

Hands on with Cloud Computing

9th & 10th -October-2020

Created by:
Ankit Velani, Consultant
Deloitte India, Bangalore

Submitted at:
Department Of MCA
Siddaganga Institute of Technology, Tumkuru



Overview

This document contains steps to set up various cloud services on the Amazon Web Service (AWS). Documents divides in various section, Create Amazon account, Setup EC2 on Amazon, Accessing Amazon EC2 machine using FileZilla and SSH, Installing Softwares (PHP, MySQL, R) on EC2 and last Working with Amazon RDS.

Prerequisites

- Audience should have clear understanding on Computer Networking.
- Audience should know installing and uninstalling software in Computer.
- Audience should know any programming and DBMS.

Create Amazon Account

- Open an amazon portal : <https://aws.amazon.com/free/>

AWS Free Tier

Gain free, hands-on experience with the AWS platform, products, and services.

[Create a Free Account](#)

Types of offers

Explore more than 60 products and start building on AWS using the free tier. Three different types of free offers are available depending on the product used. See below for details on each product.



Always free



12 months free



Trials

These free tier offers do not expire and are available to all AWS customers

Enjoy these offers for 12-months following your initial sign-up date to AWS

Short-term free trial offers are available through many different software solutions

- Click on *Create a Free account*

Create an AWS account

AWS Accounts Include 12 Months of Free Tier Access

Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB
Visit aws.amazon.com/free for full offer terms

Email address

Password

Confirm password

AWS account name

[Continue](#)

[Sign in to an existing AWS account](#)

© 2016 Amazon Web Services, Inc. or its affiliates.
All rights reserved.
[Privacy Policy](#) [Terms of Use](#)

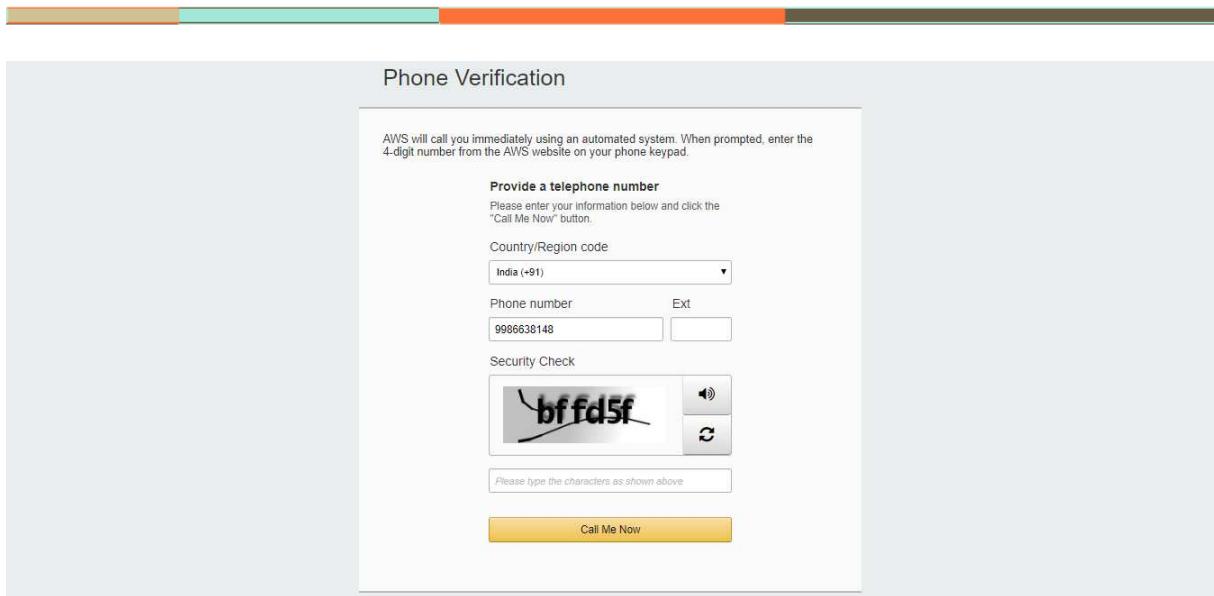
- Enter **email** address and **password** for AWS account creation.
- Enter your personal details to complete the AWS account.

The screenshot shows the 'Create Account' page. At the top, it says 'All fields are required'. Below that, instructions say 'Please select the account type and complete the fields below with your contact details.' The 'Account type' section has two options: 'Professional' (radio button) and 'Personal' (radio button). The 'Full name' field contains 'Ankit Velani'. The 'Phone number' field contains '0988638148'. The 'Country/Region' dropdown is set to 'India'. A note below it states: '* If you selected India, your country/region selection cannot be changed after creating the account.' The 'Address' field contains 'Bangalore'. Below it is a smaller field for 'Apartment, suite, unit, building, floor, etc.'. The 'City' field contains 'Bangalore'. The 'State / Province or region' field contains 'Karnataka'. The 'Postal code' field contains '560037'. At the bottom of the form, there is an 'Amazon Internet Services Pvt. Ltd. Customer Agreement' section with a note about AWS Free Tier Limits and a checkbox for accepting the terms. A yellow 'Create Account and Continue' button is at the bottom right.

5. Click on **Create Account and Continue**
6. Payment Details. Amazon will verify your credit/debit card with minimal amount which is Rs. 2 , will be refund in 3-5 business day.

The screenshot shows the 'Payment Information' page. At the top, it says 'Payment Information'. Below that, a note says 'Please type your payment information so we can verify your identity. We will not charge you unless your usage exceeds the [AWS Free Tier Limits](#). Review [frequently asked questions](#) for more information.' A callout box provides information about the secure submission process. The form includes fields for 'Credit/Debit card number', 'Expiration date' (set to 08/2019), 'Cardholder's name', 'Billing address' (set to 'Use my contact address' with 'Bangalore, Bangalore Karnataka 560037 IN'), and 'Do you have a PAN?'. It also includes 'Yes' and 'No' radio buttons. A yellow 'Secure Submit' button is at the bottom right.

7. Enter Create/Debit card details and click on **Secure Submit** for Payment.
8. Next Step, Amazon will verify the mobile number.



AWS will call you immediately using an automated system. When prompted, enter the 4-digit number from the AWS website on your phone keypad.

Provide a telephone number
Please enter your information below and click the "Call Me Now" button.

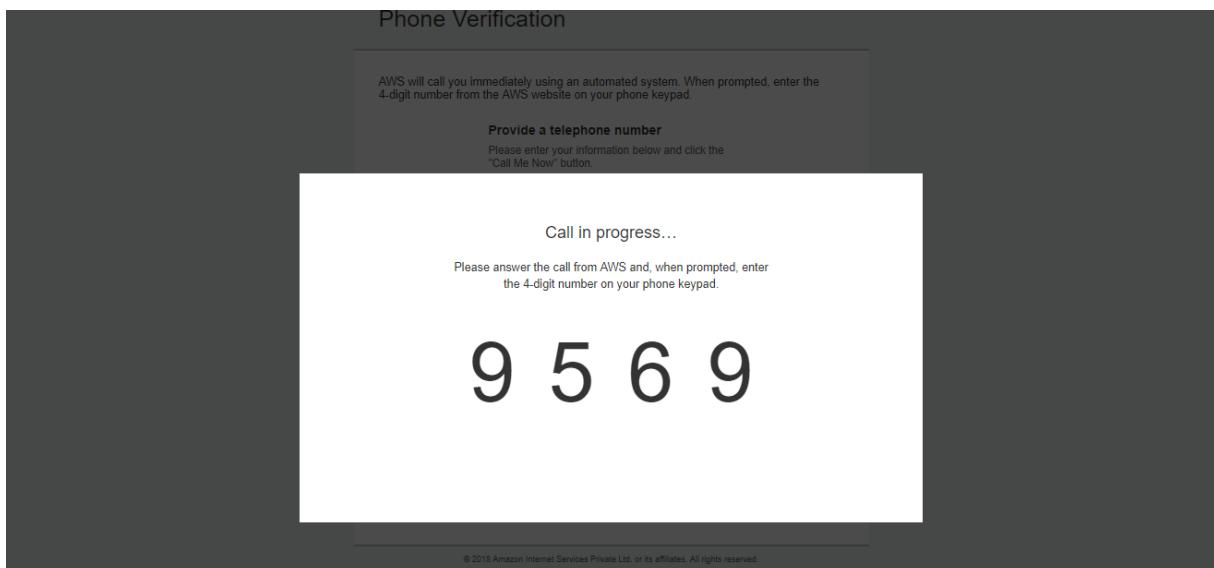
Country/Region code

Phone number Ext

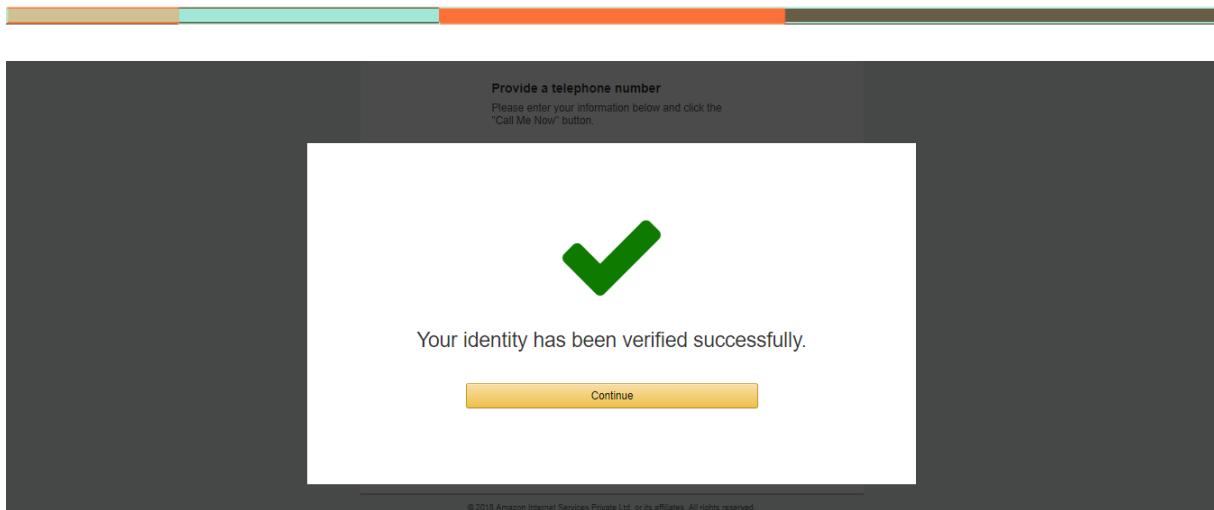
Security Check


Please type the characters as shown above

Call Me Now



Account verification may done either OTP/ by Call. Please verify your account after providing your mobile number.



9. Click on **Continue**,

10. Once verification finish, Amazon allow us to **choose the plan** for Cloud Services.

Select a Support Plan

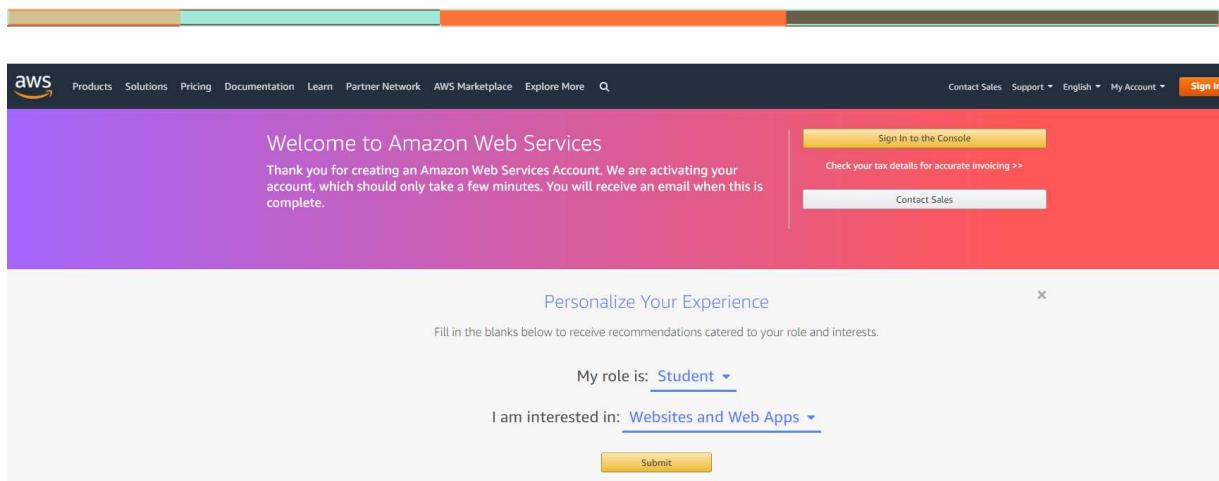
AWS offers a selection of support plans to meet your needs. Choose the support plan that best aligns with your AWS usage. [Learn more](#)

Basic Plan	Developer Plan	Business Plan
Free	From \$29/month	From \$100/month
<ul style="list-style-type: none"> Included with all accounts 24/7 self-service access to forums and resources Best practice checks to help improve security and performance Access to health status and notifications 	<ul style="list-style-type: none"> For early adoption, testing and development Email access to AWS Support during business hours 1 primary contact can open an unlimited number of support cases 12-hour response time for nonproduction systems 	<ul style="list-style-type: none"> For production workloads & business-critical dependencies 24/7 chat, phone, and email access to AWS Support Unlimited contacts can open an unlimited number of support cases 1-hour response time for production systems

11. Chose the Basic plan (**Free**).

12. Click on Continue.

13. Next it will ask for your role for providing personalize experience.



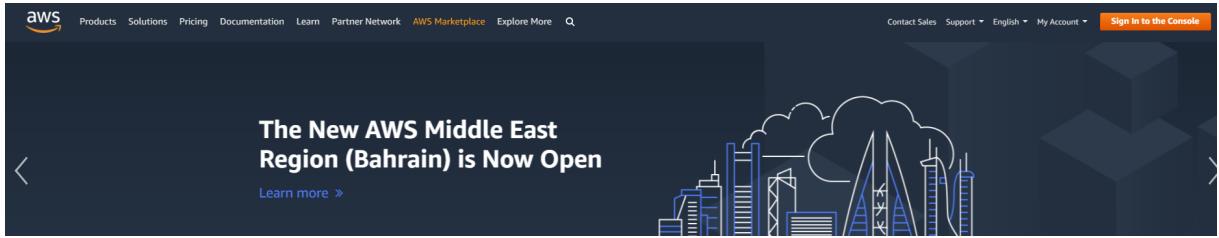
14. Click on Submit , after selecting your role.
15. Congratulation, you have successfully registered with Amazon Cloud Platform.

Setup EC-2 on Amazon Cloud.

What is Elastic Compute Cloud (EC2)?

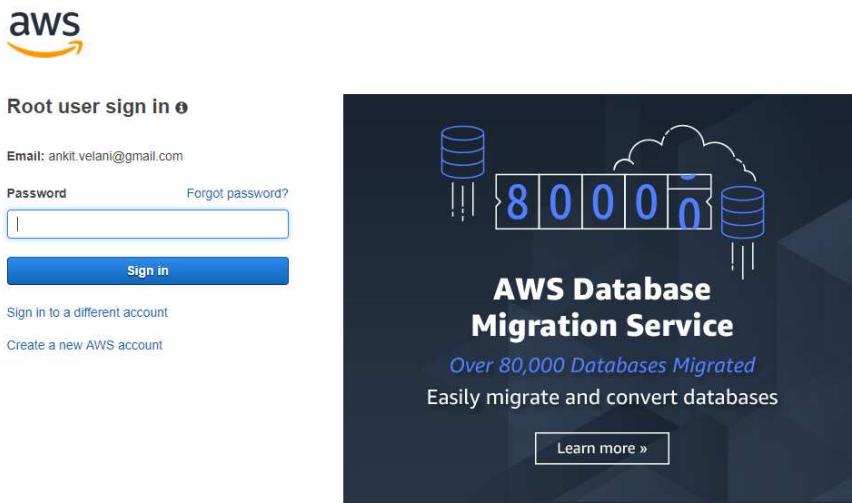
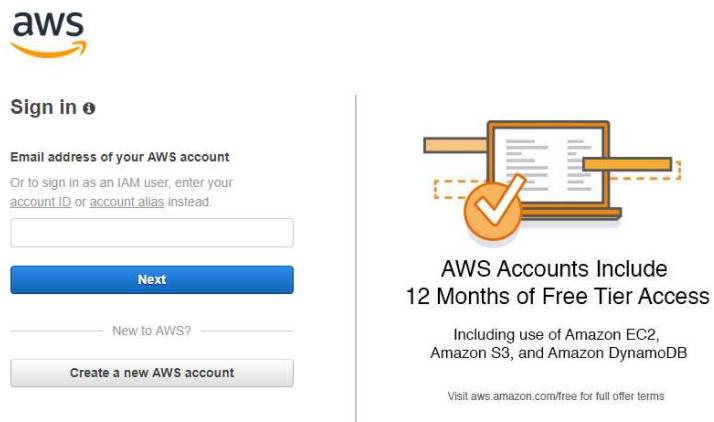
- Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
- Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.
- Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change.
- Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate them from common failure scenarios.

1. Login to Amazon cloud portal. <https://aws.amazon.com/>



Click on Sign in to the Console.

