**PROJECT**

**BUILD A 3 TIER REAL WORLD APPLICATION ON THE CLOUD**

**Build a 3 tier Real World Application on the Cloud**

**AIM**

The set up includes EC2 instance running as Front end Application Server and

connecting to an Amazon RDS service, running as back end database.

 The application has to be successfully installed on the App Server and then it should

communicate with RDS as well.

 Features of RDS such as Multi AZ deployment, automated backups are used in this

project.

**PREREQUISITS**

Launching an ec2 instance

Installing apache with PHP

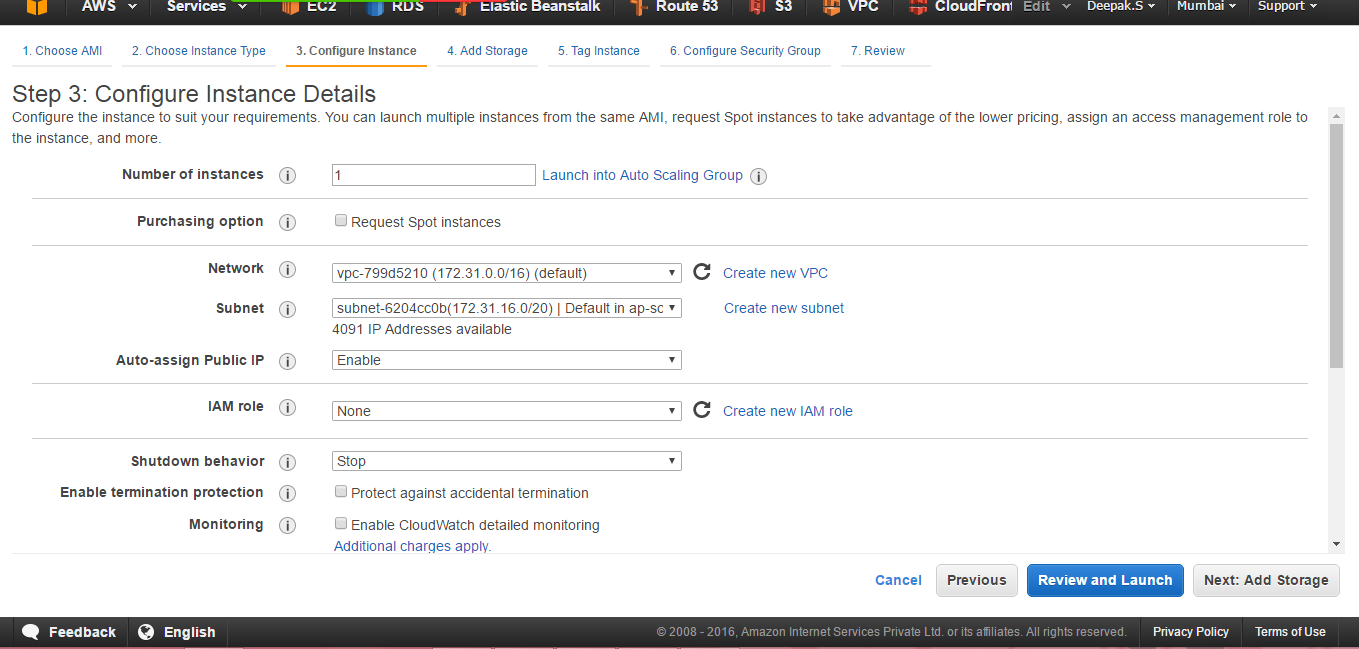
Launching an RDS-My sql instance

Basic Linux commands

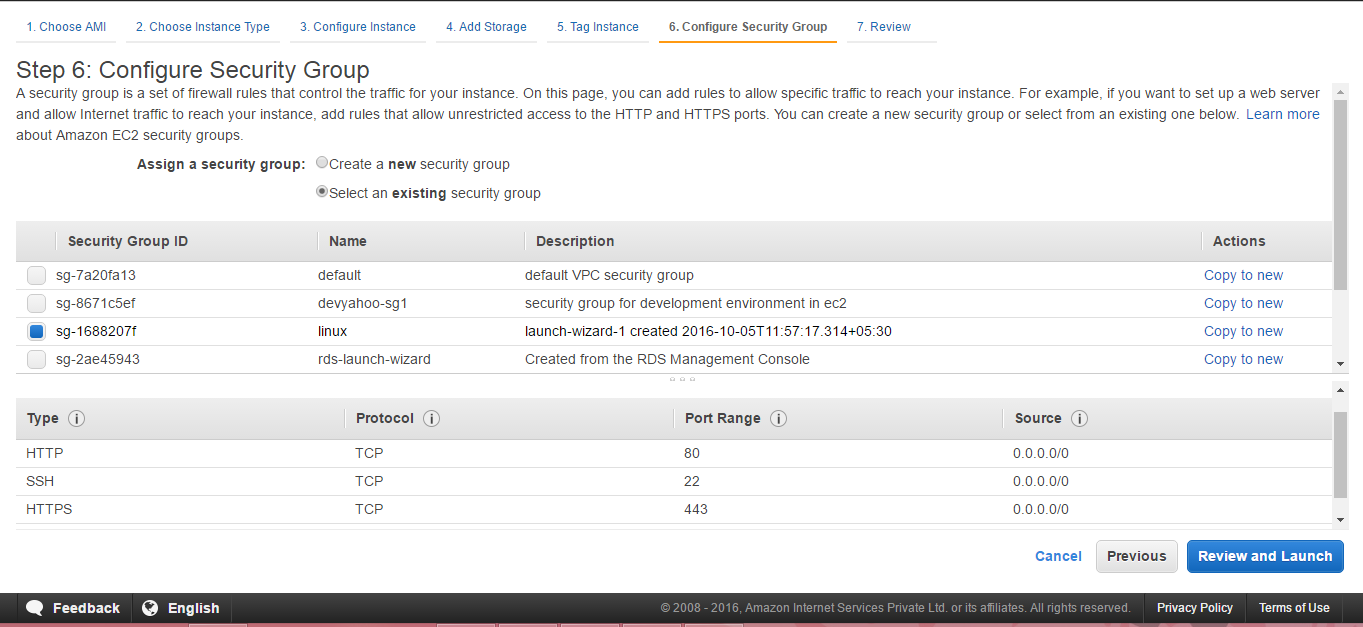
**STEPS**

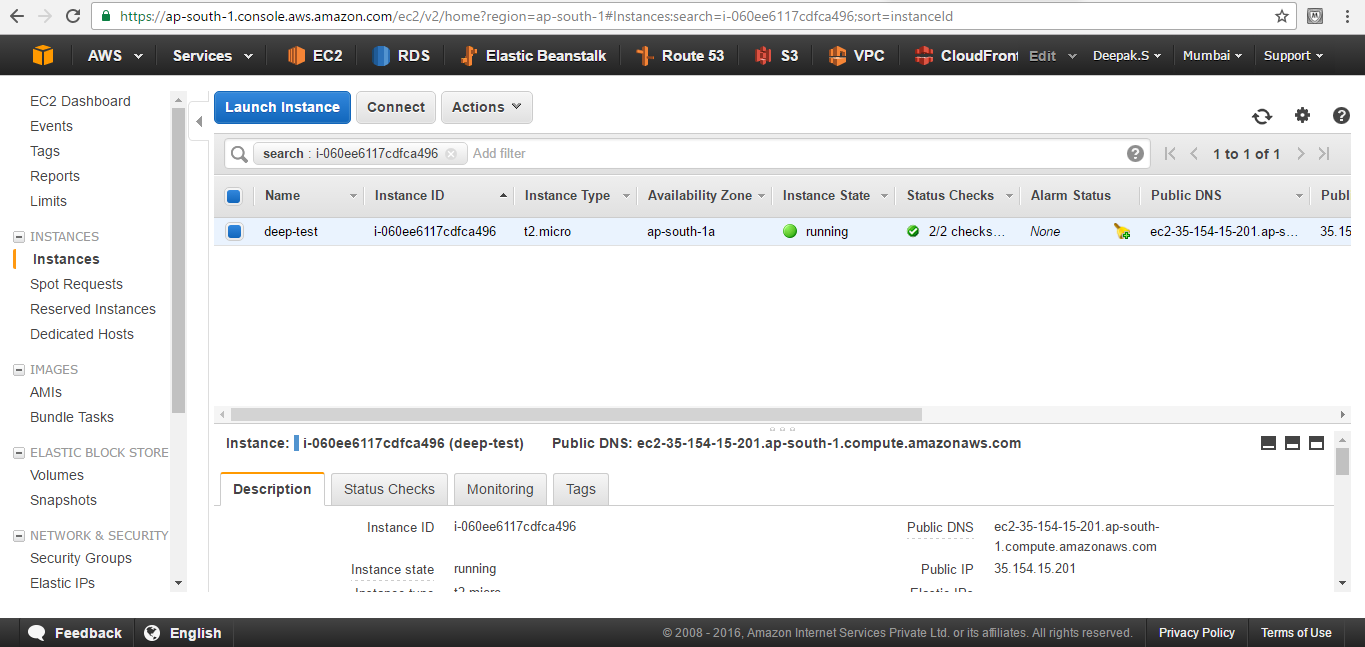
Install an Apache webserver with PHP

1.Choose ubuntu 14.04 AMI and configure instance details in freetier as below



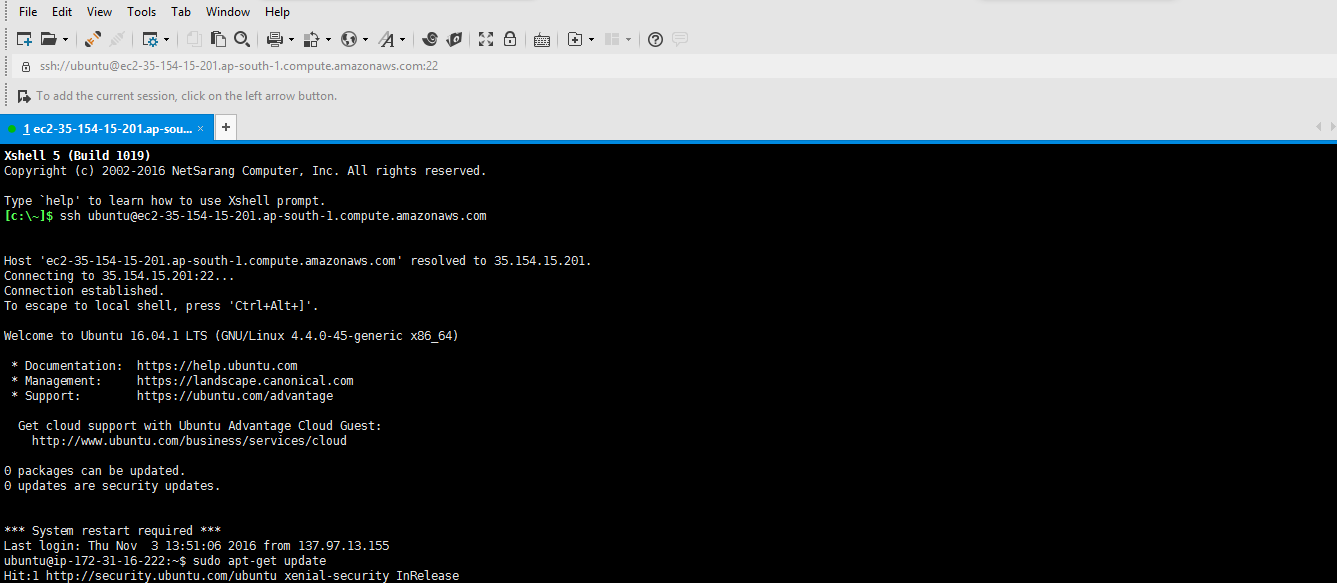
2.In the security group set ssh,http,https to open as below and proceed creating the instance



3.Launch the instance once the status checks are completed .Connect to the AMI by ssh ubuntu@<public dns> in Xshell 5 

4.Type the following commands in Xshell to install apache with php

Login to the AMI by typing ssh ubuntu@<public DNS of ur ec2 instance(deep-test) here> as shown below



sudo apt-get -y update

sudo apt-get -y upgrade

sudo apt install -y yum

sudo apt install -y yum-utils

sudo apt-get install -y apache2

//// If the above command is not working try with following

sudo apt-get install apache2

//// and later type y when asked.

sudo apt-get install -y mysql-client-core-5.6

sudo apt-get install php5 libapache2-mod-php5 ( Used on 14.04 only ) and type y

apt-get install php5-mysql //// You must install this package otherwise PHP will not connect to (MySql) RDS

