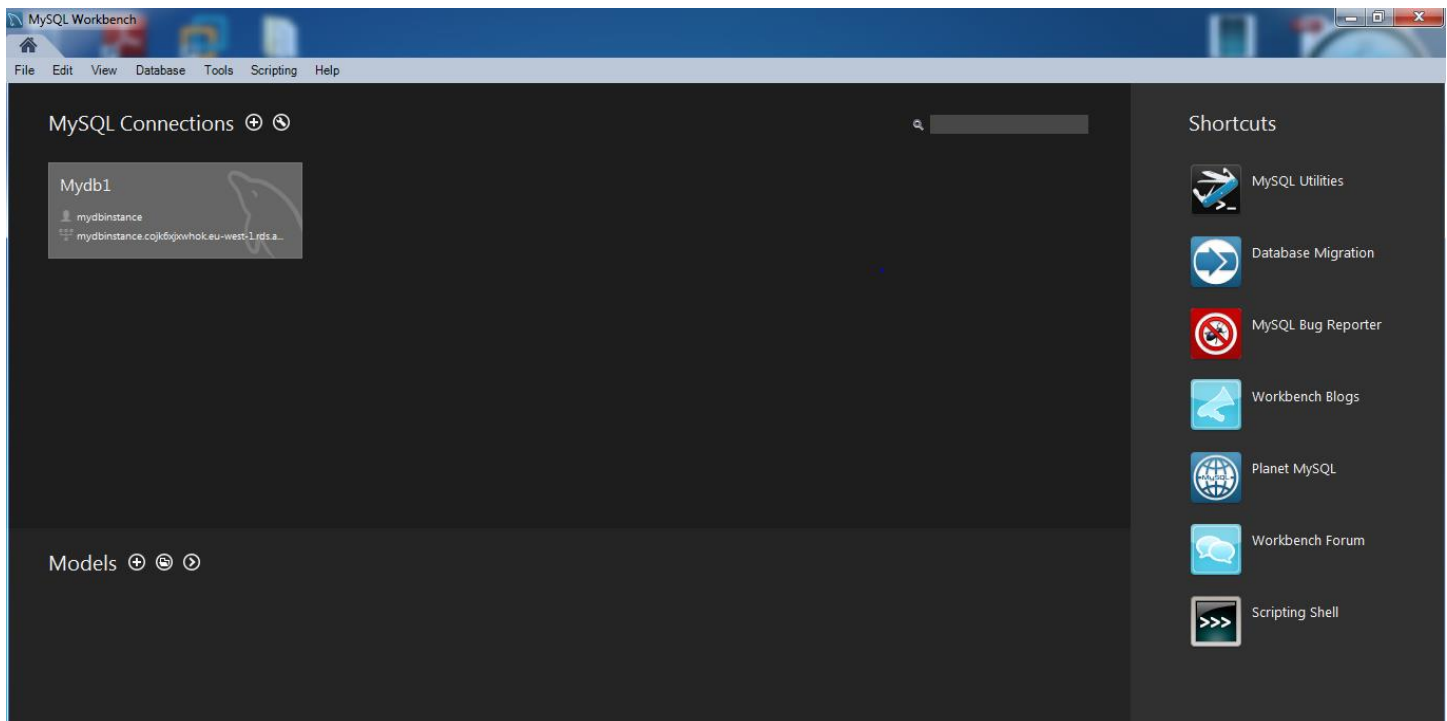
```
ec2-user@ip-10-200-0-124:~  
MySQL [mydb]>  
MySQL [mydb]>  
MySQL [mydb]> select * from prasadtraining;  
+-----+-----+-----+  
| NAME          | BATCH | PHONE      |  
+-----+-----+-----+  
| PRASAD BHAVSAR | 1     | 1234567899 |  
| SCOTT KINGSLEY | 2     | 9876543210 |  
+-----+-----+-----+  
2 rows in set (0.01 sec)  
  
MySQL [mydb]> select * from prasadtraining;  
+-----+-----+-----+  
| NAME          | BATCH | PHONE      |  
+-----+-----+-----+  
| PRASAD BHAVSAR | 1     | 1234567899 |  
| SCOTT KINGSLEY | 2     | 9876543210 |  
| RANDY BLYTHE   | 3     | 1122334455 |  
| ALAN WALKER    | 4     | 5509334872 |  
| AFROJACK       | 4     | 5566334422 |  
+-----+-----+-----+  
5 rows in set (0.00 sec)  
  
MySQL [mydb]>
```

All the Tables are also Available.