2. Enter your registered email and click on **Next**.



3. Enter your password and click on **Sign in**

4. After successfully login, Console landing page will appear.

5. Click on All Services ,

6. Choose EC2 , from category “ Compute “

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services

All services

- Compute**
 - EC2
 - EC2 Container Service
 - Lightsail
 - Elastic Beanstalk
 - Lambda
 - Batch
- Storage**
 - S3
 - EFS
 - Glacier
 - Storage Gateway
- Database**
 - RDS
- Developer Tools**
 - CodeStar
 - CodeCommit
 - CodeBuild
 - CodeDeploy
 - CodePipeline
 - X-Ray
- Management Tools**
 - CloudWatch
 - CloudFormation
 - CloudTrail
 - Config
 - OpsWorks
 - Service Catalog
 - Trusted Advisor
- Internet of Things**
 - AWS IoT
 - AWS Greengrass
- Contact Center**
 - Amazon Connect
- Game Development**
 - Amazon GameLift
- Mobile Services**
 - Mobile Hub
 - Cognito
 - Device Farm
 - Mobile Analytics
 - Pinpoint

Helpful tips

Manage your costs

Create an organization

Explore AWS

New Product Announcements

Migrate from Oracle to Amazon Aurora

7. Let's jumped to EC2 Dashboard. Click on Launch Instance

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Resources

You are using the following Amazon EC2 resources in the US East (Ohio) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
0 Key Pairs	1 Security Groups
0 Placement Groups	

Learn more about the latest in AWS Compute from AWS re:Invent 2017 by viewing the EC2 Videos.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your instances will launch in the US East (Ohio) region

Service Health

Service Status: US East (Ohio): No events

Scheduled Events

Account Attributes

Supported Platforms

VPC

Default VPC

vpc-f9e9da91

Resource ID length management

Console experiments

Additional Information

Getting Started Guide

Documentation

All EC2 Resources

Forums

Pricing

Contact Us

AWS Marketplace

Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard.

Or try these popular AMIs:

8. Choose Amazon Machine Image (AMI), for learning choose **Free tier eligible**.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

- My AMIs
- AWS Marketplace
- Community AMIs
- Free tier only ⓘ

AMI Name	Description	Root device type	Virtualization type	Select	Architecture
Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0cf31d971a3ca20d6	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.	ebs	hvm	Select	64-bit
Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0b59bfae6be064b78	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	ebs	hvm	Select	64-bit
Red Hat Enterprise Linux 7.5 (HVM), SSD Volume Type - ami-03291866	Red Hat Enterprise Linux version 7.5 (HVM), EBS General Purpose (SSD) Volume Type			Select	64-bit

9. Select AMI type Ubuntu **Server 18.04 LTS (HVM)**, click on Select.

10. Next step, choose instance type. Select the **t2.mirco (Free tier eligible)**

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by:	All instance types	Current generation	Show/Hide Columns					
Currently selected: t2.micro (Variable ECUs, 1 vCPU, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)								
	Family	Type	vCPUs ⓘ	Memory (GiB) ⓘ	Instance Storage (GB) ⓘ	EBS-Optimized Available ⓘ	Network Performance ⓘ	IPv6 Support ⓘ
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	FRS only	-	Low to Moderate	Yes

T2.mirco instance configuration.

Variable ECUs,

1 vCPUs

2.5 GHZ , Intel Xeon Family

1 GiB memory RAM

Check the free services

<https://aws.amazon.com/free/#legal>

11. Click on Review and Launch.

Hands on Cloud Computing Workshop.

Dept. Of MCA – SIT, Tumkuru

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-7d132e18
 Free tier eligible
 Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
 Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name	launch-wizard-1
Description	launch-wizard-1 created 2018-08-20T13:28:10.802+05:30

Buttons: Cancel | Previous | **Launch**

12. Click on Launch.

13. Next Screen, It will ask to create Key for accessing EC-2 Machine.

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

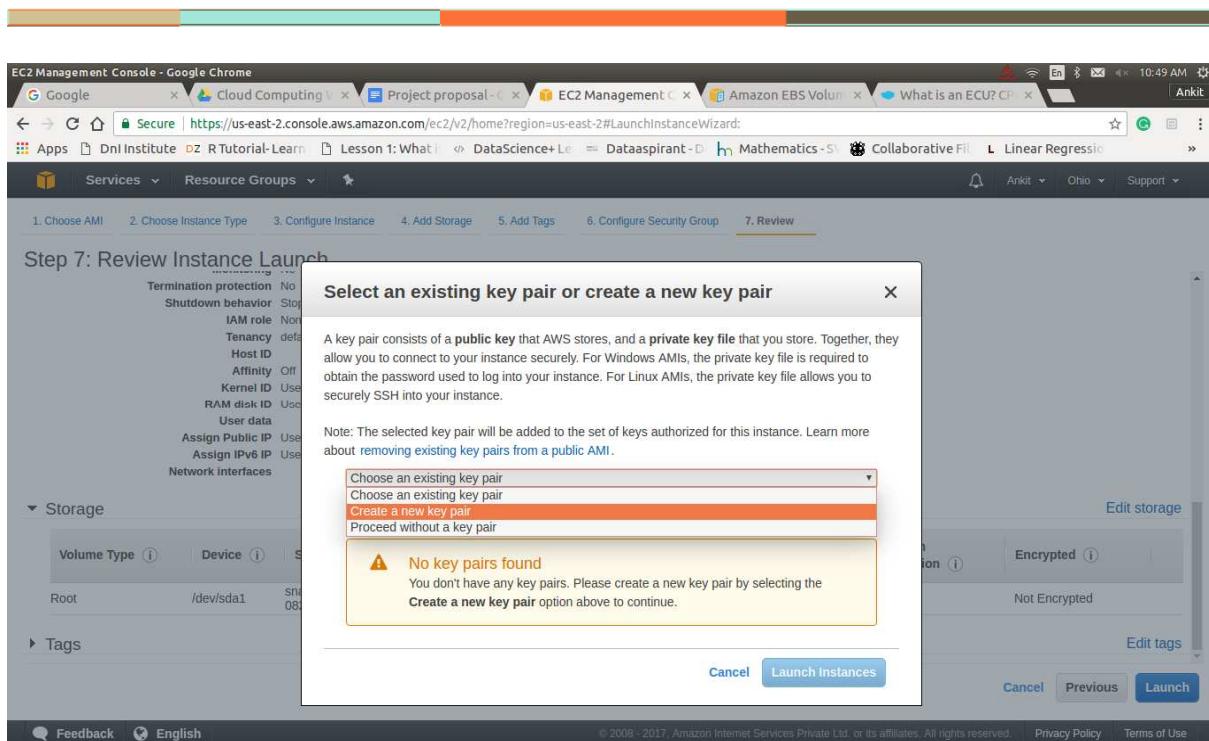
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair
Select a key pair
No key pairs found

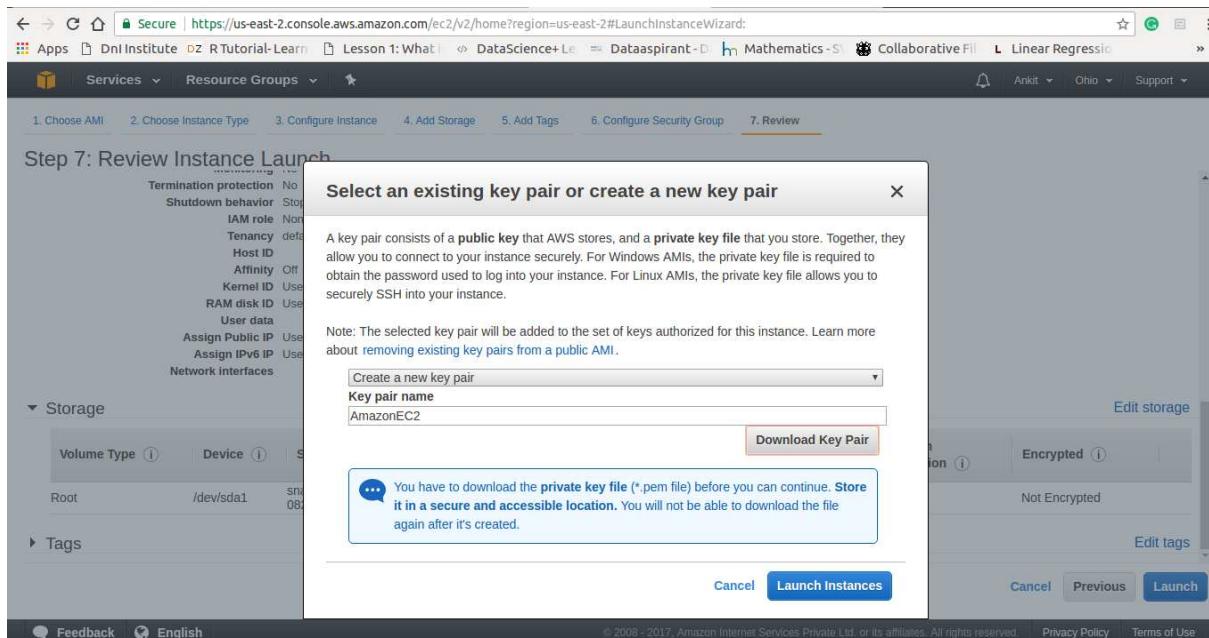
No key pairs found
 You don't have any key pairs. Please create a new key pair by selecting the [Create a new key pair](#) option above to continue.

Buttons: Cancel | **Launch instances**

14. Select the option to create a new key pair from select box.



15. Write KeyName and Download Key Pair



16. After Download, Click on Launch Instances.

The screenshot shows the AWS Launch Status page. At the top, there's a green banner with the message "Your instances are now launching" and a link to "View launch log". Below it, a blue banner says "Get notified of estimated charges" with a note about creating billing alerts. A section titled "How to connect to your instances" provides instructions and links to the User Guide and Discussion Forum. Another section, "While your instances are launching you can also", includes links for creating status check alarms and EBS volumes. At the bottom right is a "View Instances" button.

The screenshot shows the AWS EC2 Dashboard under the "Instances" tab. It displays a table of instances with one row selected. The selected instance is "i-0ad2d2f2e17281327" of type "t2.micro" in "us-east-2c" availability zone, running, with a public DNS of "ec2-52-14-56-150.us-east-2.compute.amazonaws.com". Below the table, a detailed view for the selected instance shows its configuration: Instance ID (i-0ad2d2f2e17281327), Instance state (running), Instance type (t2.micro), Elastic IP (None), Availability zone (us-east-2c), Public DNS (IPv4) (ec2-52-14-56-150.us-east-2.compute.amazonaws.com, IPv4 Public IP: 52.14.56.150), and Private DNS (ip-172-31-32-230.us-east-2.compute.internal, Private IP: 172.31.32.230).

19. Successfully launched t2.micro instance. We have description of an instance to access. For the password we must use Downloaded Key files. There is no option to create user defined password.

20. Description of the instance.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store (Volumes, Snapshots), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs), and Compute Optimizer. The main content area has tabs for Launch Instance, Connect, and Actions. Below that is a search bar and a table header with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv4. A single instance row is shown: i-0ad2d2f2e17281327, t2.micro, us-east-2c, running, 2/2 checks ... (green), None, ec2-52-14-56-150.us-east-2.compute.amazonaws.com, 52.14.56.150. Below the table, there are tabs for Description, Status Checks, Monitoring, and Tags. The Description tab is selected, displaying detailed information about the instance, including its ID, state, type, and various network and security details.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-0ad2d2f2e17281327	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-52-14-56-150.us-east-2.compute.amazonaws.com	52.14.56.150

Description

Instance ID	i-0ad2d2f2e17281327	Public DNS (IPv4)	ec2-52-14-56-150.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.56.150
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-32-230.us-east-2.compute.internal
Availability zone	us-east-2c	Private IPs	172.31.32.230
Security groups	launch-wizard-1, view inbound rules	Secondary private IPs	
Scheduled events	No scheduled events	VPC ID	vpc-bd2d6dd4
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170619 (ami-019abc64)	Subnet ID	subnet-94bd1ad9
Platform	-	Network interfaces	eth0
IAM role	-	Sourced/dest. check	True
Key pair name	AmazonEC2		

Setup EC2 with Windows Instance

- Kindly refer the same steps which we followed for Ubuntu instance, select the Windows as instance type in **step -9**, rest steps will be the same as Ubuntu.
- Choose micro instance for learning which free-tier is.

Step 1: Choose an Amazon Machine Image (AMI)

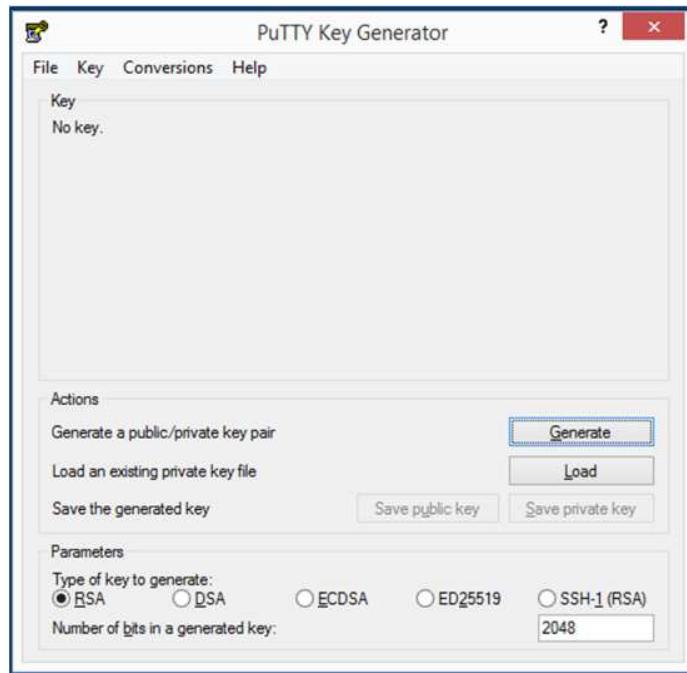
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

AMIs	Name	Description	Type	Select
My AMIs (0)				
AWS Marketplace (589)	Microsoft Windows Server 2019 Base - ami-0eaa025a752a23c5b	Microsoft Windows 2019 Datacenter edition. [English]	Root device type: ebs Virtualization type: hvm ENA Enabled: Yes	Select 64-bit (x86)
Community AMIs (6680)	Microsoft Windows Server 2019 Base with Containers - ami-0fa9b4b558e5c006d	Microsoft Windows 2019 Datacenter edition with Containers. [English]	Root device type: ebs Virtualization type: hvm ENA Enabled: Yes	Select 64-bit (x86)
Free tier only (1)	Microsoft Windows Server 1903 Base - ami-033dbfe1cb4415830	Microsoft Windows Server 1903 Semi-Annual Channel release [English]	Root device type: ebs Virtualization type: hvm ENA Enabled: Yes	Select 64-bit (x86)
	Microsoft Windows Server 2016 Base - ami-0e2c545d892567880			Select

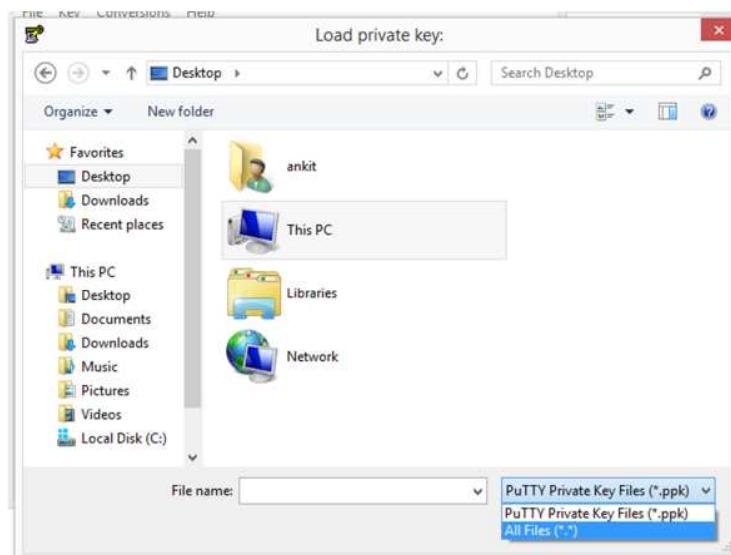
Converting Encrypted file to ppk (Private Key) to access EC2

Note: Make sure that Puttgen should be installed in your machine.

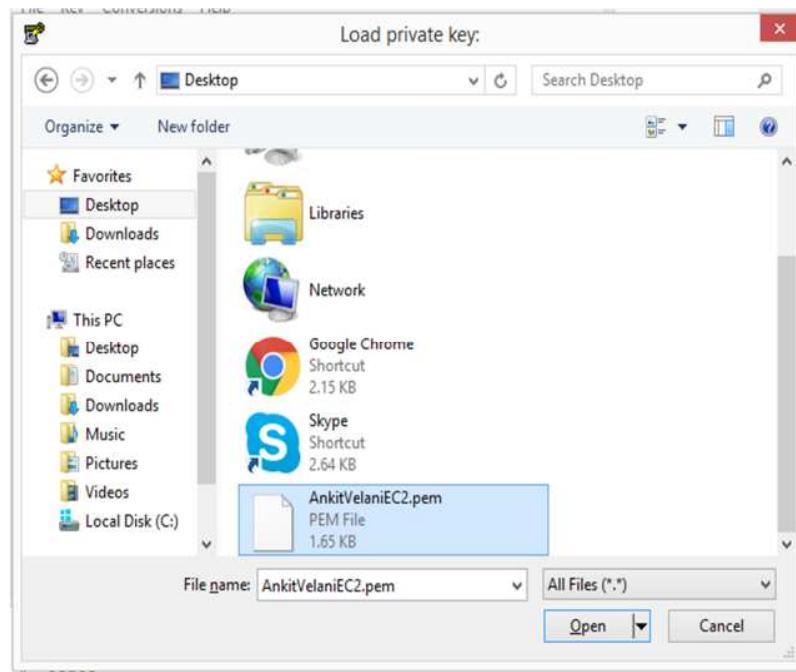
1. Open PuTTygen.