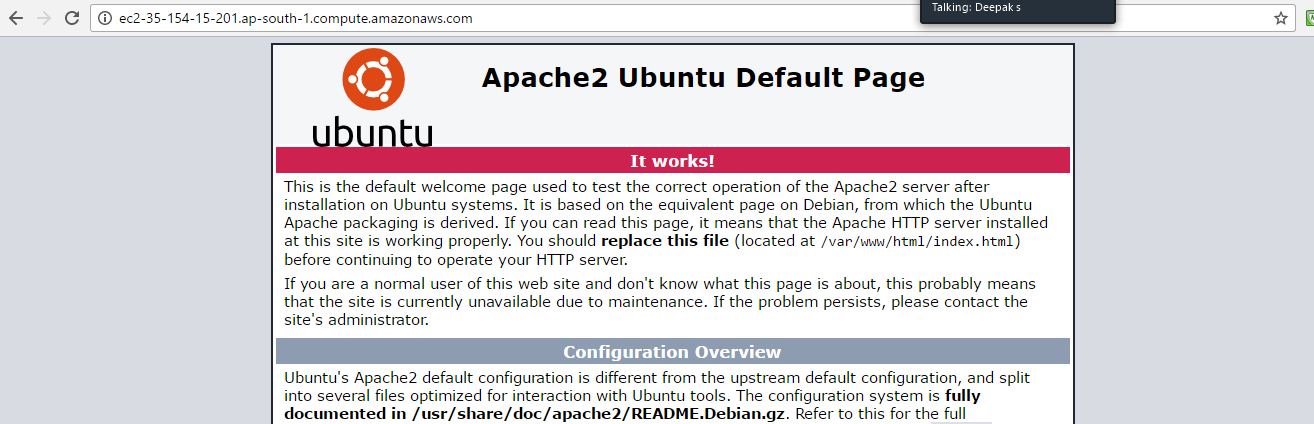
sudo /etc/init.d/apache2 restart //// Server May need a restart didn't get time to experiment without restart.

php -r 'echo "\n\nYour PHP installation is working fine.\n\n\n";'

///// above command is for checking PHP installation.

//// Following command is For Checking Apache2 //// ie. Give Public DNS in Browser Apache Test page should load.

<http://ec2-35-154-47-233.ap-south-1.compute.amazonaws.com>

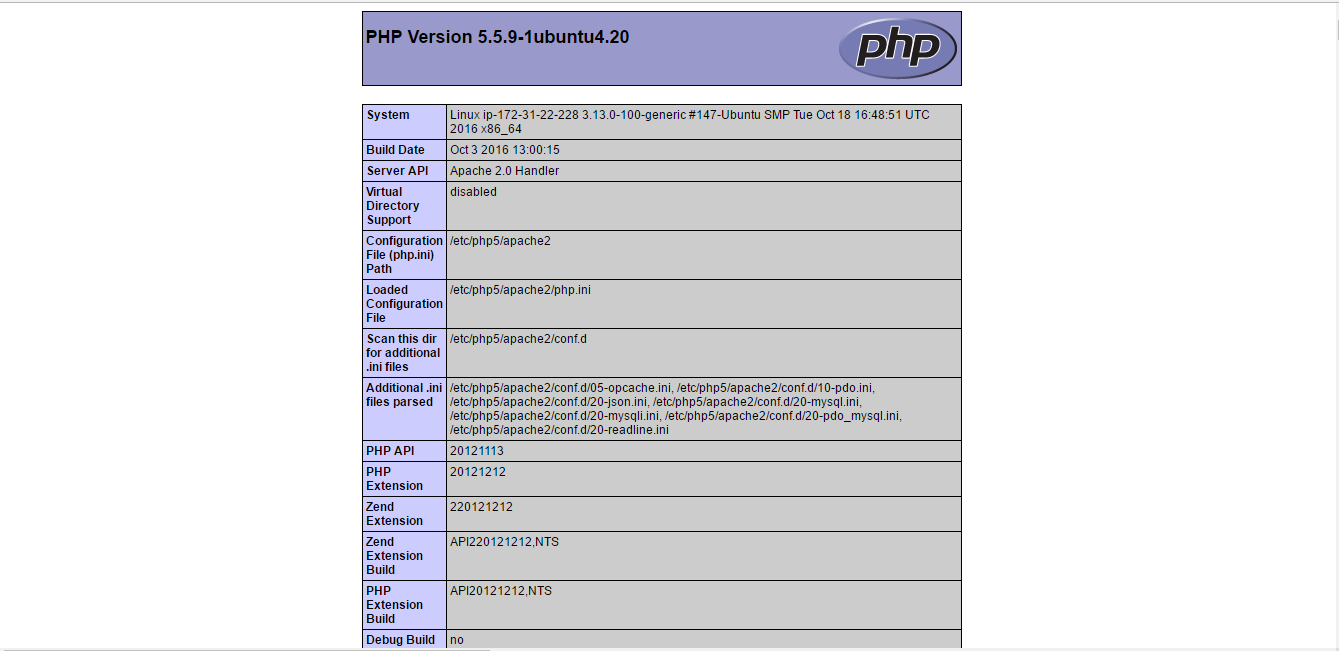


///// Following command is For writing PHP Configuration details to a file (phpinfo.php)

echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php

///// Following command is For checking PHP installation. //// Give Public DNS of EC2 and phpinfo.php after that with slash '/'

<http://ec2-35-154-47-233.ap-south-1.compute.amazonaws.com/phpinfo.php>



**To set file permissions for the Apache web server:**

//////Add the www group to your EC2 instance with the following command:

sudo groupadd www

sudo usermod -a -G www ubuntu /// Add the user "ubuntu" to the www group:

exit ///To refresh your permissions and include the new www group, log out from EC2.

ssh ubuntu@ec2-35-154-47-233.ap-south-1.compute.amazonaws.com ///Login again to EC2 with your Public DNS

//////Log back in again and verify that the www group exists with the groups command:-

. groups //// It will Display the following Information. Note that there will be 'www' group at the End.