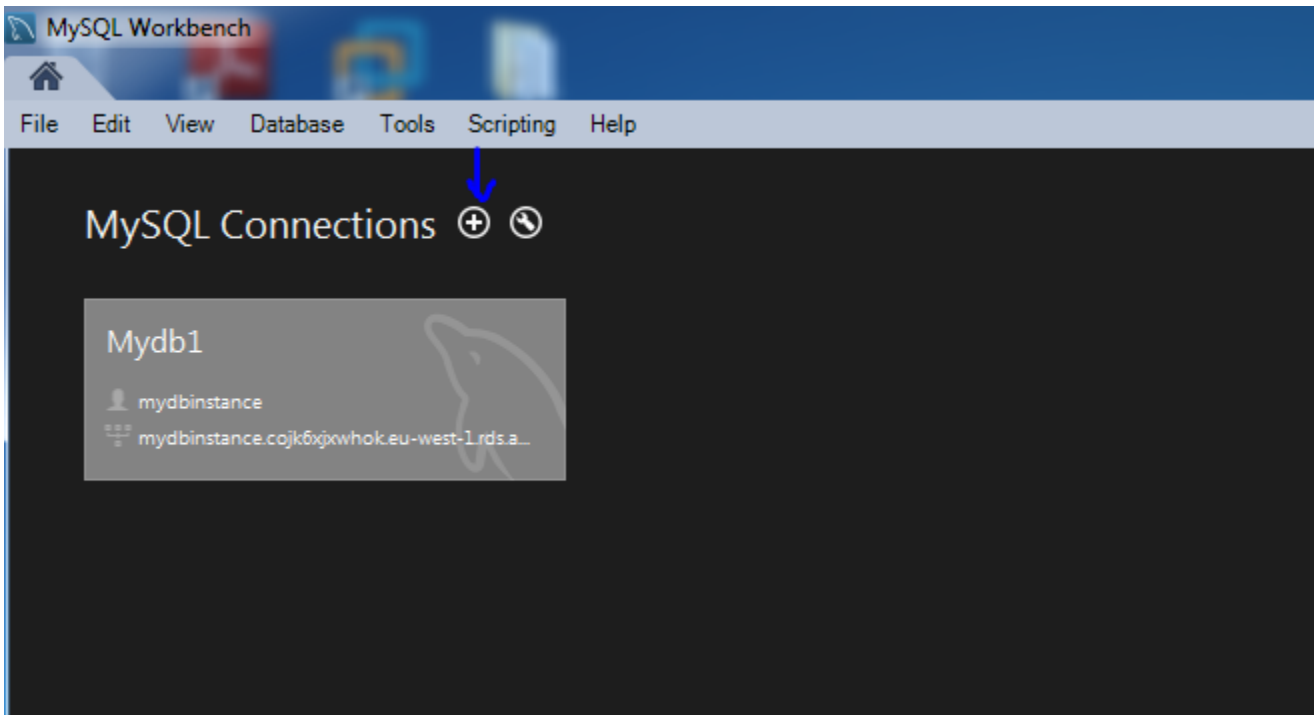
In real world scenario, Data Replication happens asynchronously. It means all the tables gets available immediately on the Read Replica in another Region. Since, in our case the table size very small, the replication happened immediately.