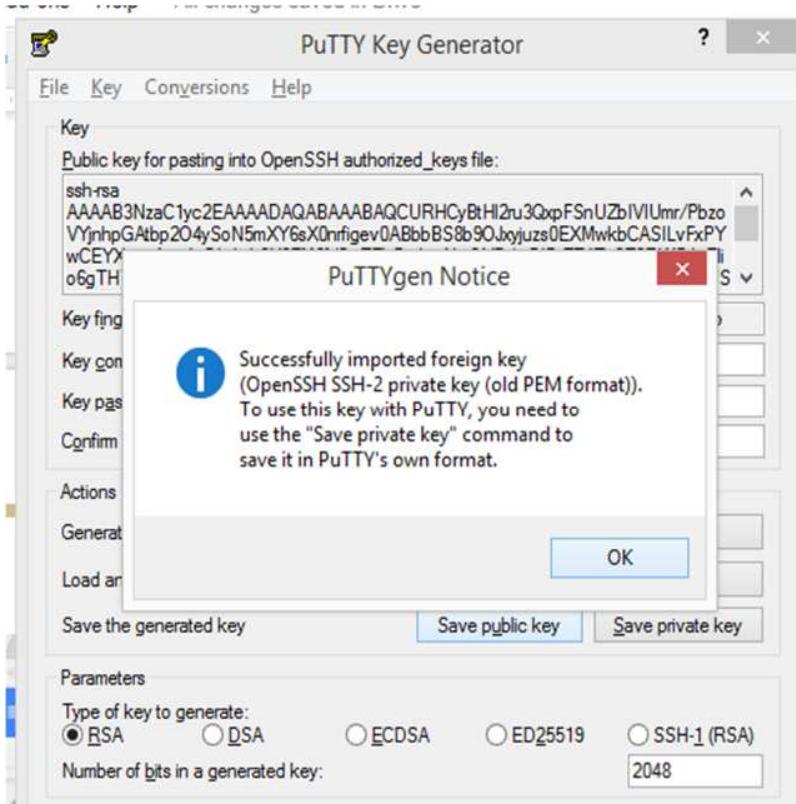
2. Click on **Load**, It gives FileBox , choose file type as **All Files(*)**



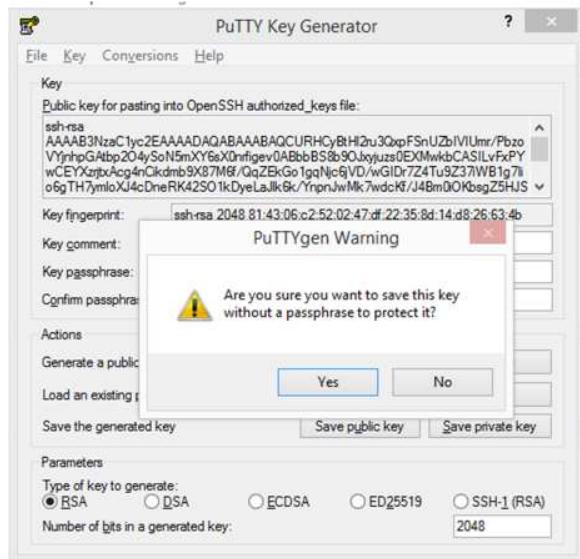
3. Select your .pem file and click on Open.



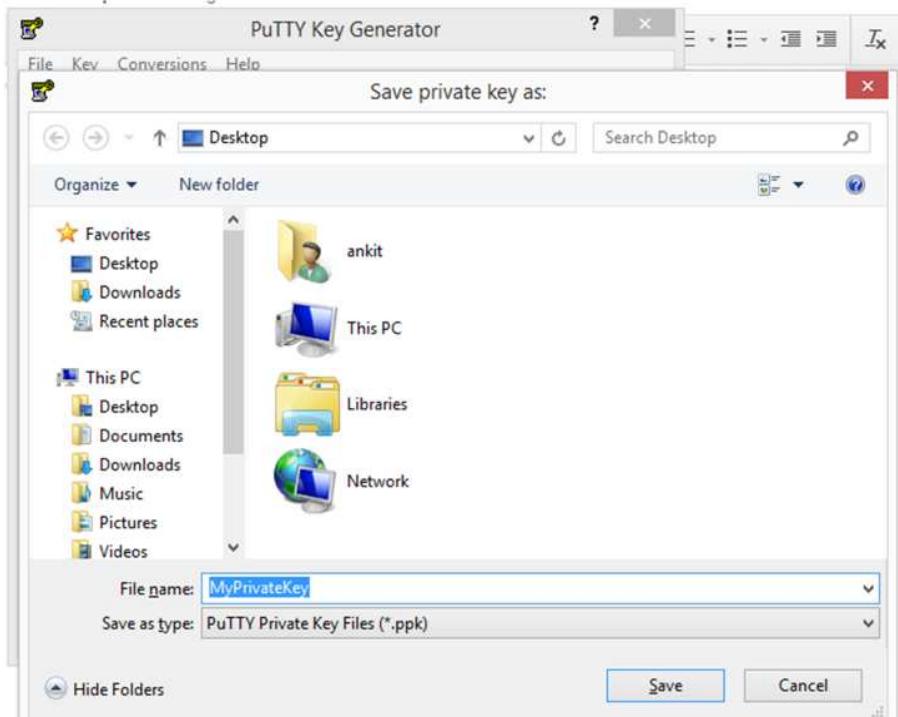
4. Click on Ok , if Putty show any notice.



5. Click on Save private key



6. Click on Yes.



7. File box file open and it allow to enter the filename which you want save the private key. Click on Save.
8. Close the PuttyGen software.

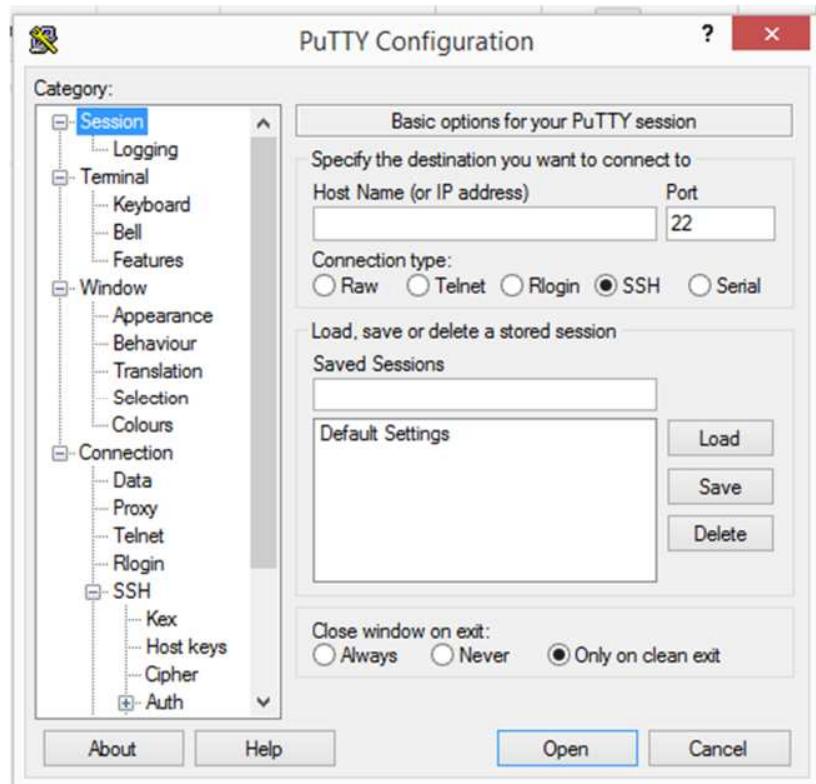
Congratulations :) Successfully created Private key(ppk) file.

Accessing Amazon EC2 Machine using SSH Client

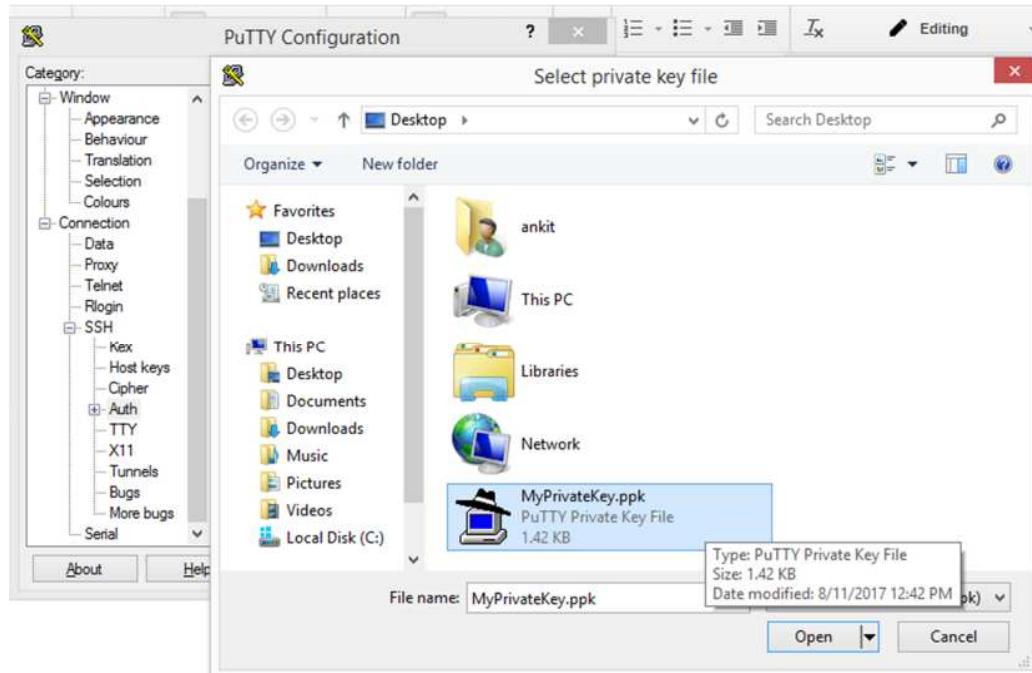
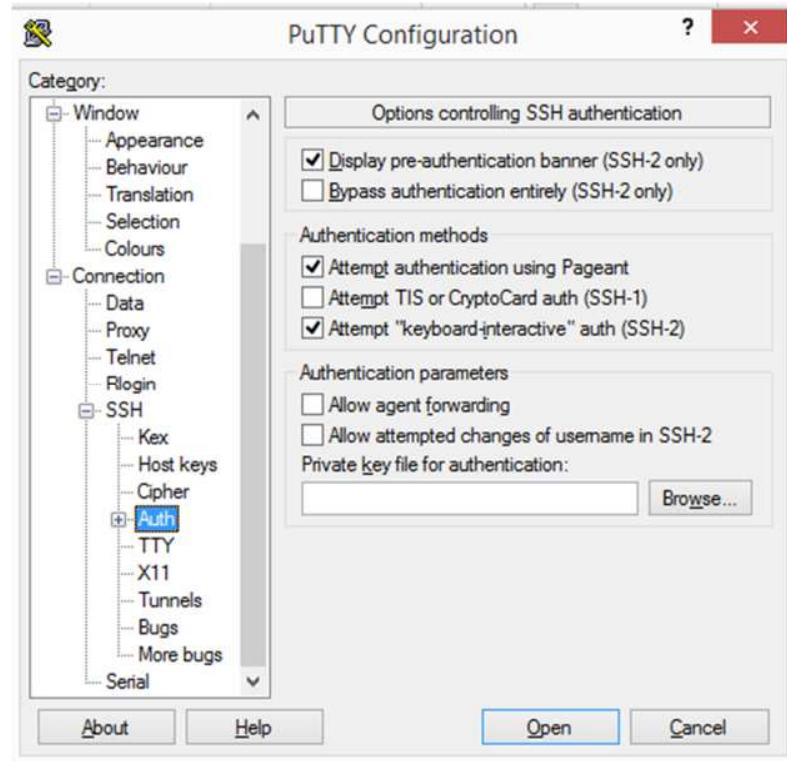
Note: Make sure that Putty ssh client should be installed in your machine.



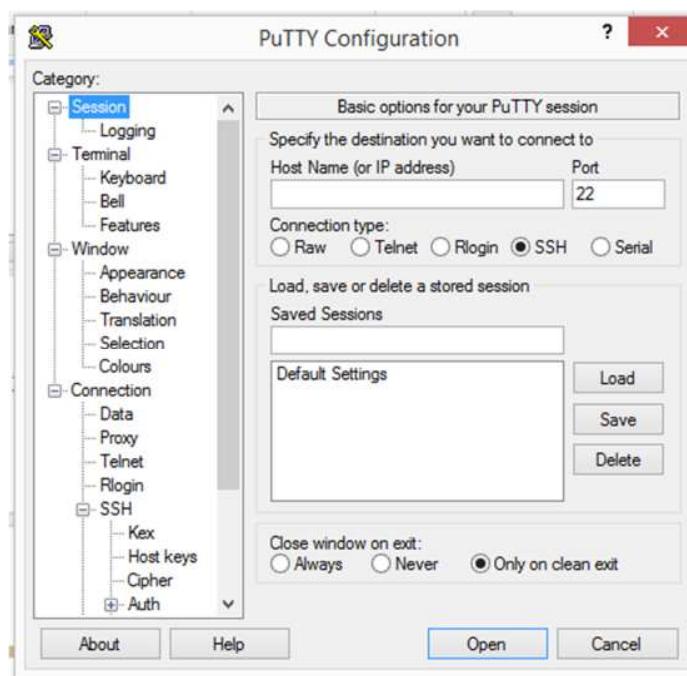
1. Open Putty



2. Go to **Connection > SSH > Auth**, Click on **Browse**, Choose private key (.ppk) file.



3. Click on open.
4. Go to Session.



5. Enter the Hostname (or IP address)
6. Open an Amazon cloud portal. <https://console.aws.amazon.com>

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services: EC2, Cost Explorer, Billing

All services:

- Compute**: EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, Batch
- Storage**: S3, EFS, Glacier, Storage Gateway
- Database**
- Developer Tools**: CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, X-Ray
- Management Tools**: CloudWatch, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog
- Internet of Things**: AWS IoT, AWS Greengrass
- Contact Center**: Amazon Connect
- Game Development**: Amazon GameLift
- Mobile Services**: Mobile Hub, Cognito, Device Farm

Helpful tips

- Manage your costs**: Get real-time billing alerts based on your cost and usage budgets. [Start now](#)
- Create an organization**: Use AWS Organizations for policy-based management of multiple AWS accounts. [Start now](#)

Explore AWS

- New Product Announcements**: View the latest announcements from the AWS Summit - San Francisco. [Learn more](#)
- Migrate from Oracle to Amazon Aurora**: Learn how to migrate from Oracle to Amazon Aurora with minimal downtime. [View project](#)

7. Click **EC2** under Compute Category.
8. Screen will be on EC2 Dashboard.

The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances, AMIs, ELASTIC BLOCK STORE, and NETWORK & SECURITY. The main content area is titled 'Resources' and shows the following resource counts:

- 1 Running Instances
- 2 Elastic IPs
- 0 Dedicated Hosts
- 0 Snapshots
- 2 Volumes
- 0 Load Balancers
- 1 Key Pairs
- 3 Security Groups
- 0 Placement Groups

A callout box points to the 'Launch Instance' button in the 'Create Instance' section. Other sections visible include 'Service Health' (with 'US East (Ohio)' status) and 'Scheduled Events'.

9. Click on Running Instances.

The screenshot shows the AWS EC2 Management Console on the 'Instances' page. The sidebar is identical to the previous screenshot. The main area shows a table of instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
	i-0fcfc4fc3afe2f71e	t2.micro	us-east-2c	running	2/2 checks ...	None	ec2-52-14-202-117.us-east-2.compute.amazonaws.com

Below the table, a detailed view of the selected instance ('i-0fcfc4fc3afe2f71e') is shown. The 'Description' tab is active, displaying the following details:

Instance ID	i-0fcfc4fc3afe2f71e	Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.202.117
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Availability zone	us-east-2c	Private IPs	172.31.43.82

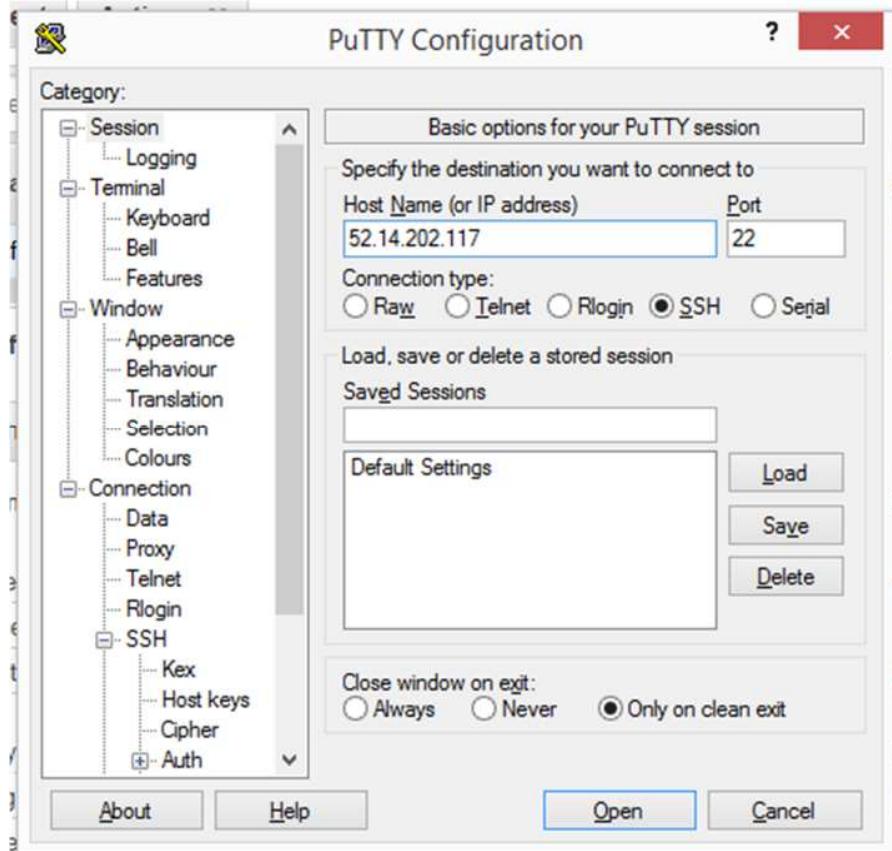
10. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc details.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, and Network & Security. The 'Instances' link is highlighted. The main content area shows a table with one row for an instance named 'i-0fcfc4fc3afe2f71e'. The instance is of type 't2.micro' in 'us-east-2c' and is 'running'. It has a Public DNS of 'ec2-52-14-202-117.us-east-2.compute.amazonaws.com'. Below the table, there are tabs for Description, Status Checks, Monitoring, and Tags. The 'Description' tab is selected, showing detailed information: Instance ID (i-0fcfc4fc3afe2f71e), Public DNS (IPv4) (ec2-52-14-202-117.us-east-2.compute.amazonaws.com), IPv4 Public IP (52.14.202.117), Instance state (running), Instance type (t2.micro), Elastic IPs, Availability zone (us-east-2c), Security groups (launch-wizard-1), Scheduled events (No scheduled events), AMI ID (ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-0230610751)), Private DNS (ip-172-31-43-82.us-east-2.compute.internal), Private IPs (172.31.43.82), Secondary private IPs, VPC ID (vpc-09490460), and Subnet ID (subnet-2304a56e). At the bottom of the page, there are links for Feedback, English, and footer text: © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use.