//// Following is not command just I have copy pasted the displayed information.

ubuntu adm dialout cdrom floppy sudo audio dip video plugdev netdev www //////<<< Observe that www at the end

/////Change the group ownership of the /var/www directory and its contents to the www group:

. sudo chown -R root:www /var/www

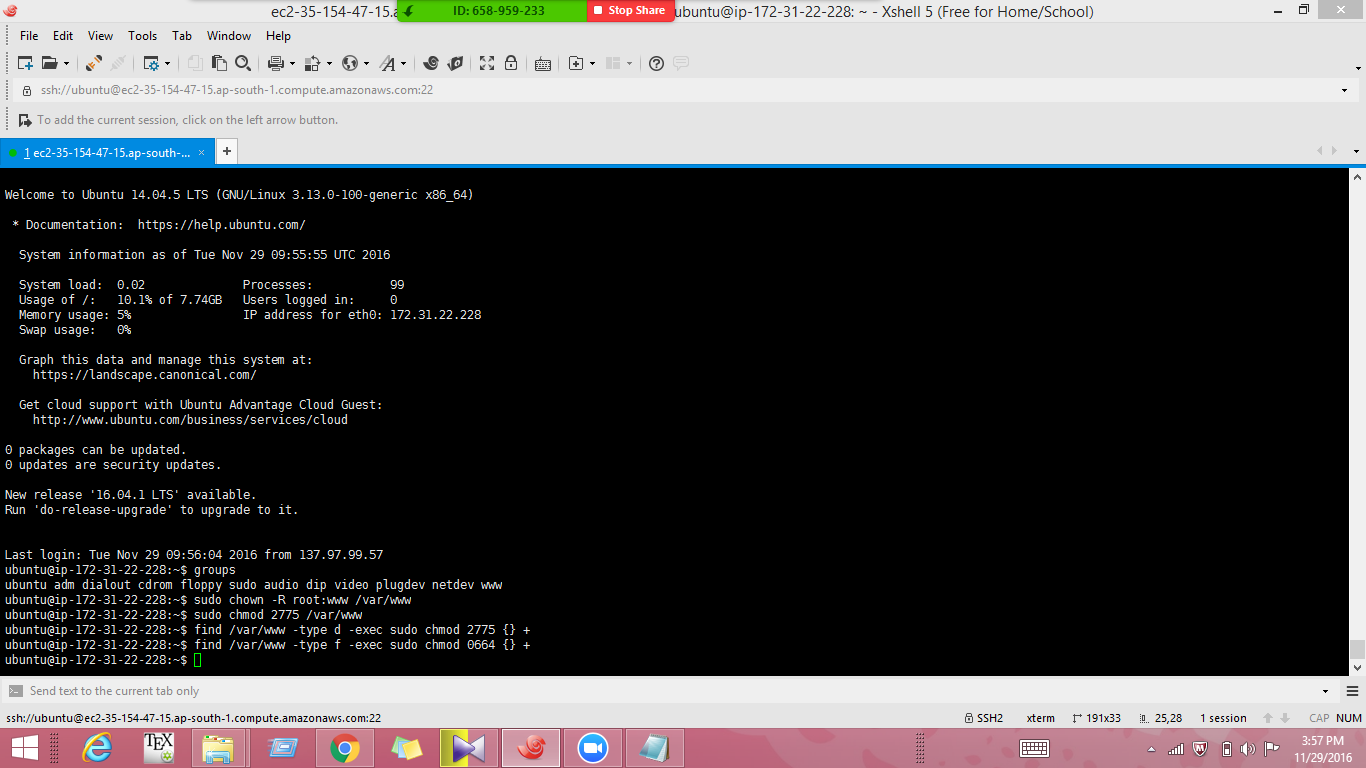
//////Change the directory permissions of /var/www and its subdirectories to add group write permissions and set the group ID on subdirectories created in the future:

sudo chmod 2775 /var/www

find /var/www -type d -exec sudo chmod 2775 {} +

////Recursively change the permissions for files in the /var/www directory and its subdirectories to add group write permissions:

find /var/www -type f -exec sudo chmod 0664 {}



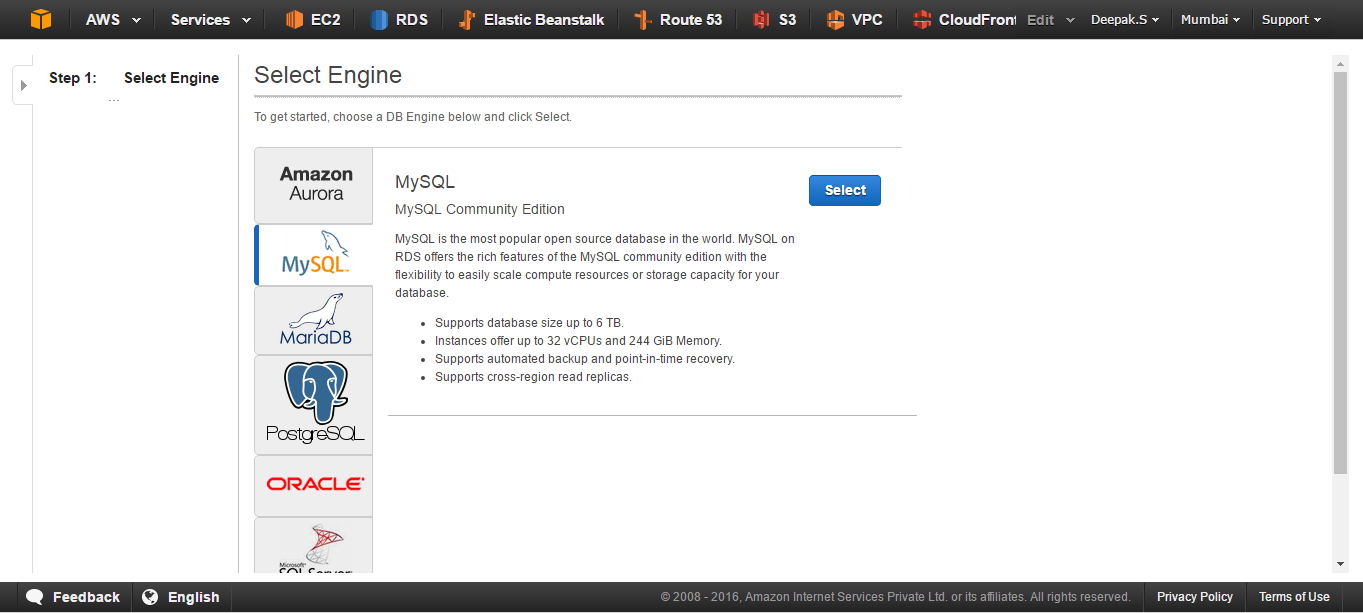
Launch RDS-Mysql Instance

1.Click on Services go to RDS

2. RDS Dashboard. Click on DB Instances

3. Launch DB Instance

4. Select MySql Tab. Click on Select.



5. Select Purpose of DB Instance. ( Slect Dev/Test MySql ( Other Options are chargeble))

Click on Next Step.