## **Task 9: Introduction to MySQL Workbench**

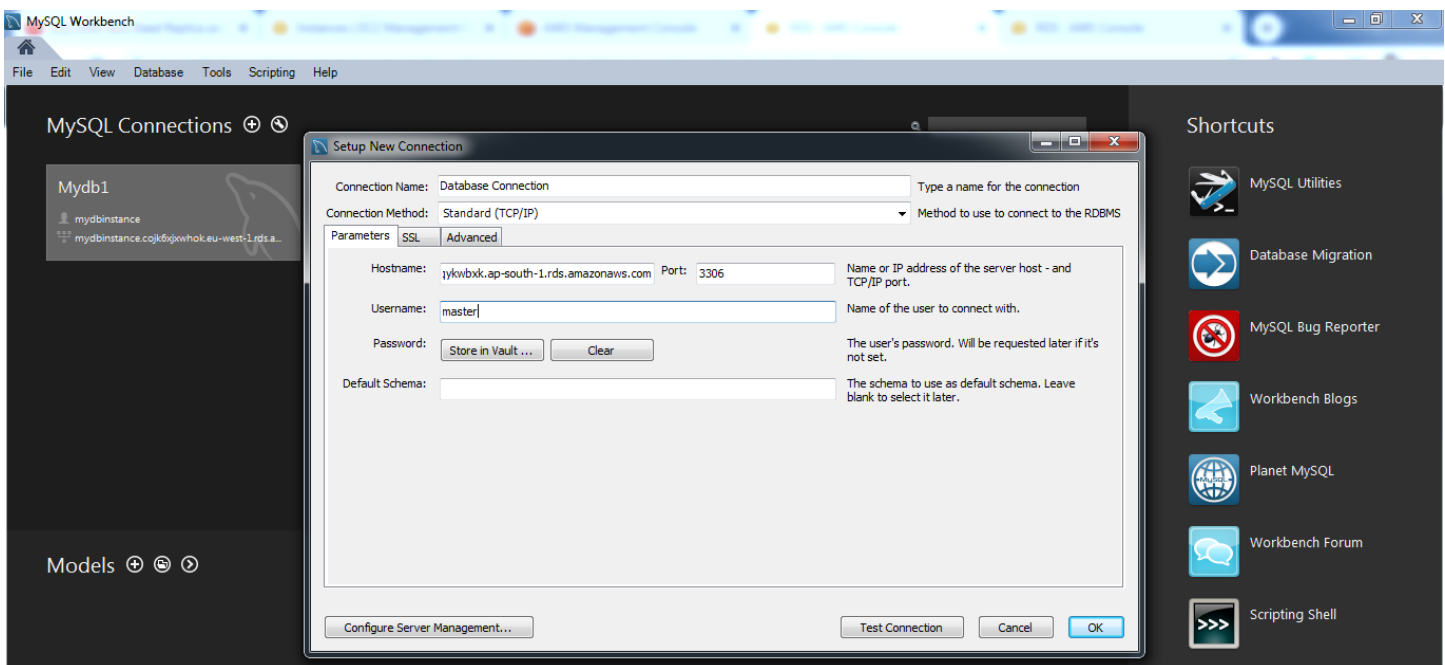
If your DB Instance is in Public Subnets and publicly accessible, then you can use MySQL Workbench software to create tables from your machine.