11. Copy the Public IP.

Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
IPv4 Public IP	52.14.202.117
IPv6 IPs	-
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e

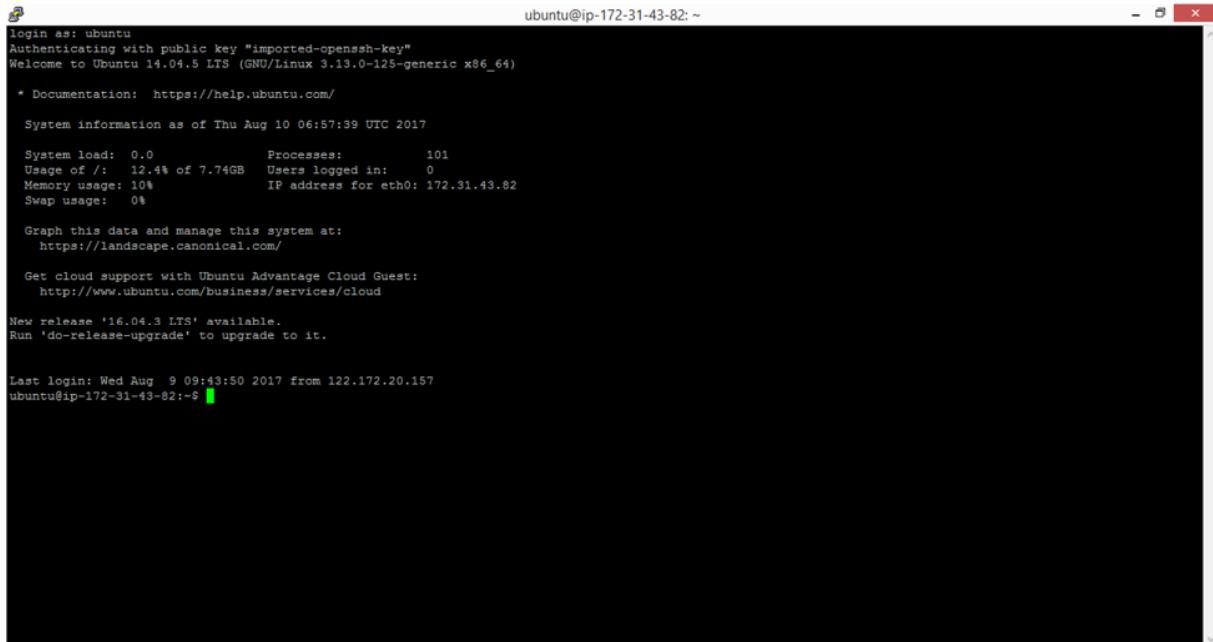
12. Go back to Putty SSH client , enter the Hostname as a Public IP.



13. Click on Open.



-
14. Ask for username , login as : enter **ubuntu** as user name
 15. After successfully login, will have terminal of EC2 virtual server.



```
ubuntu@ip-172-31-43-82: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-125-generic x86_64)

 * Documentation: https://help.ubuntu.com/
 
 System information as of Thu Aug 10 06:57:39 UTC 2017

 System load: 0.0          Processes:      101
 Usage of /: 12.4% of 7.74GB   Users logged in: 0
 Memory usage: 10%          IP address for eth0: 172.31.43.82
 Swap usage: 0%

 Graph this data and manage this system at:
 https://landscape.canonical.com/

 Get cloud support with Ubuntu Advantage Cloud Guest:
 http://www.ubuntu.com/business/services/cloud

New release '16.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Aug  9 09:43:50 2017 from 122.172.20.157
ubuntu@ip-172-31-43-82:~$
```

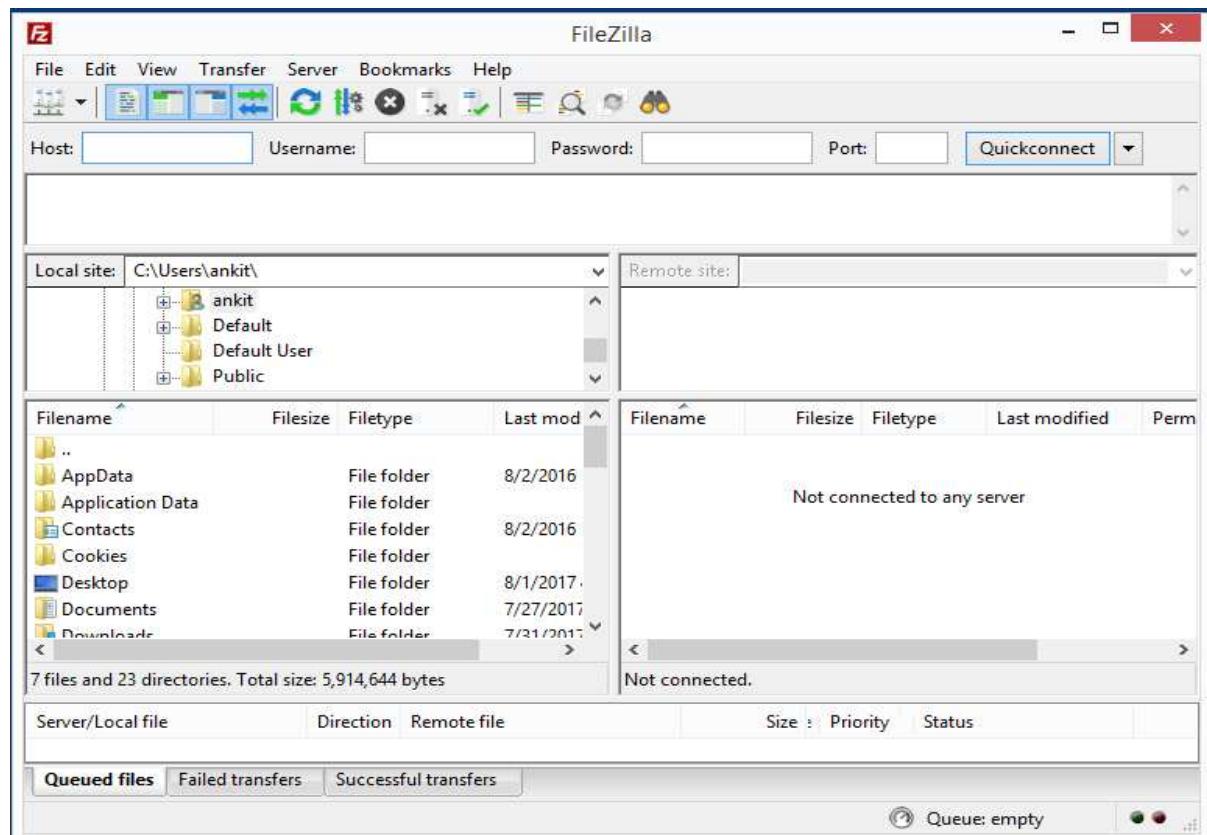
Congratulations :) you are connected to virtual machine using Putty SSH Client.

Accessing Amazon EC2 Machine using FileManager

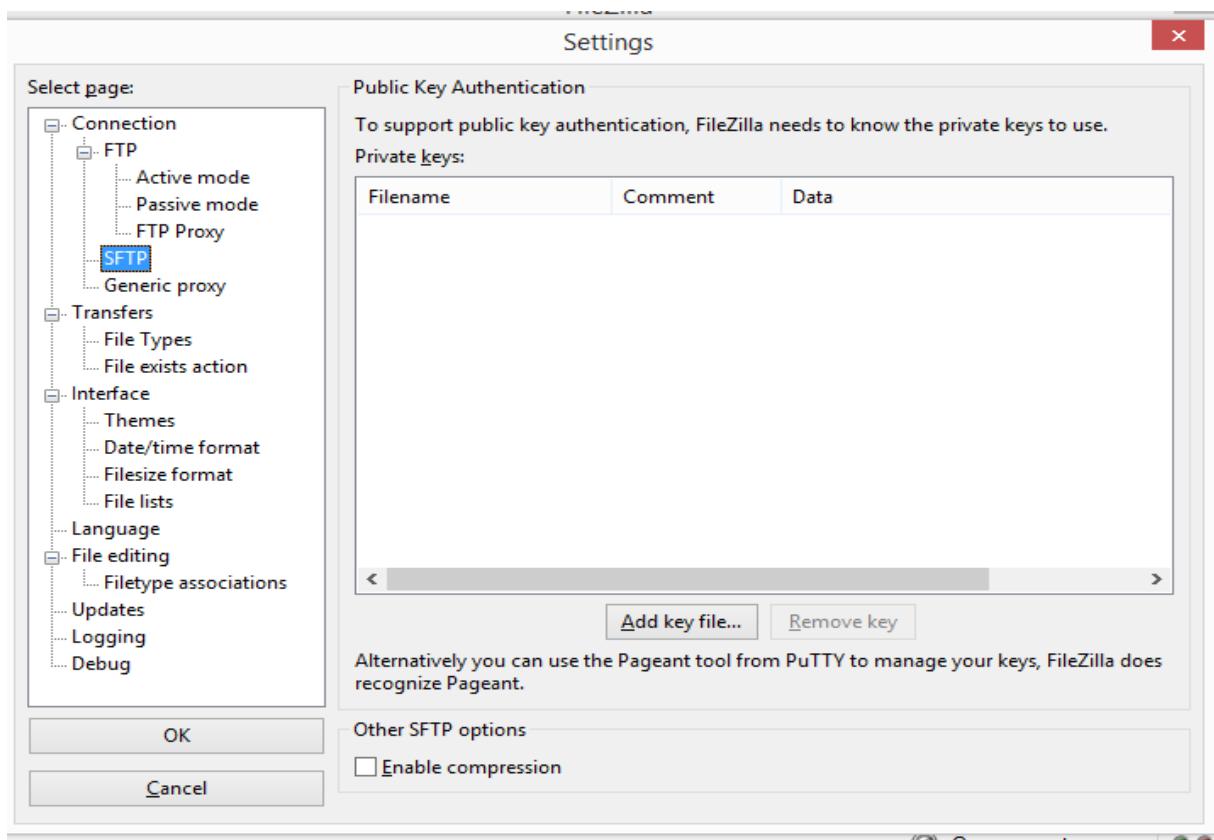
Note: Make sure that any one file manager (FileZilla, WinScp ..etc) should be installed in your machine.



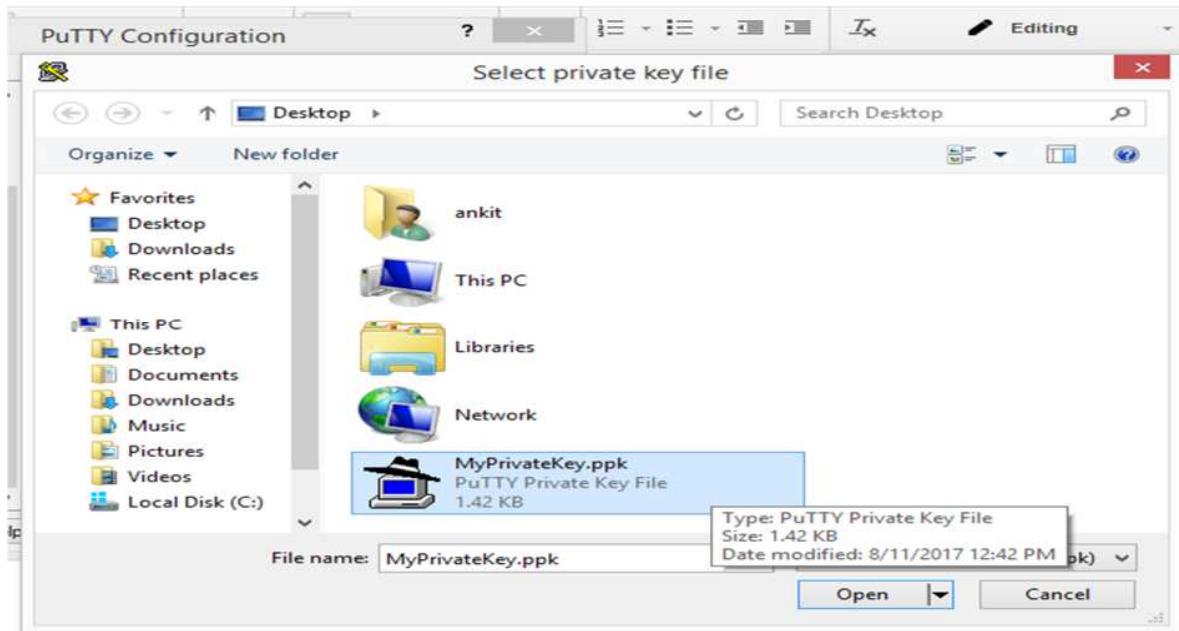
1. Open FileZilla.



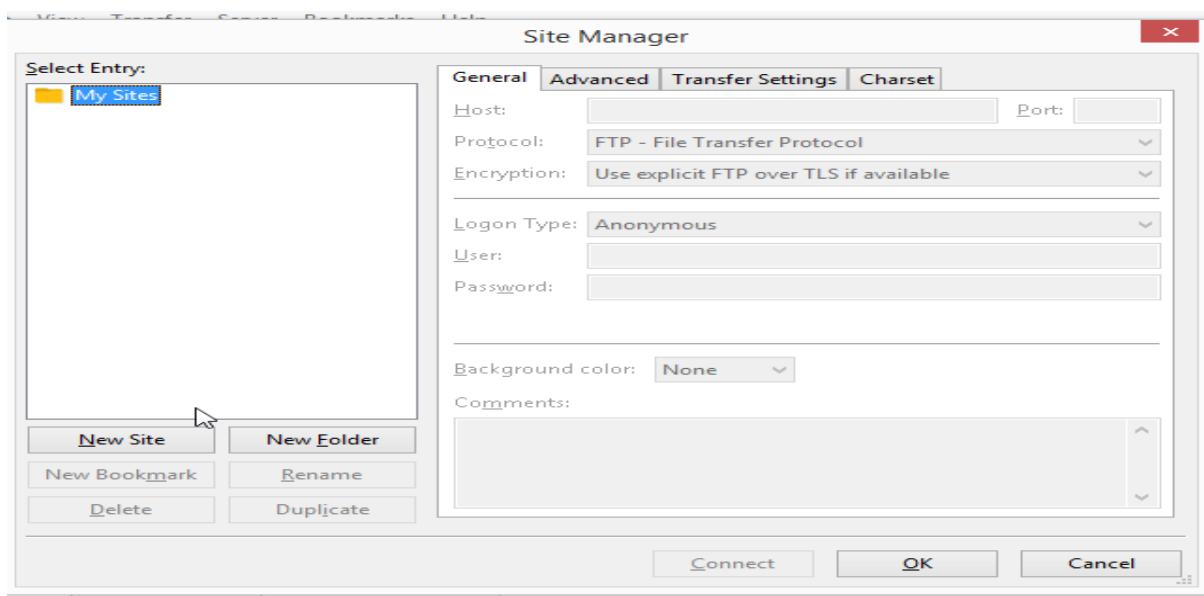
2. Go to **Edit Menu > Connection > SFTP.**



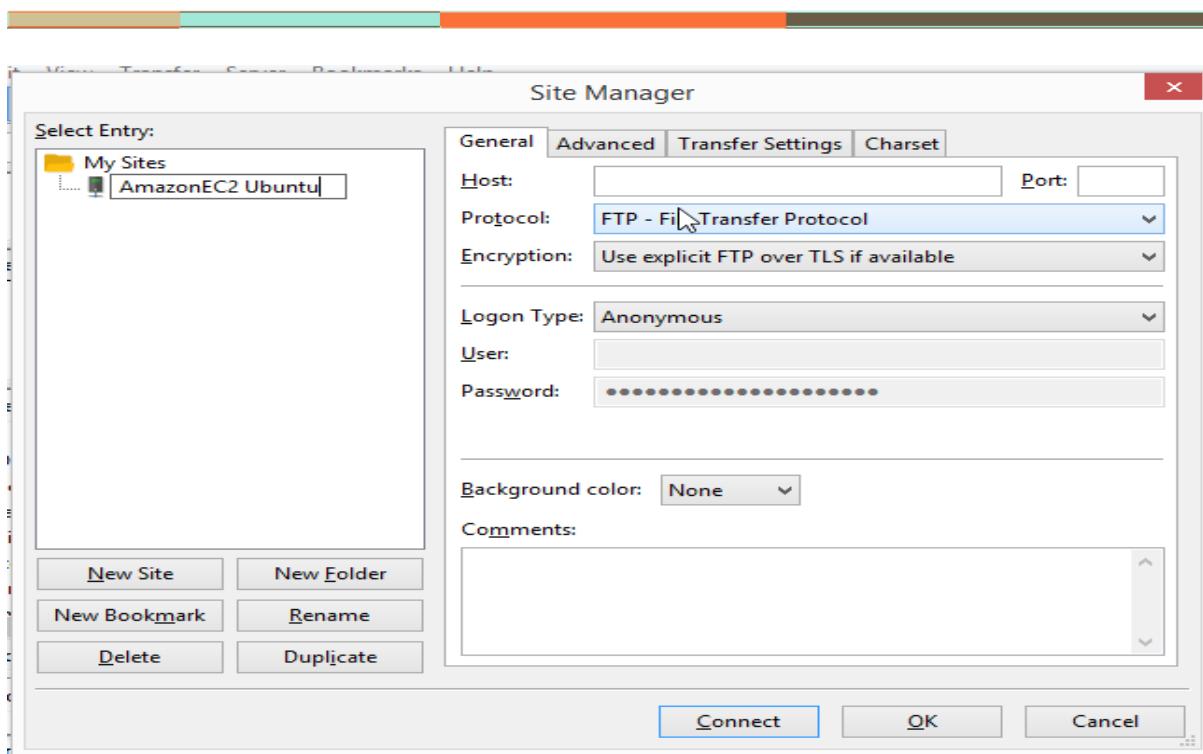
3. Click on **Add key file...**



4. Select your **.pem** or **.ppk** file and click on **Open**.
5. Click on **OK**.
6. Go to **File Menu > Site Manager**.