6. Specify DB Details Page:-

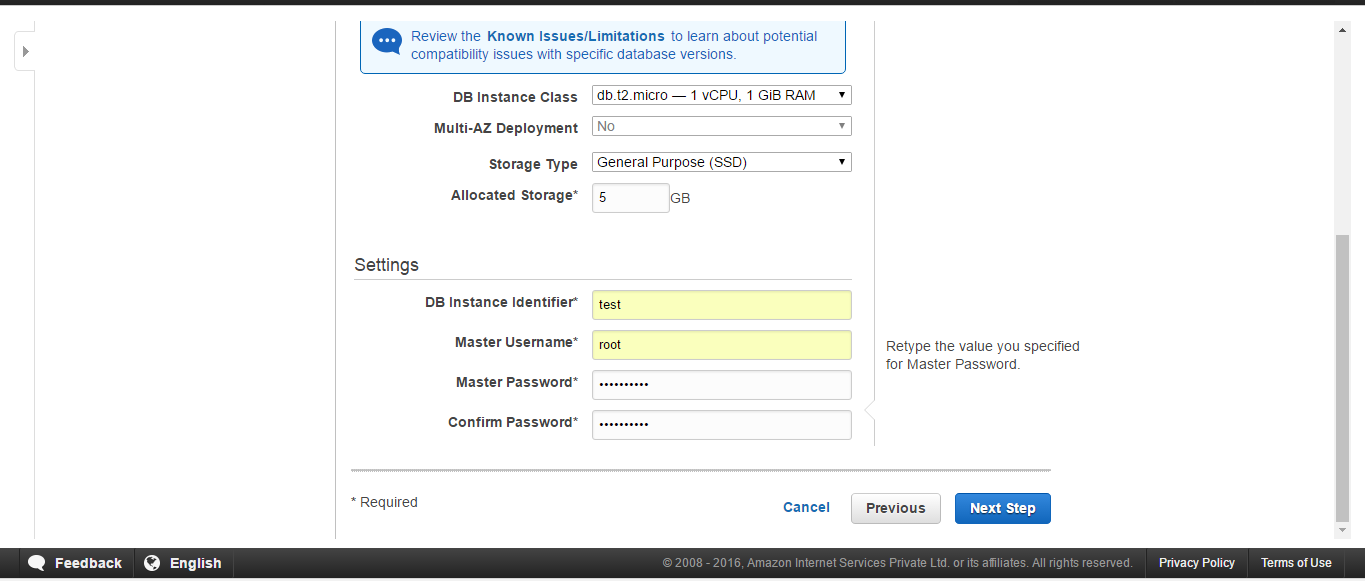
NB. Select the Option " Only show options that are eligible for RDS Free Tier "

7. DB Instance Class db.t2.micro--1vCPU. 1 GB RAM also select multi Availability Zone Deployment if needed (Chargable)DB instances using Multi-AZ deployments may have increased write and commit latency compared to a Single-AZ deployment,alsoIn the event of a planned or unplanned outage of your DB instance, Amazon RDS automatically switches to a standby replica in another Availability Zone if you have enabled Multi-AZ.

8. Storage Type -- General Purpose ( SSD )

9. NB>> Dont Change Allocated Storage 5 GB ( If you Increase it will be chargeable. )

10. Give name for DB Instance Identifier



11. Give " root " as Master Username

12. Password ( your Password ) ( welcome123 ). //// Copy your Password to a notepad. It is needed for PHP Connection

Click on Next Step

13. Configure Advanced Setting Page:-

Select Same VPC as given for Linux EC2 Instance.

Default Subnet Group will be selected automatically.

14. Select " Yes " for Publicly Accessible.

If you are selecting " No " in the Security Group you have to give required IP Address.

15. Select any Availability Zone.

16. For RDS you need to Create one Security Group

Select " Create new Security Group "

17. Database Options:-

Give a name in the Database Name field. //// //// Copy your Databse Name to a notepad. It is needed for PHP Connection

18. Leave the Database Port intact ( 3306 for MySql ; 1433 for MSSQL Server)

Leave Remaining Options in Database Options Default.DB Parameter Group, Option Group, etc.

19. Backup :-

///// Very NB : You can select 0 days to avoid charges.

Backup Retention Period 7 Days

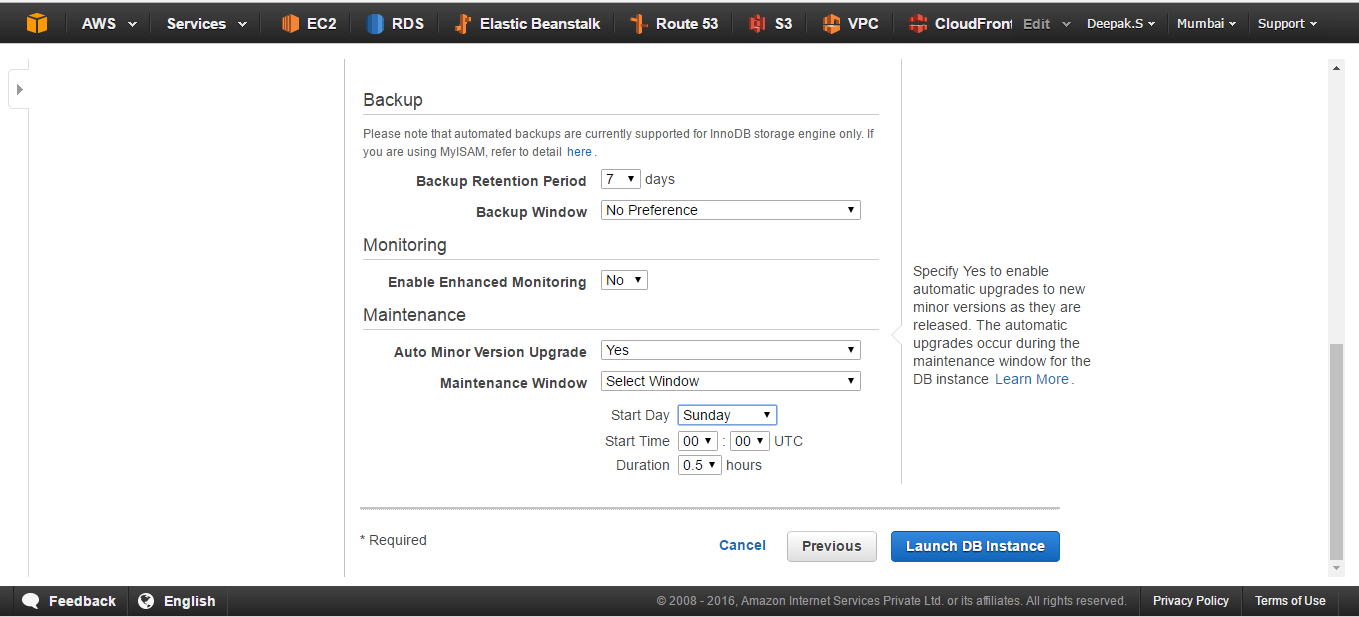
20. Backup Window ( Which Time you want to take Backup )