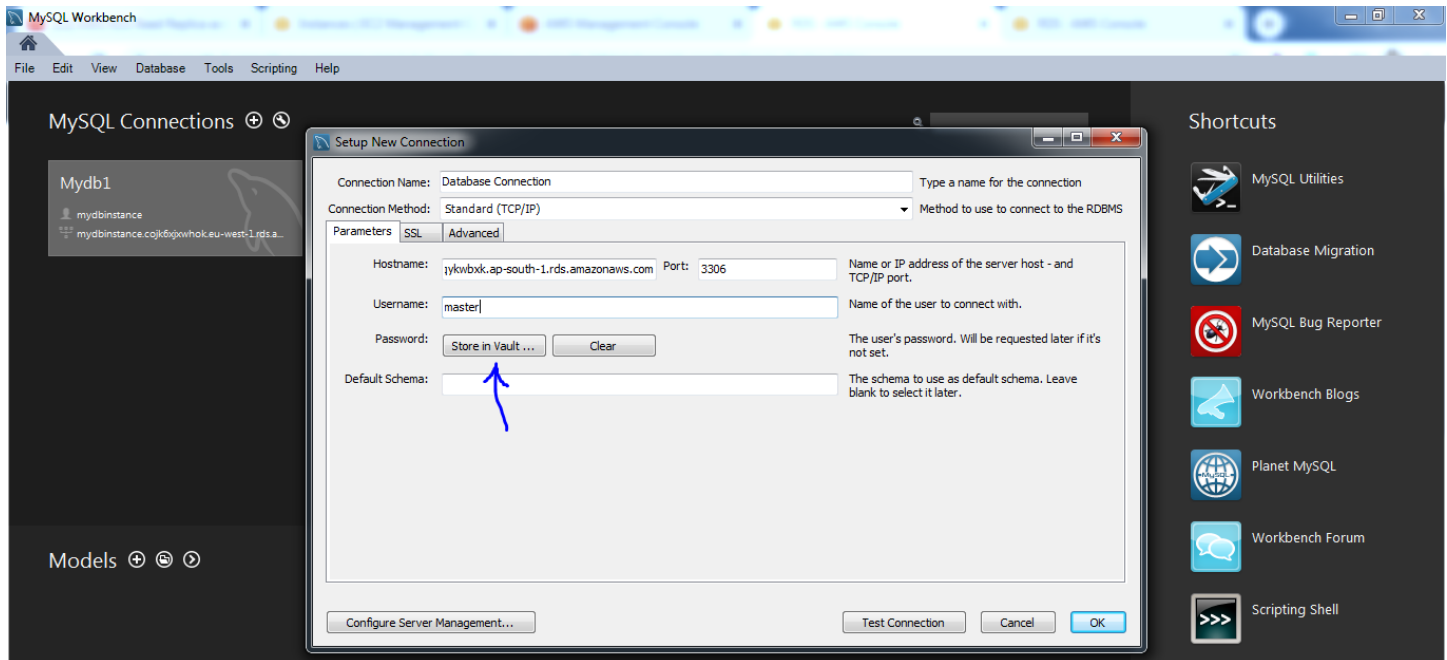
Click on the Plus sign.



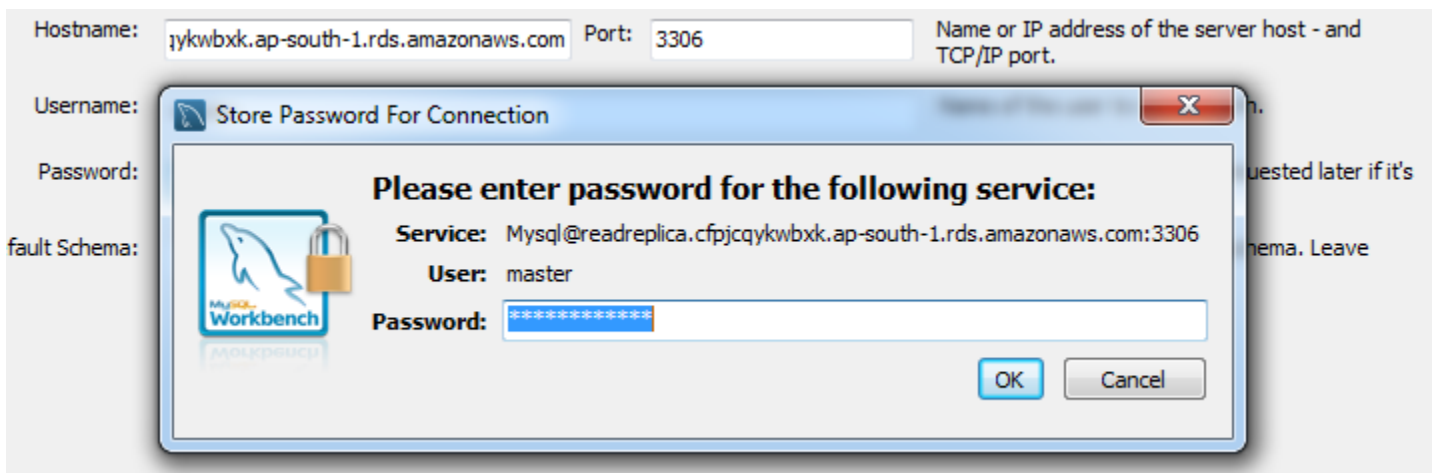
Give the Connection Name of your choice. Specify the Endpoint as Endpoint of the RDS Instance. Give the RDS Instance Username.