7. Click on **New Site**.
8. Enter Site Name as **AmazonEC2 Ubuntu**



9. Open an Amazon cloud portal. <https://console.aws.amazon.com>

10. Click **EC2** under Compute Category.

11. Screen will be on EC2 Dashboard.

EC2 Dashboard

- Events
- Tags
- Reports
- Limits

INSTANCES

- Instances
- Spot Requests
- Reserved Instances
- Dedicated Hosts

IMAGES

- AMIs
- Bundle Tasks

ELASTIC BLOCK STORE

- Volumes
- Snapshots

NETWORK & SECURITY

- Security Groups
- Elastic IPs
- Placement Groups
- Key Pairs
- Network Interfaces

Resources

You are using the following Amazon EC2 resources in the US East (Ohio) region:

1 Running Instances	2 Elastic IPs
0 Dedicated Hosts	0 Snapshots
2 Volumes	0 Load Balancers
1 Key Pairs	3 Security Groups
0 Placement Groups	

Just need a simple virtual private server? Get everything you need to jumpstart your project - compute, storage, and networking – for a low, predictable price. Try [Amazon Lightsail](#) for free.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your Instances will launch in the US East (Ohio) region

Service Health

Service Status: US East (Ohio): This region is operating normally.

Scheduled Events

US East (Ohio): No events

Account Attributes

- Supported Platforms: VPC
- Default VPC: vpc-bd2d6dd4
- Resource ID length management

Additional Information

- Getting Started Guide
- Documentation
- All EC2 Resources
- Forums
- Pricing
- Contact Us

AWS Marketplace

Find free software trial products in the AWS Marketplace from the [EC2 Launch Wizard](#). Or try these popular AMIs:

- Barracuda NextGen Firewall F-Series - PAYG

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

12. Click on Running Instances.

EC2 Dashboard

- Events
- Tags
- Reports
- Limits

INSTANCES

- Instances
- Spot Requests
- Reserved Instances
- Dedicated Hosts

IMAGES

- AMIs
- Bundle Tasks

ELASTIC BLOCK STORE

- Volumes
- Snapshots

NETWORK & SECURITY

- Security Groups

Resource Groups

Actions

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public
	i-0fcfc4fc3afe2f71e	t2.micro	us-east-2c	running	2/2 checks ...	None	ec2-5...

Instance: i-0fcfc4fc3afe2f71e Public DNS: ec2-52-14-202-117.us-east-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID	i-0fcfc4fc3afe2f71e	Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.202.117
Instance type	t2.micro	IPv6 IPs	-
Elastic IP		Private DNS	in-172-31-43-87.us-east-

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

13. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc. details.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, and Network & Security. The 'Instances' link is currently selected. The main content area has tabs for Launch Instance, Connect, and Actions. Below that is a search bar and a table with one row. The table columns include Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS. The instance listed is 'i-0fcfc4fc3afe2f71e' with 't2.micro' type, 'us-east-2c' zone, and 'running' state. It has 2/2 checks and no alarms. The Public DNS is 'ec2-52-14-202-117.us-east-2.compute.amazonaws.com'. At the bottom, there are tabs for Description, Status Checks, Monitoring, and Tags, followed by detailed instance information.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
	i-0fcfc4fc3afe2f71e	t2.micro	us-east-2c	running	2/2 checks ...	None	ec2-52-14-202-117.us-east-2.compute.amazonaws.com

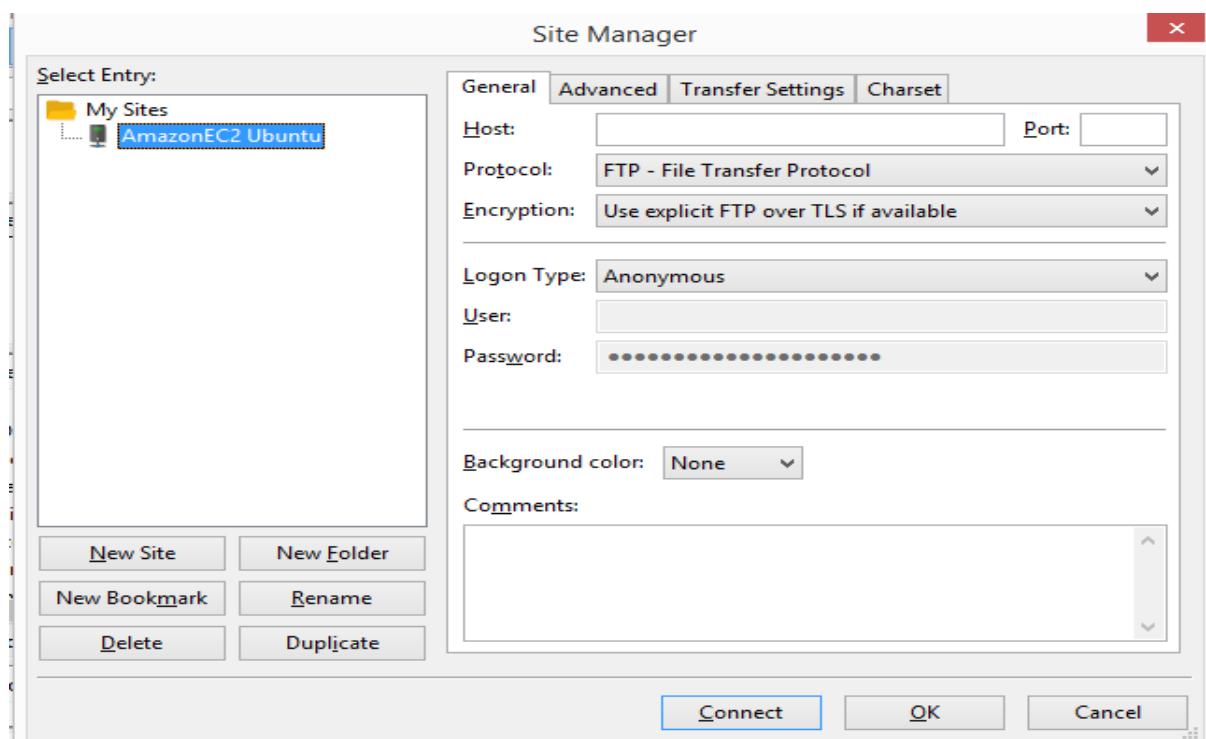
Description

Instance ID	i-0fcfc4fc3afe2f71e	Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.202.117
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Availability zone	us-east-2c	Private IPs	172.31.43.82
Security groups	launch-wizard-1 . view inbound rules	Secondary private IPs	
Scheduled events	No scheduled events	VPC ID	vpc-09490460
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-43201076)	Subnet ID	subnet-2304a56e

14. Copy the Public IP.

Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
IPv4 Public IP	52.14.202.117
IPv6 IPs	-
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e

15. Next , Open the FileZilla. And Go to **File Menu > SiteManager > AmazonEC2 Ubuntu**



16. Enter following details.

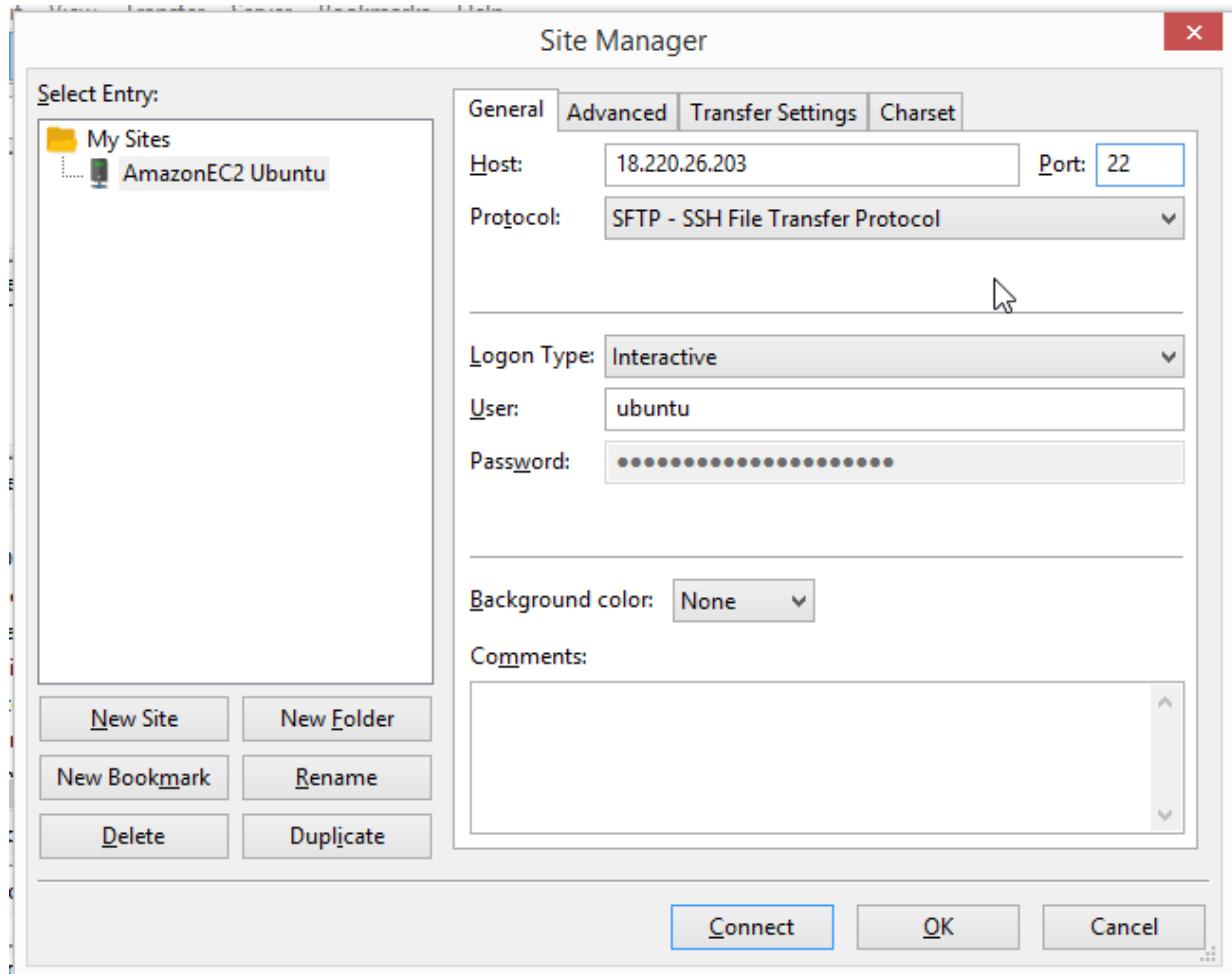
Host : Public IP

Port : 22

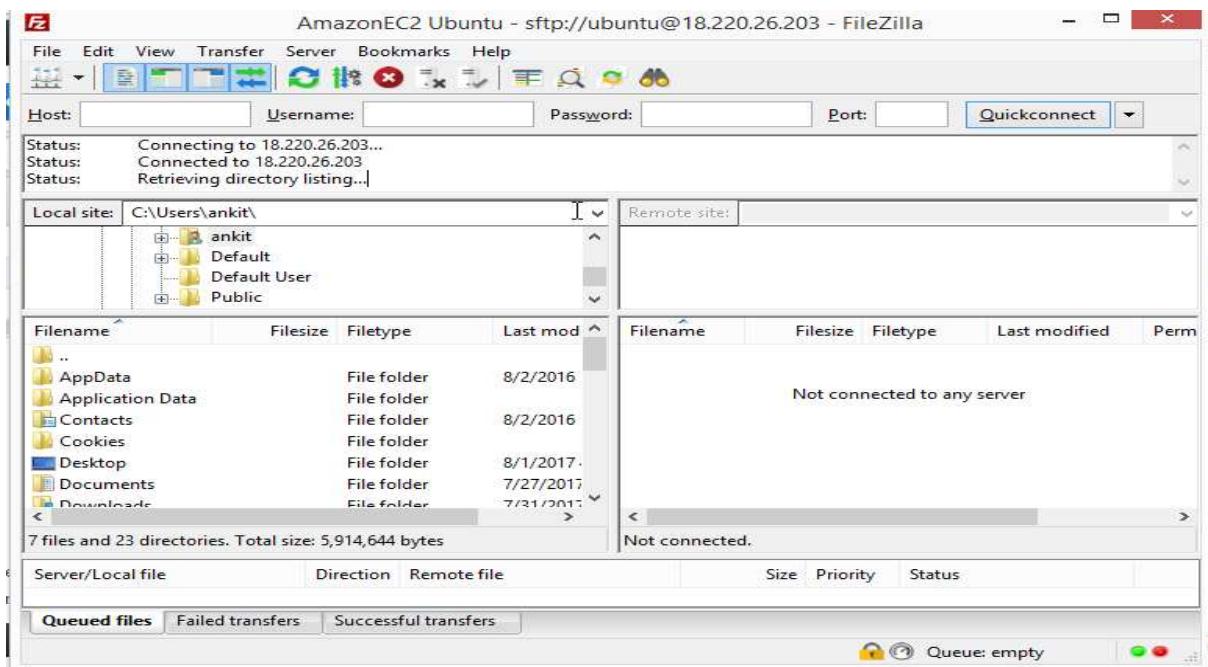
Protocol : SFTP

Logon Type : Interactive

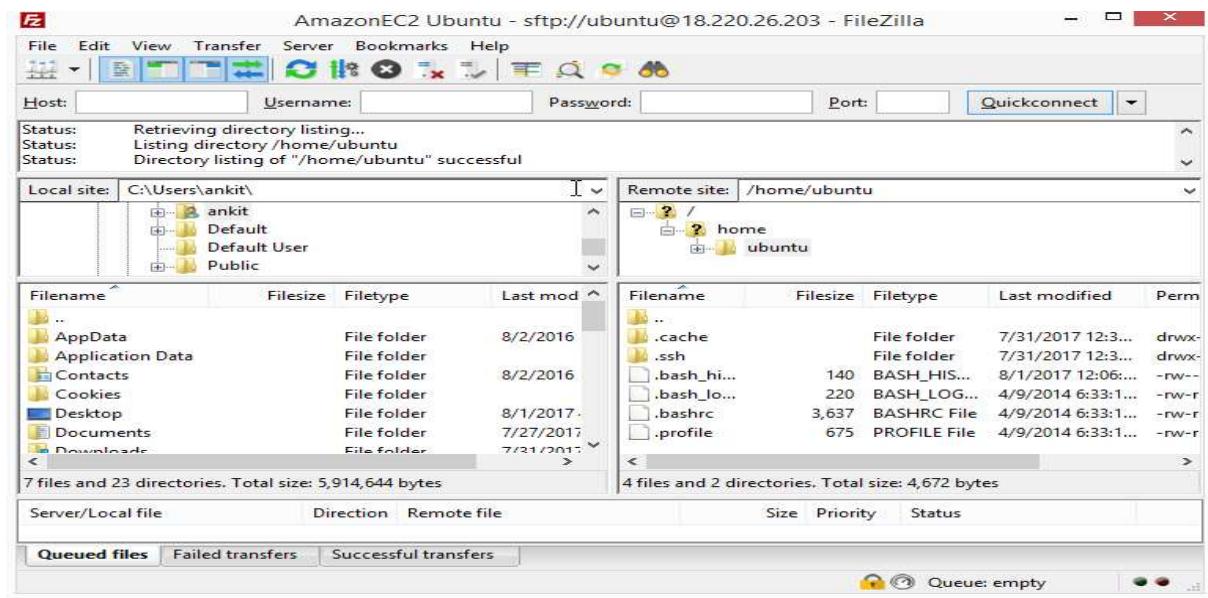
User : ubuntu



17. Click on connect and it will take you on another screen.



18. Once it successfully connected it will show the Remote Site in the right side of the box.

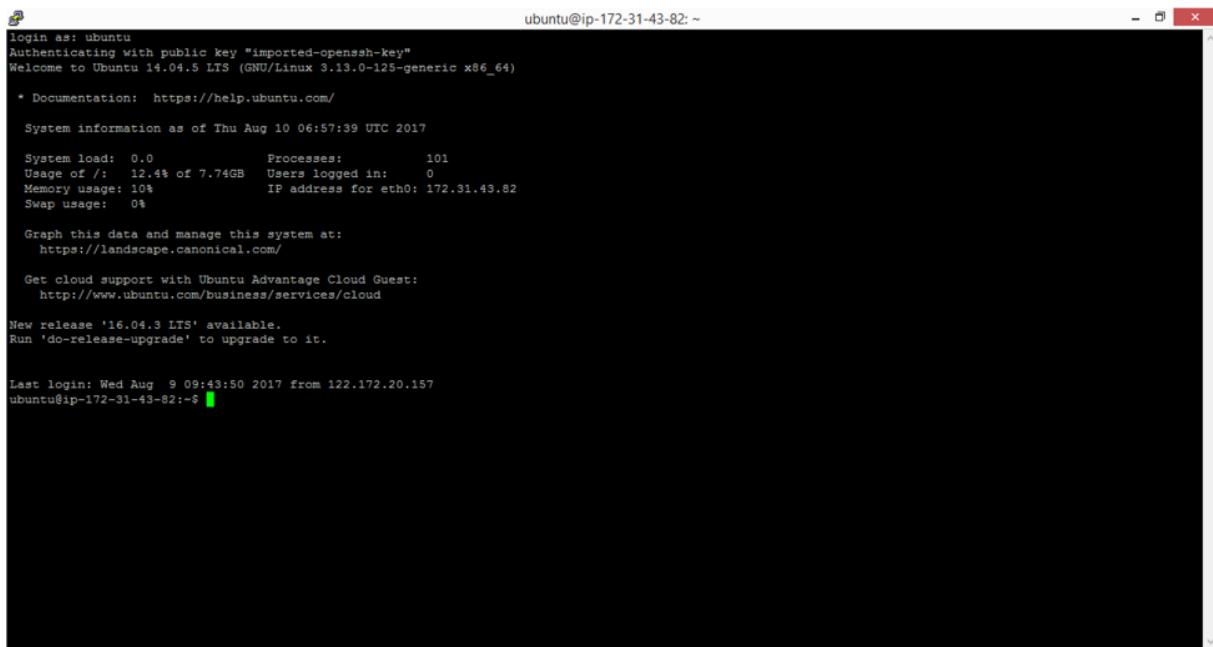


Note: You are accessing Amazon EC Virtual machine using FileZilla file manager, it's easy to download, upload and file manipulation.