" No Preference " ( If selecting Give for Half an Hour )

21. Cloudwatch Monitoring

" No " for Enable Enhanced Monitoring ( If given 'Yes' It will be chargeable. )

22. Maintenance:-" Yes " for Auto Minor Version Upgrade ( No Charges )



23. Maintenance Window ( If selecting Give for Half an Hour )

Click on Launch DB Instance.

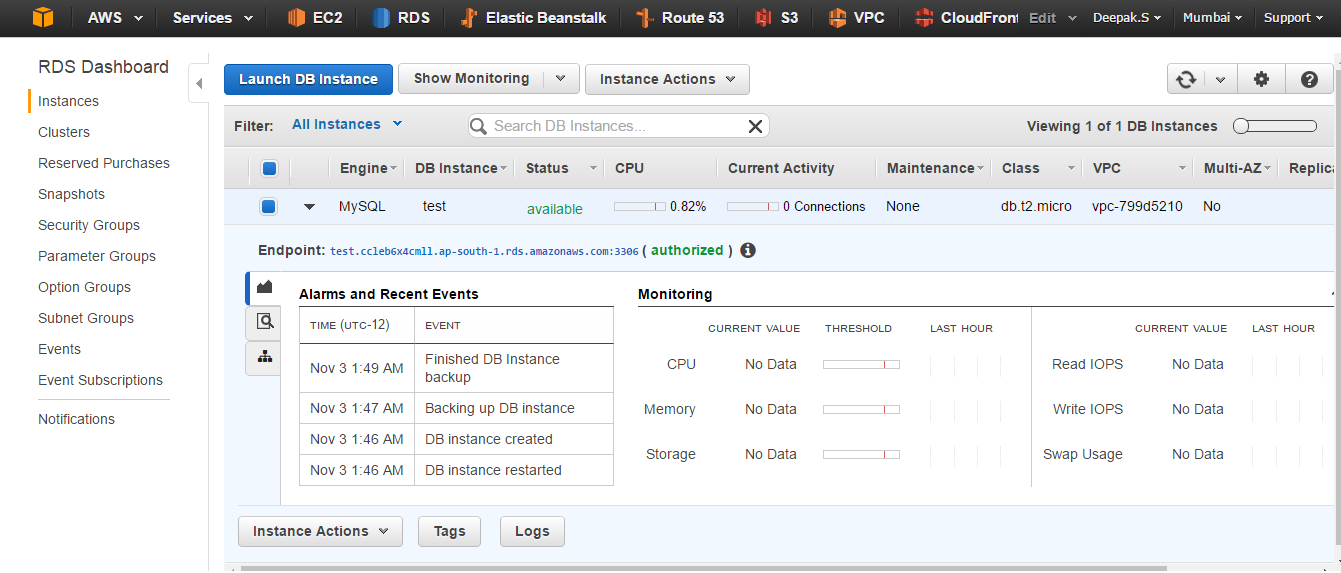
24. Click on view your DB Instances.

25. Go to RDS Dashboard and Check whether DB Instance is availbale or not.

RDS will take an initial backup. It will be deleted when you delete DB Instance

Click on Snapshot Tab on the RDS Dashboard to see the Snapshot of DB Instance.

26. Copy the Endpoint URL of DB Instance and Paste in Notepad and Remove the ':' and Port no Click on Security Group of RDS DB Instance ( rds-launch-wizard-1 ) You will go to Security Groups window of EC2 Dashboard, Select the Security Group and Click on Inbound.Add Inbound Rule Type " MySql/Aurora " and Select source as 'anywhere' ie (0.0.0.0/.0)



**Connect your Apache web server to your RDS DB instance**

////In EC2 instance, change the directory to /var/www and create a new subdirectory named 'inc'

. cd /var/www

. mkdir inc //// create a new subdirectory

. cd inc /////change to a new subdirectory

///// Create a new file in the inc directory named dbinfo.inc, and then edit the file by calling nano (or the editor of your choice).

>dbinfo.inc /////create a file named as dbinfo and with extention .inc

//// Make sure that you are in " /var/www/inc/ " directory while creating this file.

///// Command prompt will show somthing like this " /var/www/inc ~]$ "

vi dbinfo.inc //// Editing dbinfo.inc with vi editor for adding Public DNS of Server, User, Password

//////Press 'Insert' Key for Adding lines to dbinfo.inc

//// Add the following contents to the dbinfo.inc file, where endpoint is the endpoint of your RDS MySQL DB instance, without the port, and master password is the master password for your RDS MySQL DB instance.

///// Copy the Part between Lines '---' and paste to dbinfo.inc.

//// Change the Password and Database name ( given while you create RDS ) also.

<?php

define('DB\_SERVER', 'techart.cpqsf00brjcy.ap-south-1.rds.amazonaws.com');

define('DB\_USERNAME', 'root');

define('DB\_PASSWORD', 'welcome123');

define('DB\_DATABASE', 'testlab');

?>

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

///// Save and close the dbinfo.inc file.

Press 'Esc' Key Then ':' Then 'wq' /////for Save and Quit

/////Change the directory to /var/www/html //// Manadatory. Otherwise it will not work.