Then click on Store in Vault.



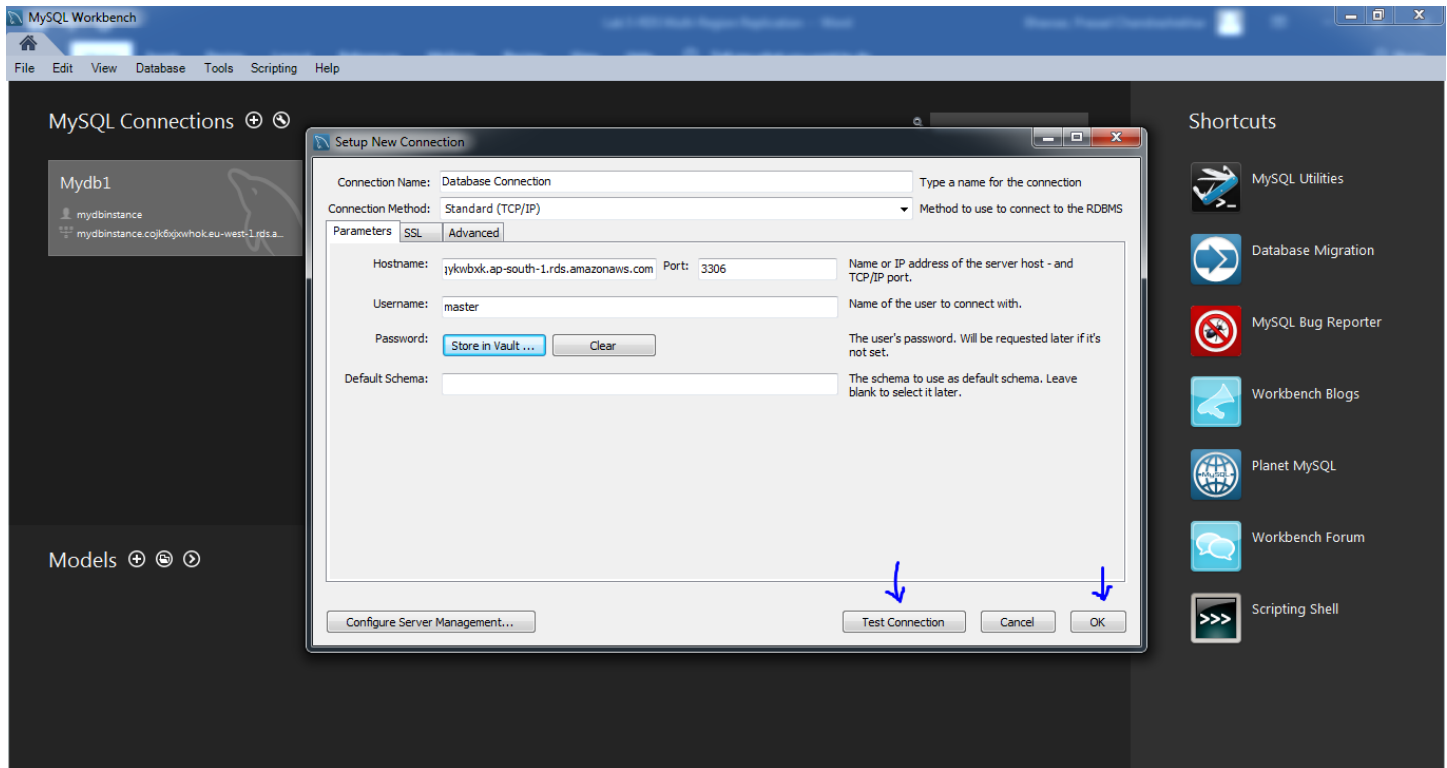
Give the RDS Instance Password and click OK.



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](mailto:pbhavsar@smu.edu).