Congratulations :) you are connected to virtual machine using File Manager.

Configuring Apache2 on Ubuntu 14

Note: Before you start executing command, first connect EC-2 with Putty SSH client.



```

login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-125-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
 
 System information as of Thu Aug 10 06:57:39 UTC 2017

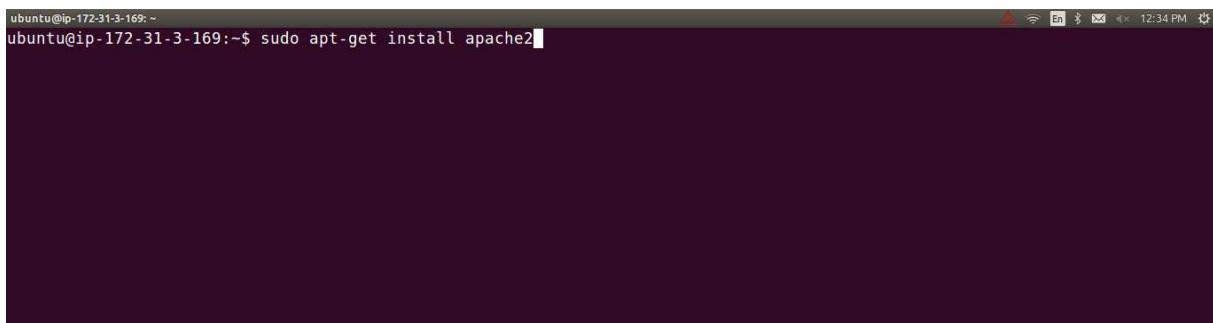
 System load: 0.0          Processes:           101
 Usage of /: 12.4% of 7.74GB   Users logged in: 0
 Memory usage: 10%          IP address for eth0: 172.31.43.82
 Swap usage: 0%
 
 Graph this data and manage this system at:
 https://landscape.canonical.com/
 
 Get cloud support with Ubuntu Advantage Cloud Guest:
 http://www.ubuntu.com/business/services/cloud

 New release '16.04.3 LTS' available.
 Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Aug  9 09:43:50 2017 from 122.172.20.157
ubuntu@ip-172-31-43-82:~
```

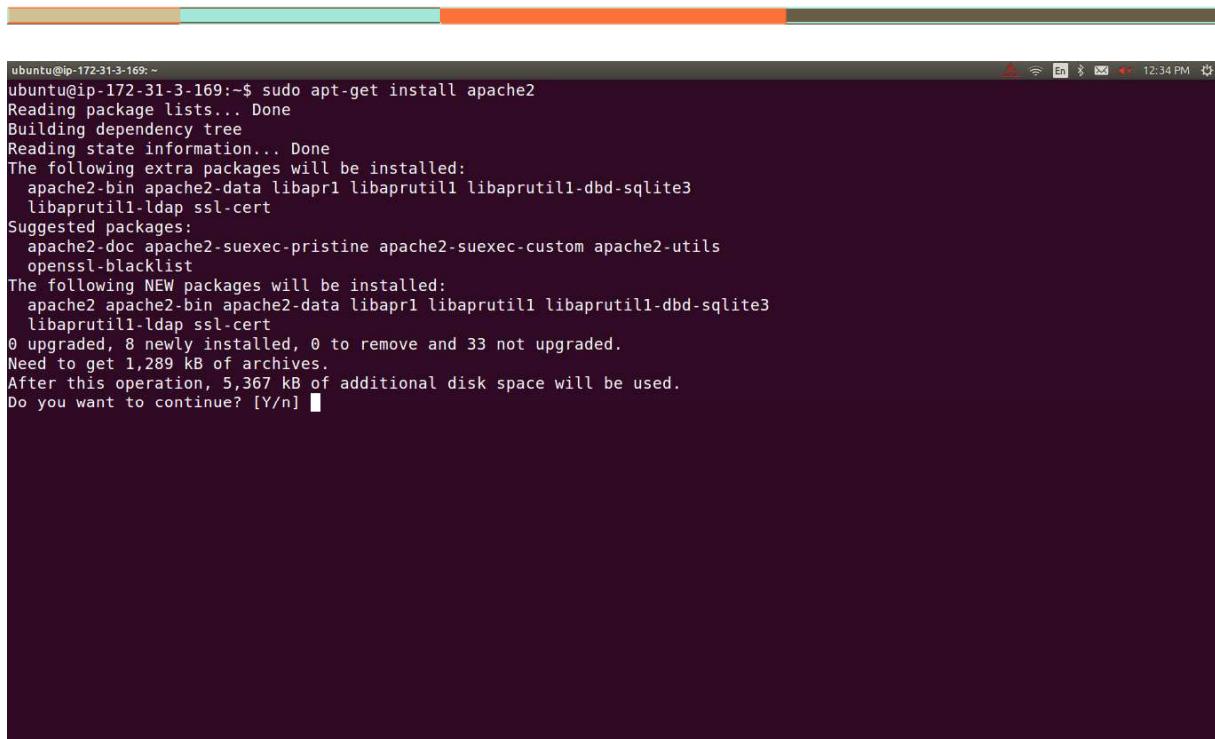
1. Execute command

```
sudo apt-get update
sudo apt-get install apache2
```



```

ubuntu@ip-172-31-3-169:~
ubuntu@ip-172-31-3-169:~$ sudo apt-get install apache2
```

A screenshot of a terminal window on an Ubuntu desktop environment. The terminal shows the command `sudo apt-get install apache2` being run, along with its output. The output includes package lists, dependency building, state information, extra packages to be installed (such as apache2-bin, apache2-data, libapr1, libaprutil1, libaprutil1-dbd-sqlite3, libaprutil1-ldap, and ssl-cert), suggested packages (apache2-doc, apache2-suexec-pristine, apache2-suexec-custom, apache2-utils, and openssl-blacklist), and a summary of the operation. It indicates 0 upgraded, 8 newly installed, 0 to remove, and 33 not upgraded, totaling 1,289 kB to be downloaded from 5,367 kB of additional disk space. A question is posed: "Do you want to continue? [Y/n]".

```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-bin apache2-data libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine apache2-suexec-custom apache2-utils
  openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap ssl-cert
0 upgraded, 8 newly installed, 0 to remove and 33 not upgraded.
Need to get 1,289 kB of archives.
After this operation, 5,367 kB of additional disk space will be used.
Do you want to continue? [Y/n] 
```

2. Enter [Y] for continue installation of an apache2
3. Change the permission to html directory , so outside it can be accessible.

`sudo chmod -R 777 /var/www/html`

Congratulations :) Apache service is installed on Ubuntu EC2 machine.

Configuring PHP on Ubuntu 14

1. Execute command

```
sudo apt-get update
sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

A screenshot of a terminal window on an Ubuntu 14.04 LTS system. The terminal is dark-themed. At the top, there's a status bar with icons for signal strength, battery level, and the time (11:44 AM). The main area shows the command being typed:

```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

A screenshot of a terminal window showing the output of the apt-get install command. The terminal shows the progress of the package installation, including reading package lists, building dependency trees, and checking state information. It also lists extra packages to be installed (libmcrypt4, php5-cli, php5-common, php5-json, php5-readline) and suggested packages (php-pear, libmcrypt-dev, mcrypt, php5-user-cache). The output indicates 0 upgraded, 8 newly installed, 0 to remove, and 33 not upgraded. A total of 4,950 kB of disk space will be used. The user is prompted with 'Do you want to continue? [Y/n]'. The terminal has a dark theme with a light-colored status bar at the top.

```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libmcrypt4 php5-cli php5-common php5-json php5-readline
Suggested packages:
  php-pear libmcrypt-dev mcrypt php5-user-cache
The following NEW packages will be installed:
  libapache2-mod-php5 libmcrypt4 php5 php5-cli php5-common php5-json
  php5-mcrypt php5-readline
0 upgraded, 8 newly installed, 0 to remove and 33 not upgraded.
Need to get 4,950 kB of archives.
After this operation, 20.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] ■
```

2. Enter [Y] for continue installation of an php
3. Next, Restart the apache2 service to recognise php is installed.

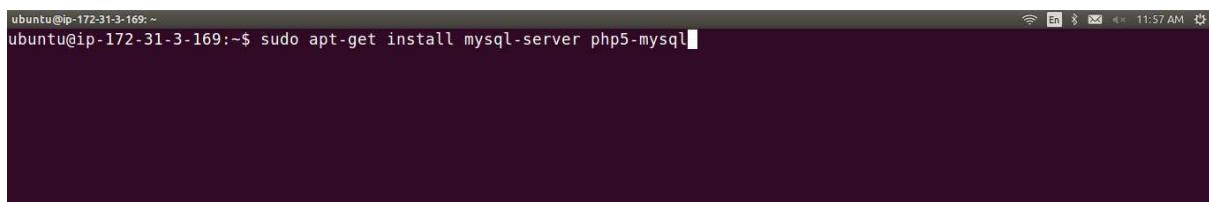
sudo service apache2 restart

Congratulations :) PHP is installed on Ubuntu EC2 machine.

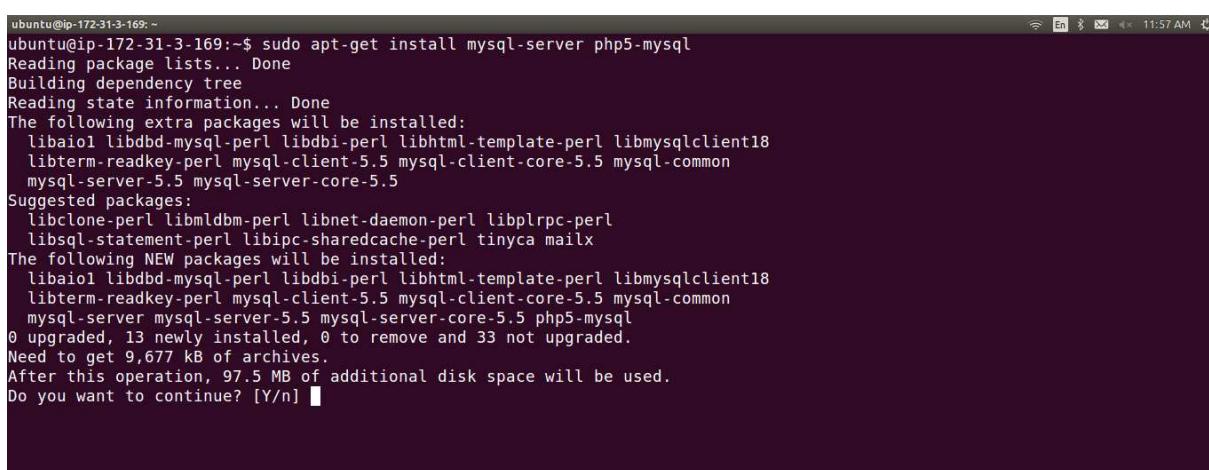
Configuring MySQL on Ubuntu 14

1. Execute command

```
sudo apt-get update
sudo apt-get install mysql-server php5-mysql
```



```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install mysql-server php5-mysql
[Output of package installation process]
```



```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install mysql-server php5-mysql
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
 libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
 libterm-readkey-perl mysql-client-5.5 mysql-client-core-5.5 mysql-common
 mysql-server-5.5 mysql-server-core-5.5
Suggested packages:
 libclone-perl libldb-perl libnet-daemon-perl libplrpc-perl
 libsql-statement-perl libipc-sharedcache-perl tinyca mailx
The following NEW packages will be installed:
 libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
 libterm-readkey-perl mysql-client-5.5 mysql-client-core-5.5 mysql-common
 mysql-server mysql-server-5.5 mysql-server-core-5.5 php5-mysql
0 upgraded, 13 newly installed, 0 to remove and 33 not upgraded.
Need to get 9,677 kB of archives.
After this operation, 97.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] ■
```

2. Enter [Y] for continue installation of MySql

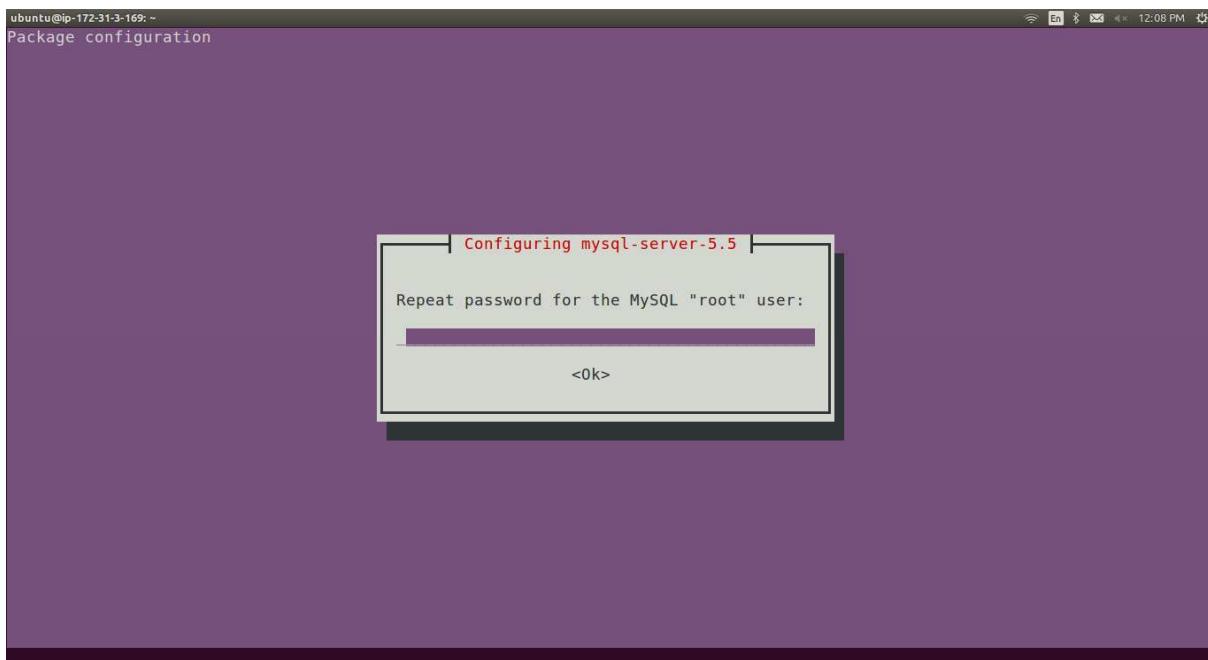
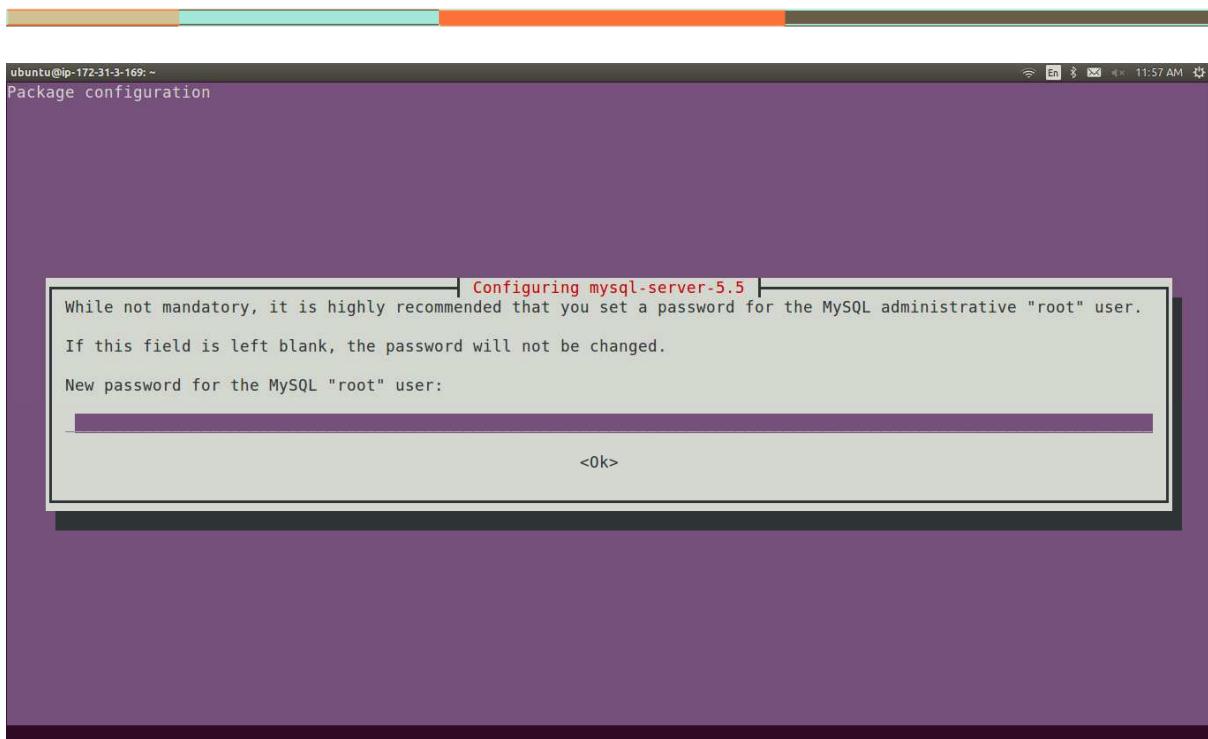
3. Next, MySQL will ask username for root user

Provide password: **root**

At the end we have user credential for MySQL database.

Username: root

Password: root



4. Next, restart the apache2 service to recognize MySQL-server is installed.

```
sudo service apache2 restart
```

Congratulations :) MySQL is installed on Ubuntu EC2 machine.

Configuring HTTP and HTTPS port on EC2 (Ubuntu)

1. Open an Amazon cloud portal. <https://console.aws.amazon.com>

The screenshot shows the AWS Management Console homepage. On the left, there's a sidebar titled "AWS services" with sections for "Recently visited services" (EC2), "All services" (Compute, Storage, Database), and "Helpful tips". The "Compute" section includes EC2, Lambda, and Batch. The "Storage" section includes S3, EFS, Glacier, and Storage Gateway. The "Database" section includes MySQL, PostgreSQL, Oracle, MongoDB, and Amazon RDS. The "Helpful tips" section includes "Manage your costs" (real-time billing alerts) and "Create an organization" (AWS Organizations for policy-based management). On the right, there's a "Explore AWS" section with links to "New Product Announcements" (AWS Summit - San Francisco), "Migrate from Oracle to Amazon Aurora" (minimal downtime), and "Contact Center" (Amazon Connect).