. cd /var/www/html////// Create a new file in the html directory named SamplePage.php

30. >SamplePage.php

31. vi SamplePage.php

//////Press 'Insert' Key for Adding lines to SamplePage.php

///// Copy All Lines between the lines below

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<?php include "../inc/dbinfo.inc"; ?>

<html>

<body>

<h1>THANK YOU </h1>

<?php

/\* Connect to MySQL and select the database. \*/

$connection = mysqli\_connect(DB\_SERVER, DB\_USERNAME, DB\_PASSWORD);

if (mysqli\_connect\_errno()) echo "Failed to connect to MySQL: " . mysqli\_connect\_error();

$database = mysqli\_select\_db($connection, DB\_DATABASE);

/\* Ensure that the Employees table exists. \*/

VerifyEmployeesTable($connection, DB\_DATABASE);

/\* If input fields are populated, add a row to the Employees table. \*/

$employee\_name = htmlentities($\_POST['Name']);

$employee\_address = htmlentities($\_POST['Address']);

if (strlen($employee\_name) || strlen($employee\_address)) {

AddEmployee($connection, $employee\_name, $employee\_address);

}

?>

<!-- Input form -->

<form action="<?PHP echo $\_SERVER['SCRIPT\_NAME'] ?>" method="POST">

<table border="0">

<tr>

<td>Name</td>

<td>Address</td>

</tr>

<tr>

<td>

<input type="text" name="Name" maxlength="45" size="30" />

</td>

<td>

<input type="text" name="Address" maxlength="90" size="60" />

</td>

<td>

<input type="submit" value="Add Data" />

</td>

</tr>

</table>

</form>

<!-- Display table data. -->

<table border="1" cellpadding="2" cellspacing="2">

<tr>

<td>ID</td>

<td>Name</td>

<td>Address</td>

</tr>

<?php

$result = mysqli\_query($connection, "SELECT \* FROM Employees");

while($query\_data = mysqli\_fetch\_row($result)) {

echo "<tr>";

echo "<td>",$query\_data[0], "</td>",

"<td>",$query\_data[1], "</td>",

"<td>",$query\_data[2], "</td>";

echo "</tr>";

}

?>

</table>

<!-- Clean up. -->

<?php

mysqli\_free\_result($result);

mysqli\_close($connection);

?>

</body>

</html>

<?php

/\* Add an employee to the table. \*/

function AddEmployee($connection, $name, $address) {

$n = mysqli\_real\_escape\_string($connection, $name);

$a = mysqli\_real\_escape\_string($connection, $address);

$query = "INSERT INTO `Employees` (`Name`, `Address`) VALUES ('$n', '$a');";

if(!mysqli\_query($connection, $query)) echo("<p>Error adding employee data.</p>");

}

/\* Check whether the table exists and, if not, create it. \*/

function VerifyEmployeesTable($connection, $dbName) {

if(!TableExists("Employees", $connection, $dbName))

{

$query = "CREATE TABLE `Employees` (

`ID` int(11) NOT NULL AUTO\_INCREMENT,

`Name` varchar(45) DEFAULT NULL,

`Address` varchar(90) DEFAULT NULL,

PRIMARY KEY (`ID`),

UNIQUE KEY `ID\_UNIQUE` (`ID`)

) ENGINE=InnoDB AUTO\_INCREMENT=1 DEFAULT CHARSET=latin1";

if(!mysqli\_query($connection, $query)) echo("<p>Error creating table.</p>");

}

}

/\* Check for the existence of a table. \*/

function TableExists($tableName, $connection, $dbName) {

$t = mysqli\_real\_escape\_string($connection, $tableName);

$d = mysqli\_real\_escape\_string($connection, $dbName);

$checktable = mysqli\_query($connection,

"SELECT TABLE\_NAME FROM information\_schema.TABLES WHERE TABLE\_NAME = '$t' AND TABLE\_SCHEMA = '$d'");

if(mysqli\_num\_rows($checktable) > 0) return true;

return false;

}

?>

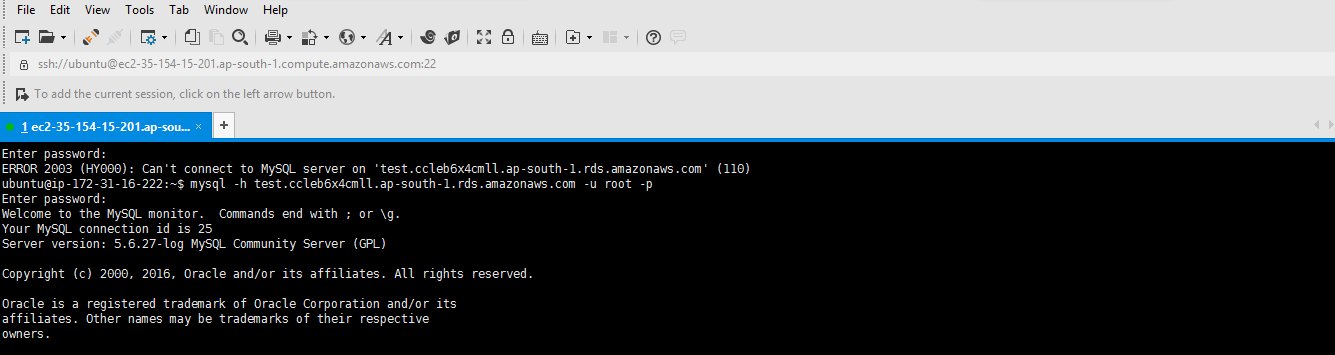
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

//////Press 'Esc'+':'+'wq' ///Save and close the SamplePage.php file. ie. Press 'Esc' Key Then ':' Then 'wq'

///////Check Mysql DB by connecting to it. Open another tab in Xshell and do following.

///// Connect to your ( MySQL ) RDS and check that you are able access your database. by giving Public DNS of RDS.

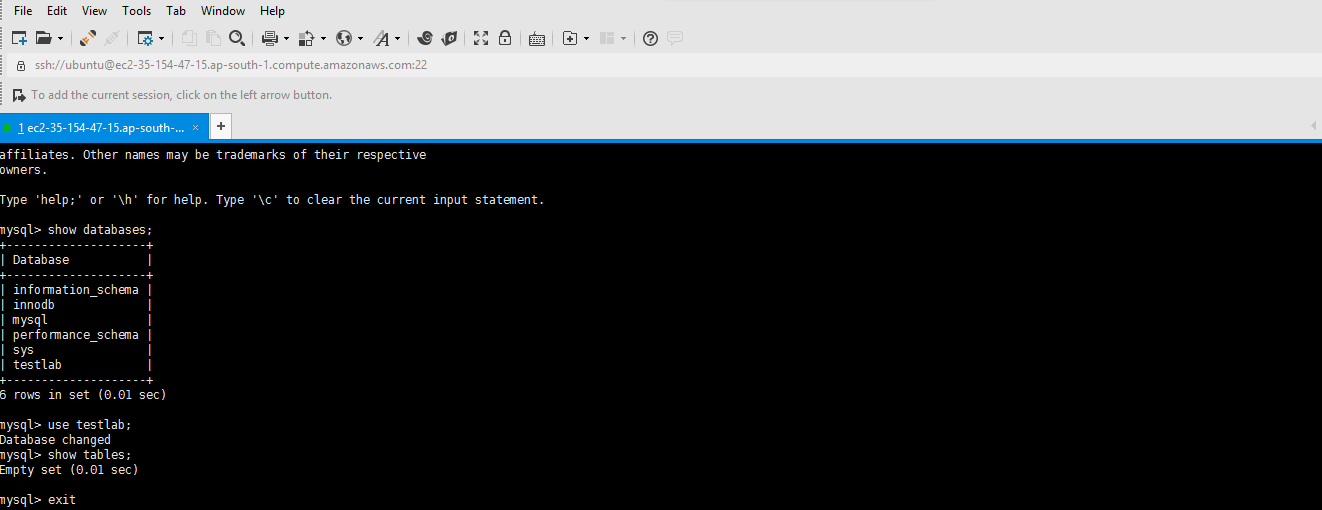
//////mysql -h techart.cpqsf00brjcy.ap-south-1.rds.amazonaws.com -u root –p and give the password