2. Click EC2 under Compute Category.
3. Screen will be on EC2 Dashboard.

The screenshot shows the EC2 Management Console dashboard. The left sidebar has sections for EC2 Dashboard, Instances, Images, Elastic Block Store, Network & Security, and Support. The main area is titled "Resources" and shows statistics for Running Instances (1), Dedicated Hosts (0), Volumes (2), Key Pairs (1), and Placement Groups (0). It also features a callout for Amazon Lightsail. Below this is a "Create Instance" section with a "Launch Instance" button. The right side shows "Account Attributes" like Supported Platforms (VPC), Default VPC (vpc-bd2d6dd4), and Resource ID length management. There are also sections for "Additional Information" (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us) and "AWS Marketplace" (free software trial products).

4. Click on Running Instances.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images (AMIs, Bundle Tasks), Elastic Block Store (Volumes, Snapshots), and Network & Security (Security Groups). The main area has tabs for Launch Instance, Connect, and Actions. A search bar at the top says "Filter by tags and attributes or search by keyword". Below it is a table with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS. One row is visible: Name i-0fcfc4fc3afe2f71e, Instance ID i-0fcfc4fc3afe2f71e, Instance Type t2.micro, Availability Zone us-east-2c, Instance State running, Status Checks 2/2 checks, Alarm Status None, and Public DNS ec2-52-14-202-117.us-east-2.compute.amazonaws.com. Below the table is a detailed view for the selected instance, showing fields like Description, Status Checks, Monitoring, and Tags. The instance details include Instance ID (i-0fcfc4fc3afe2f71e), Instance state (running), Instance type (t2.micro), Elastic IPs, Availability zone (us-east-2c), Public DNS (ec2-52-14-202-117.us-east-2.compute.amazonaws.com), IPv4 Public IP (52.14.202.117), IPv6 IPs (-), Private DNS (ip-172-31-43-82.us-east-2.compute.internal), and Private IPs (172.31.43.82).

- Click on Security group for example here is **launch-wizard-1** name for security group.

The screenshot shows the AWS Instance details page for the instance i-0fcfc4fc3afe2f71e. It includes tabs for Description, Status Checks, Monitoring, and Tags. The instance details table shows the following information:

Instance ID	i-0fcfc4fc3afe2f71e
Instance state	running
Instance type	t2.micro
Elastic IPs	-
Availability zone	us-east-2c
Security groups	launch-wizard-1 , view inbound rules
Scheduled events	No scheduled events
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-43391926)

At the bottom, there's a URL in the address bar: amazonaws.com/ec2/v2/home?region=us-east-2#SecurityGroups:groupId=sg-906393f8;sort=groupId. The status bar at the bottom right says "© 2008 - 2017, Amazon Internet Services Private Ltd, or its affiliates. All rights reserved." and links for Privacy Policy and Terms of Use.

- It show the security group information for the selected group name.

The screenshot shows the AWS EC2 Dashboard with the 'Resource Groups' tab selected. On the left, a sidebar lists various services: EC2 Dashboard, Events, Tags, Reports, Limits, Instances, AMIs, Elastic Block Store, Volumes, Snapshots, Network & Security, and Security Groups. The 'Security Groups' section is expanded, showing 'Elastic IPs', 'Placement Groups', and 'Key Pairs'. In the main content area, a table lists one security group:

Name	Group ID	Group Name	VPC ID	Description
sg-906393f8		launch-wizard-1	vpc-09490460	launch-wizard-1 created 2017-08-09T15:0...

Below the table, a detailed view of the security group 'sg-906393f8' is shown. The 'Inbound' tab is selected. The group name is 'launch-wizard-1', group description is 'launch-wizard-1 created 2017-08-09T15:02:54.361+05:30', VPC ID is 'vpc-09490460', and Group ID is 'sg-906393f8'. The 'Edit' button is visible.

7. Click on Inbound

This screenshot is identical to the previous one, showing the AWS EC2 Security Groups page with the 'Inbound' tab selected. The security group 'sg-906393f8' is listed with its details: Group name 'launch-wizard-1', Group description 'launch-wizard-1 created 2017-08-09T15:02:54.361+05:30', VPC ID 'vpc-09490460', and Group ID 'sg-906393f8'. The 'Edit' button is highlighted.

8. Click on Edit

The screenshot shows the AWS EC2 Management Console with the 'Resource Groups' tab selected. A search bar at the top has 'Group ID: sg-906393f8' and 'Add filter'. Below it is a table with columns: Name, Group ID, Group Name, VPC ID, and Description. A modal window titled 'Edit inbound rules' is open, showing a table with one row: Type: SSH, Protocol: TCP, Port Range: 22, Source: Custom 0.0.0.0/0. There is also an 'Add Rule' button.

9. Click on Add Rule

This screenshot is similar to the previous one, but the 'Add Rule' button is highlighted with a red box. The 'Protocol' dropdown is set to 'TCP' and the 'Port Range' dropdown is set to '22'. The 'Source' dropdown is set to 'Custom 0.0.0.0/0'.

The screenshot shows the AWS EC2 Management Console with the 'Edit inbound rules' dialog open. The dialog has four columns: Type, Protocol, Port Range, and Source. There are two rows of rules: one for SSH (TCP port 22) with Source set to 'Custom' and '0.0.0.0/0'; and another for HTTP (TCP port 80) with Source set to 'Anywhere'. A note at the bottom of the dialog box states: 'NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.' At the bottom right of the dialog are 'Cancel' and 'Save' buttons.

10. Select Rule type as **HTTP** and Source as **Anywhere**.

11. Click on **Save**.

12. For adding more rules follow the same steps.

13. Copy EC-2 public IP and access from the web-browser.

Description		Status Checks		Monitoring		Tags	
Instance ID	i-0fcfc4fc3afe2f71e	Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com				
Instance state	running	IPv4 Public IP	52.14.202.117				
Instance type	t2.micro	IPv6 IPs	-				
Elastic IPs		Private DNS	ip-172-31-43-82.us-east-2.compute.internal				
Availability zone	us-east-2c	Private IPs	172.31.43.82				
Security groups	launch-wizard-1 . view inbound rules	Secondary private IPs					
Scheduled events	No scheduled events	VPC ID	vpc-09490460				
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-13291026)	Subnet ID	subnet-2304a56e				

Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
IPv4 Public IP	52.14.202.117
IPv6 IPs	-
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e



14. If Web-browser is displaying Apache2 Ubuntu Default Page then we have successfully configures HTTP port on EC-2.

Congratulation :) Now EC-2 Machine can be accessible using HTTP port.

Configuring HTTP and HTTPS port on EC2 (Windows)

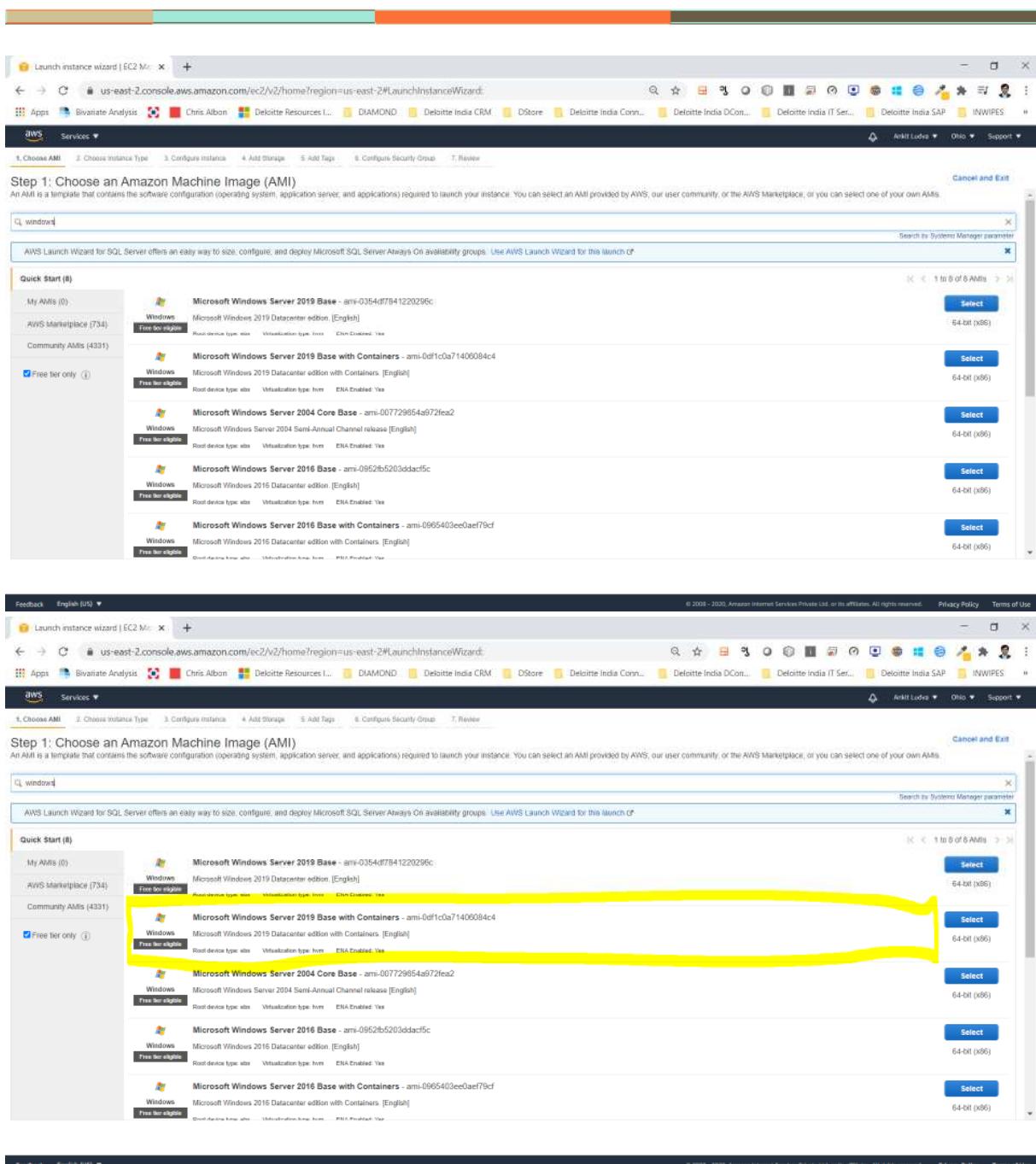
1. Open an Amazon cloud portal. <https://console.aws.amazon.com>

The screenshot shows the AWS Management Console homepage. On the left, there's a sidebar titled "AWS services" with sections for "Recently visited services" (EC2), "All services" (Compute, Storage, Database), and "Helpful tips" (Manage your costs, Create an organization). The main content area has sections for "Explore AWS" (New Product Announcements, Migrate from Oracle to Amazon Aurora) and "Service Health Dashboard".

2. Click **EC2** under Compute Category.
3. Click on Launch Instance.

The screenshot shows the "New EC2 Experience" console. The left sidebar includes sections for Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs), and Elastic Block Store (Volumes, Snapshots, Lifecycle Manager). The main area displays "Resources" (Instances, Snapshots, Security groups, Running instances) and "Account attributes" (Supported platforms: VPC, Default VPC: vpc-f22c8699, Settings, EBS encryption, Zones, Default credit specification, Console experiments). It also features "Service health" (Region: US East (Ohio), Status: This service is operating normally) and "Zone status" (Zone: us-east-1a (use2-ac1), Status: Zone is operating normally).

4. Choose AMI (Amazon Machine Image) which is **free tier** and **Windows** type.



The screenshot shows the AWS Launch Instance Wizard Step 1: Choose an Amazon Machine Image (AMI). The search bar at the top has "windows" typed into it. A list of AMIs is displayed, with the first item, "Microsoft Windows Server 2019 Base - ami-0354df7841220296c", highlighted with a yellow arrow pointing to it. This item is described as "Microsoft Windows 2019 Datacenter edition [English]" and "Root device type: hvm Virtualization type: hvm ENA Enabled: Yes". There are "Select" buttons next to each item, and the "Microsoft Windows Server 2019 Base with Containers" item also has a "Select" button.

5. Select Microsoft Windows Server 2019 Base with Containers AMI type.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance families ▾ Current generation ▾ Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~ 1 GB memory, EBS only)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
t2	t2.micro (Free tier eligible)	1	1	EBS only	-	Low to Moderate	Yes
t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

6. Select t2.micro type instance, which is an only eligible for free tier type.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

Microsoft Windows Server 2019 Base with Containers - ami-0dfc0a71408084c4
Free tier eligible Microsoft Windows 2019 Datacenter edition with Containers, [English]
Root Device Type: /dev/sda1 Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	1	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name: launch-wizard-10
Description: launch-wizard-10 created 2020-11-04T09:31:33.737+05:30

Type	Protocol	Port Range	Source	Description
This security group has no rules.				

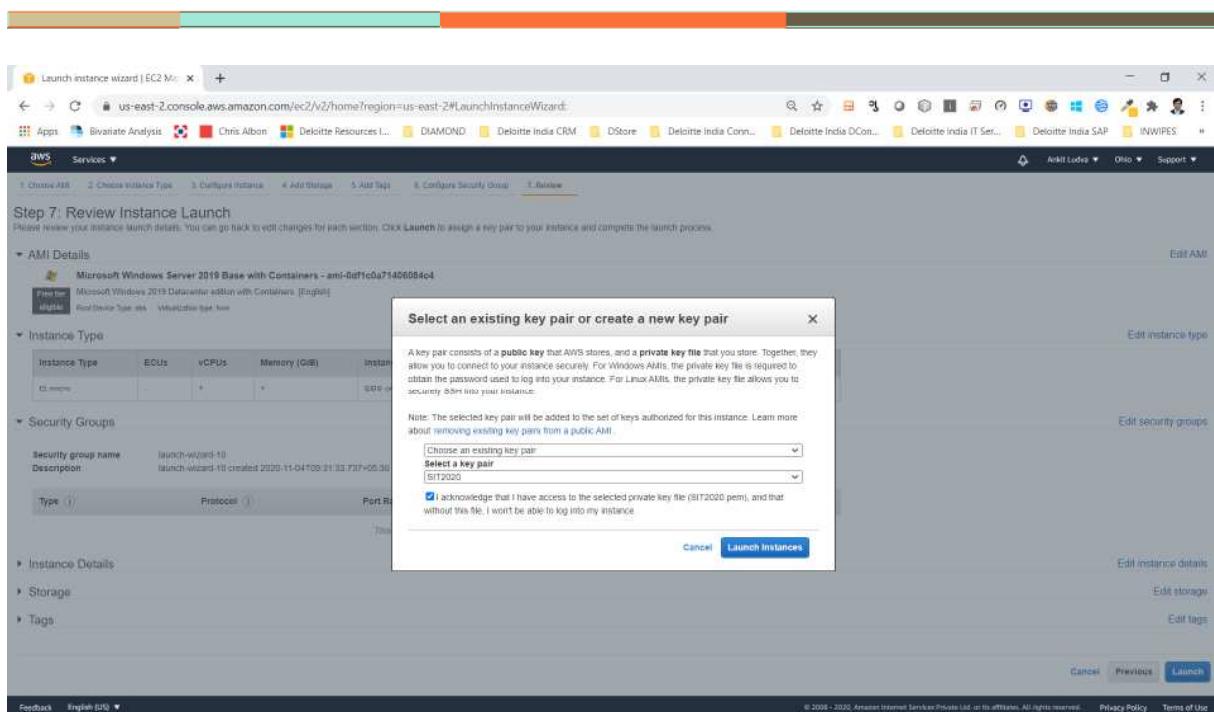
Instance Details

Storage

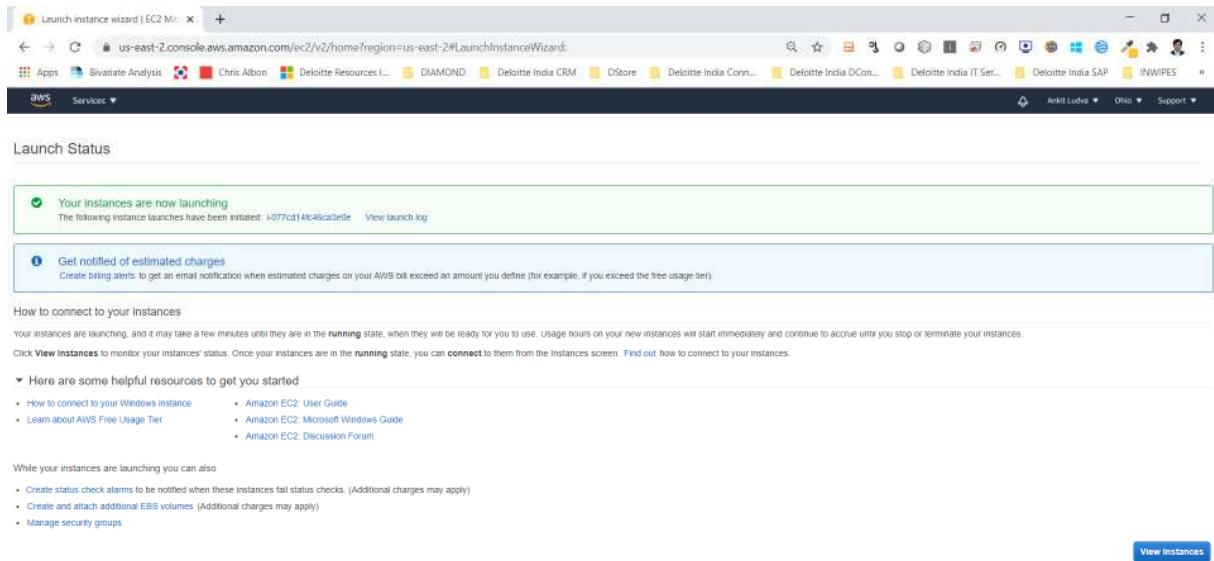
Tags

Cancel Previous Launch

7. Click on **Launch**



8. Create or acknowledge the key file. If you wish to existing key pair file then select Choose an existing key pair option from first drop down and select the name of existing key pair file from second dropdown and then check acknowledge.
9. After selecting key pair, click on Launch instances.