. show databases;

use <your databasename>;

show tables; //// It will show empty set. It is fine.



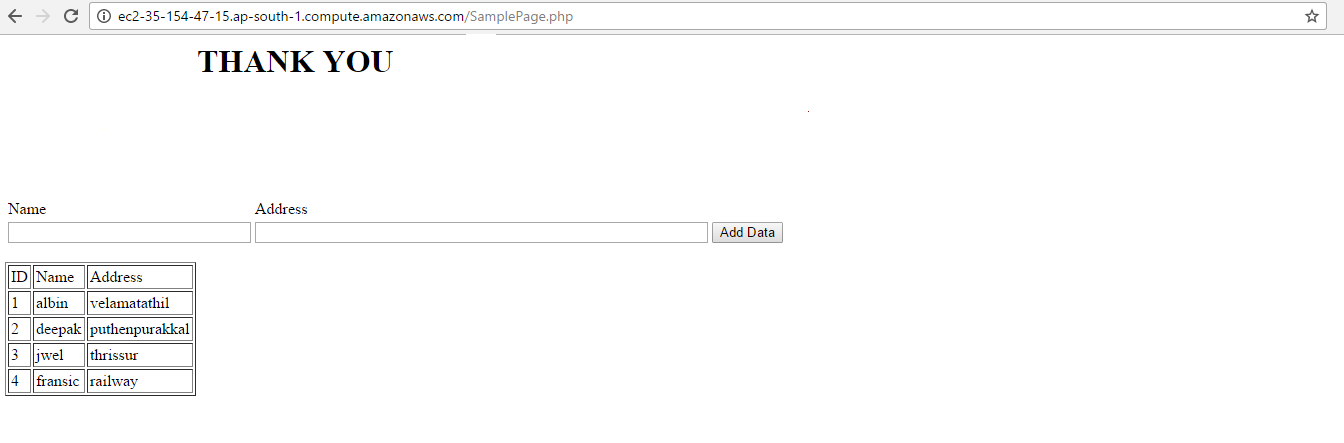
///// Come back to EC2 Instance Tab in Xshell and Restart Apache

sudo /etc/init.d/apache2 restart

/////Verify that your web server successfully connects to your RDS MySQL DB instance

/////by opening a web browser and browsing to http://EC2 instance endpoint/SamplePage.php,

/////for example: http://ec2-55-122-41-31.us-west-2.compute.amazonaws.com/SamplePage.php.



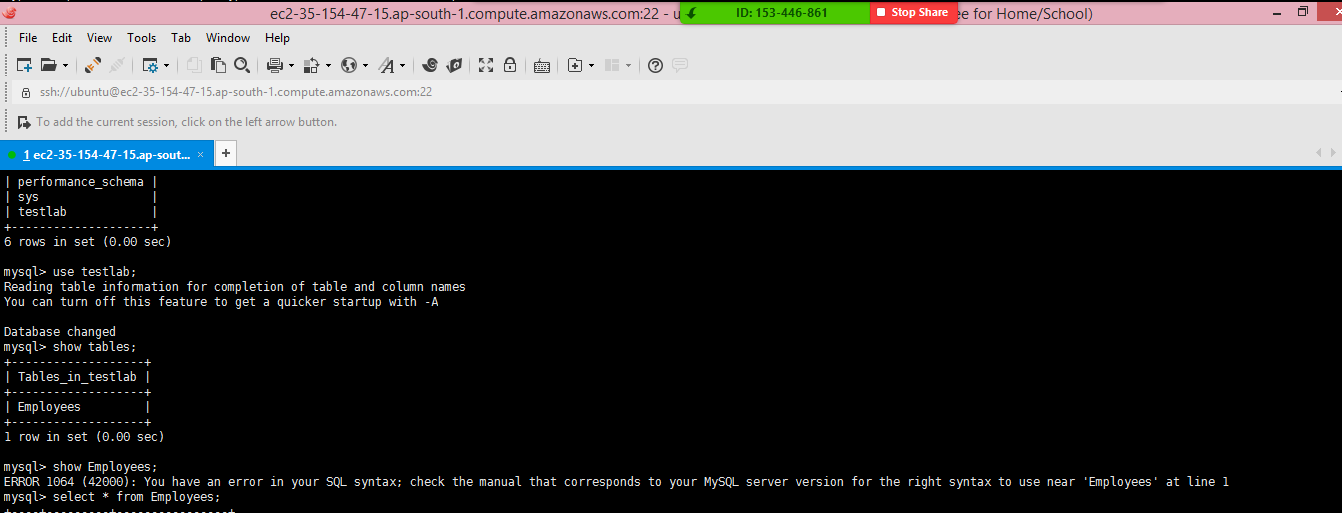
/////////You can use SamplePage.php to add data to your RDS MySQL DB instance. The data that you add is then displayed on the page.

////// Enter as much data as you want into both text boxes and click on Submit Button.

////// Each time the enter data it will get displayed on the webpage.

///// ENJOY. !!!!!!!! YOU ARE CONNECTED TO MYSQL FROM PHP !!!!!!!!!!

The same data can be viewed in the testlab by loging in to the MySQL from the Xshell as below



**RESULT**

1. EC2 instance running as Front end Application Server and connecting to an Amazon RDS service, running as back end database was installed and configured

2. The applicationwas successfully installed on the App Server and also communicated with RDS

3. Features of RDS such as Multi AZ deployment, automated backups are shown how to use in this project.

=====================================================================================================

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------