10. EC-2 windows instance is configured successfully. Click on View instances to view.

The screenshot shows the AWS EC2 Management Console. The left sidebar navigation includes: New EC2 Experience, EC2 Dashboard, Events, Tags, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Feedback, English (US), and links for Help, Privacy Policy, and Terms of Use.

The main content area displays the 'Instances (1) Info' table:

Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone	Public IPv4 DNS	Public IPv4 ...
	i-077cd14fc46ca3e0e	Running	t2.micro	initializing	No alarms	us-east-2c	ec2-18-188-76-161.us-east-2.co...	18.188.76.161

Below the table, a message says 'Select an instance above'.

11. We can see t2.micro instance is running. Select instance to view the IP and DNS to access instance.

The screenshot shows the AWS EC2 Management Console with the 'Instances (1/1) Info' table. The instance i-077cd14fc46ca3e0e is selected. The main content area displays the 'Instance: i-077cd14fc46ca3e0e' details:

Details Tab:

- Instance summary: Instance ID (i-077cd14fc46ca3e0e), Instance state (Running), Instance type (t2.micro), IAM Role (None).
- Public IPv4 address: 18.188.76.161 | open address
- Public IPv4 DNS: ec2-18-188-76-161.us-east-2.compute.amazonaws.com | open address
- Elastic IP addresses: None
- VPC ID: vpc-f22b8699
- Subnet ID: subnet-4d93a001
- Platform: Amazon Linux 2
- AMI ID: ami-0f11671a71a600fa4
- Monitoring: disabled

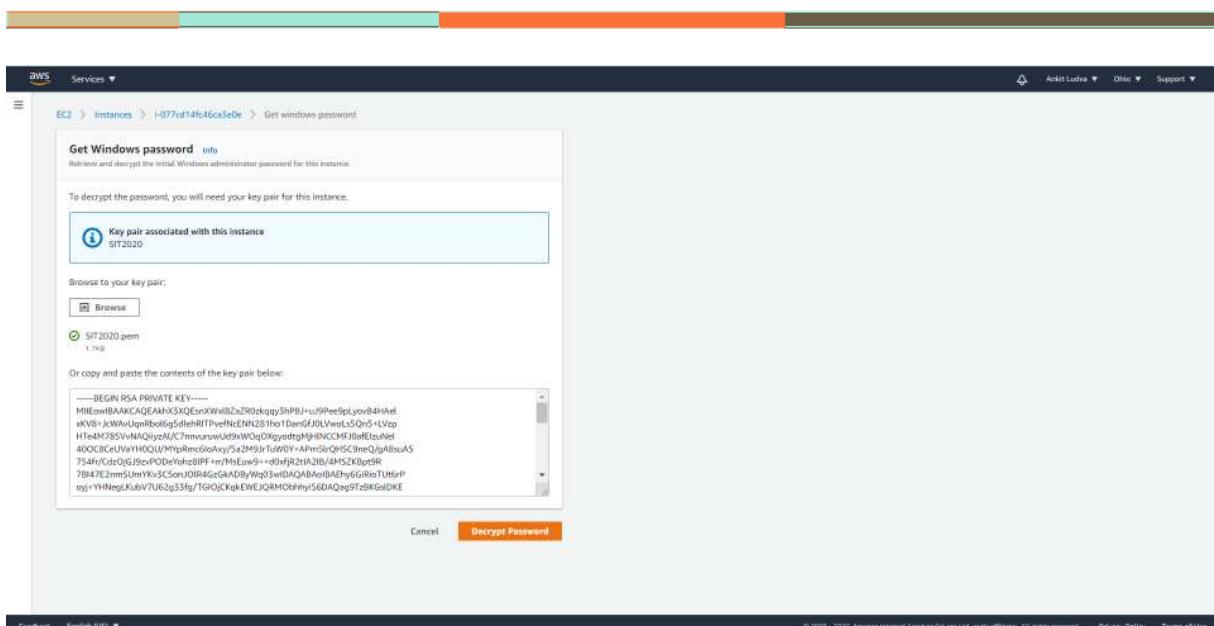
12. Access the Windows instance using Remote Desktop Connection. Remote Desktop Connection software required Public DNS/IP, username and password to access any instance.

13. To get windows instance password, select instance and click on Actions > Connect.

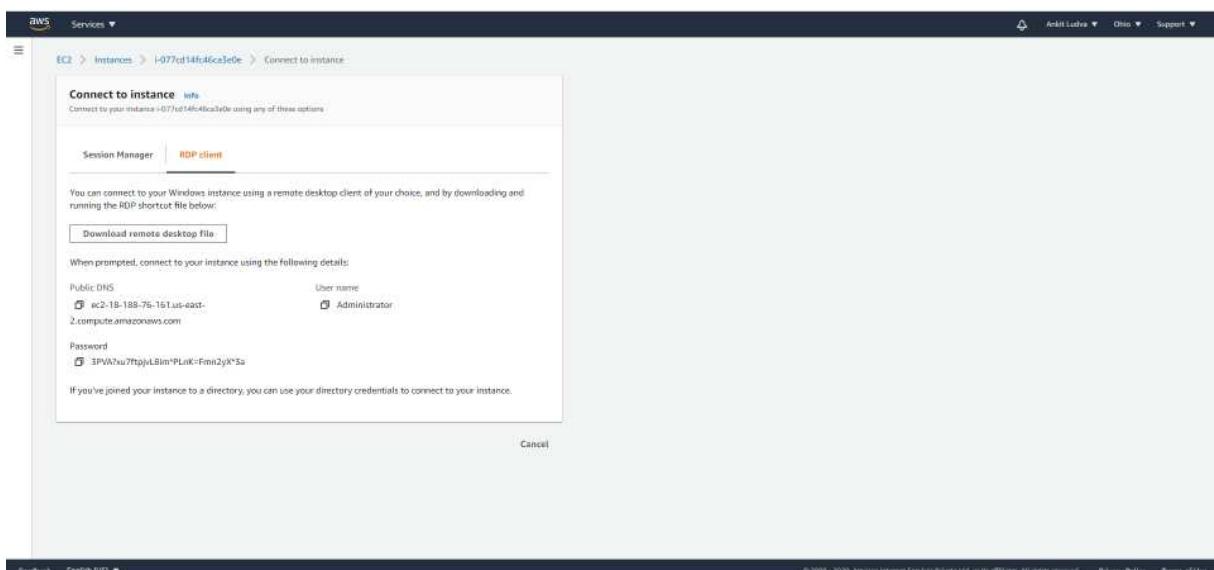
The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with various navigation options like EC2 Dashboard, Events, Tags, Instances, Images, Elastic Block Store, Network & Security, and more. The main area shows a table of instances with one row selected. The selected instance is 'i-077cd14fc46ca3e0e'. The 'Actions' menu is open, and the 'Connect' option is highlighted with a yellow box. Below the table, the instance details are shown in a modal window. The 'Details' tab is selected, displaying information such as Instance ID (i-077cd14fc46ca3e0e), Instance state (Running), Instance type (t2.micro), Public IPv4 address (18.188.76.161), Private IPv4 address (172.31.41.89), and VPC ID (vpc-f123c0699). Other tabs like Security, Networking, Storage, Status Checks, Monitoring, and Tags are also present.

This screenshot shows the 'Connect to instance' dialog box. At the top, it says 'EC2 > Instances > i-077cd14fc46ca3e0e > Connect to instance'. Below that, it says 'Connect to instance' and provides instructions: 'Connect to your instance i-077cd14fc46ca3e0e using any of these options'. Two tabs are shown: 'Session Manager' and 'RDP client', with 'RDP client' highlighted with a yellow box. The dialog then asks, 'You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:'. It includes a 'Download remote desktop file' button. Below this, it says 'When prompted, connect to your instance using the following details:'. It lists 'Public DNS' as 'ec2-18-188-76-161.us-east-2.compute.amazonaws.com', 'User name' as 'Administrator', and 'Password' as 'Get password'. A note at the bottom says, 'If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.' At the bottom right of the dialog is a 'Cancel' button.

14. Click on RDP Client, and click on Get password.



15. Select .pem file which is key pair file for instance, and click on **Decrypt Password**.

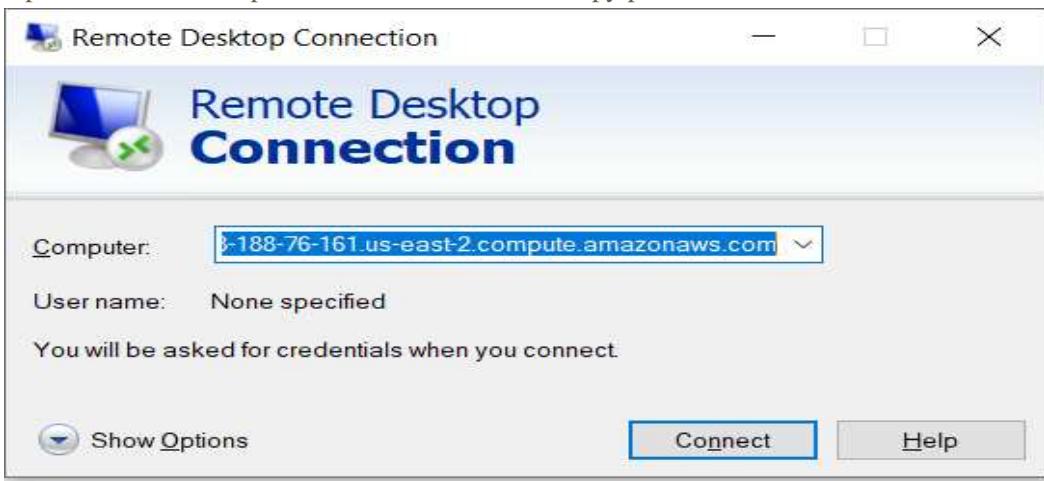


16. Decrypt password will enable windows passwords, using this credential we can able to access using Remote Desktop Connection software.

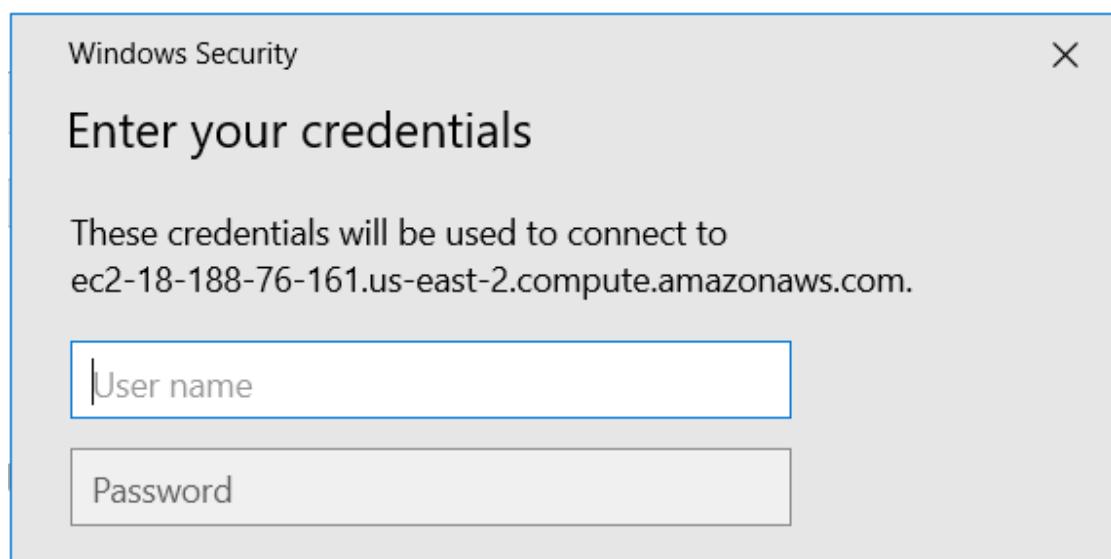
When prompted, connect to your instance using the following details:

Public DNS	User name
<input checked="" type="checkbox"/> ec2-18-188-76-161.us-east-2.compute.amazonaws.com	<input checked="" type="checkbox"/> Administrator
Password	
<input checked="" type="checkbox"/> 3PVA?xu7ftpjvL8im+PLnK=Fmn2yX*3a	

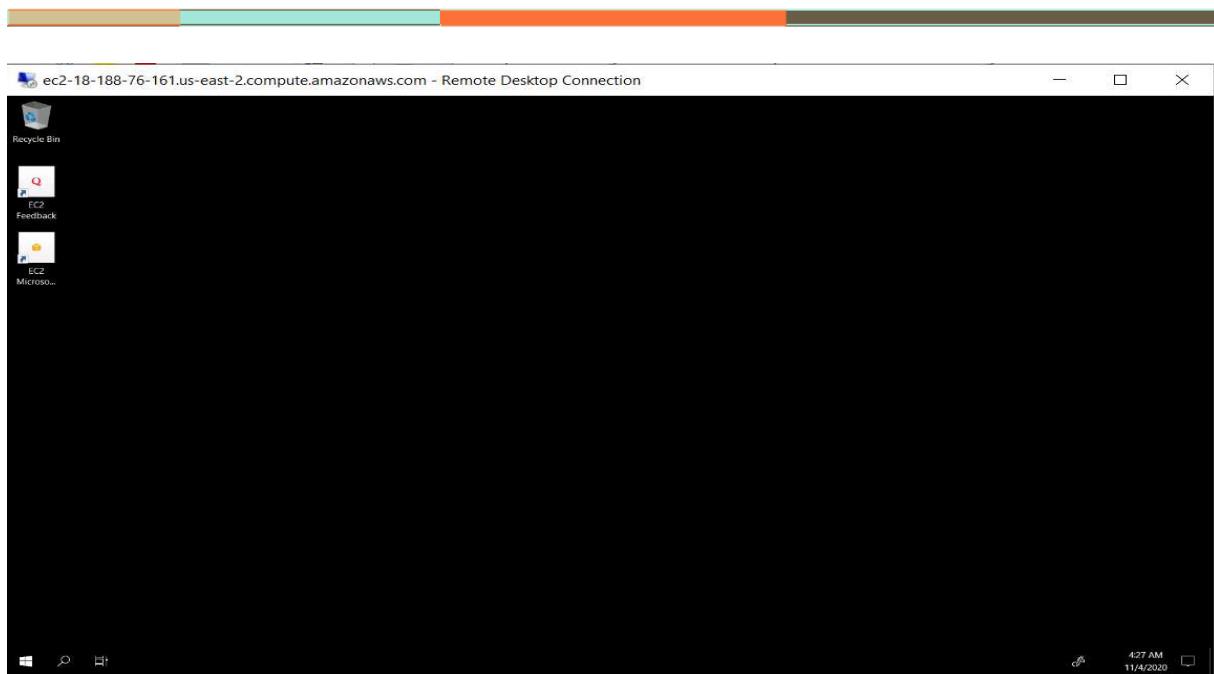
17. Open Remote Desktop Connection software and copy paste credential.



18. Copy public DNS and click on Connect

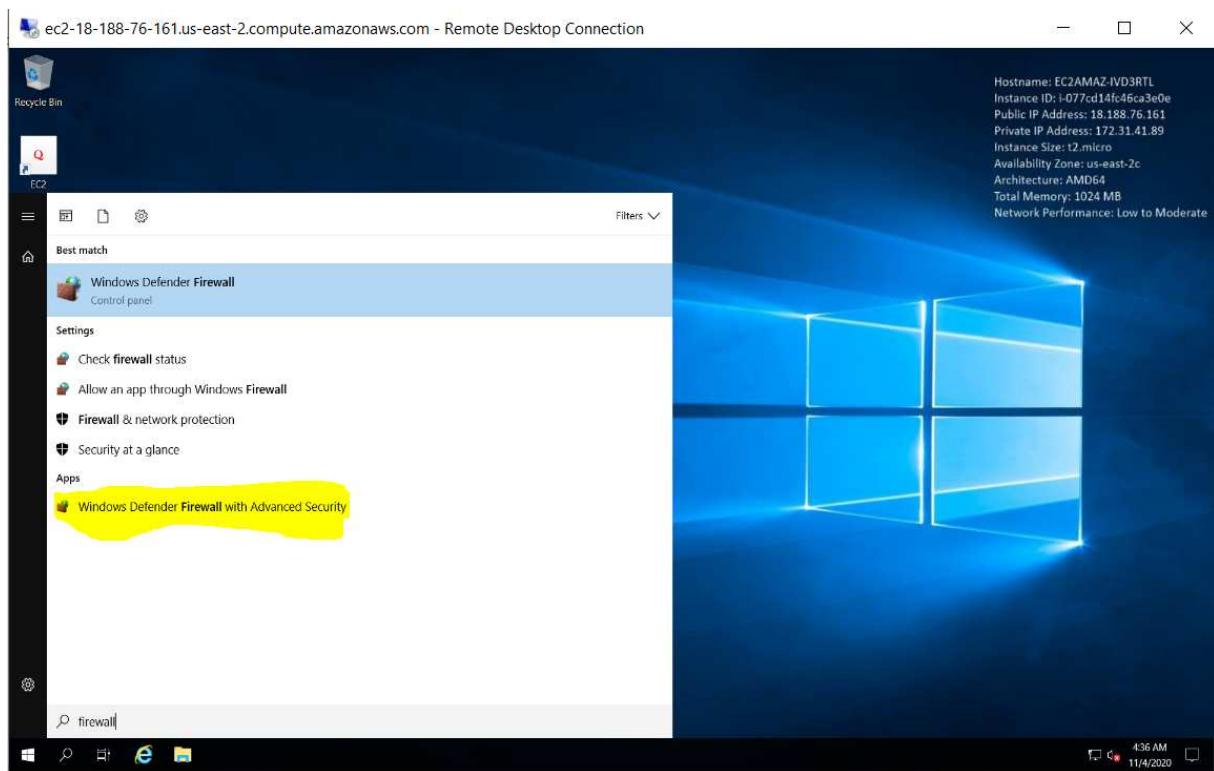


19. Enter username and password, click on connect to login into Windows EC2.

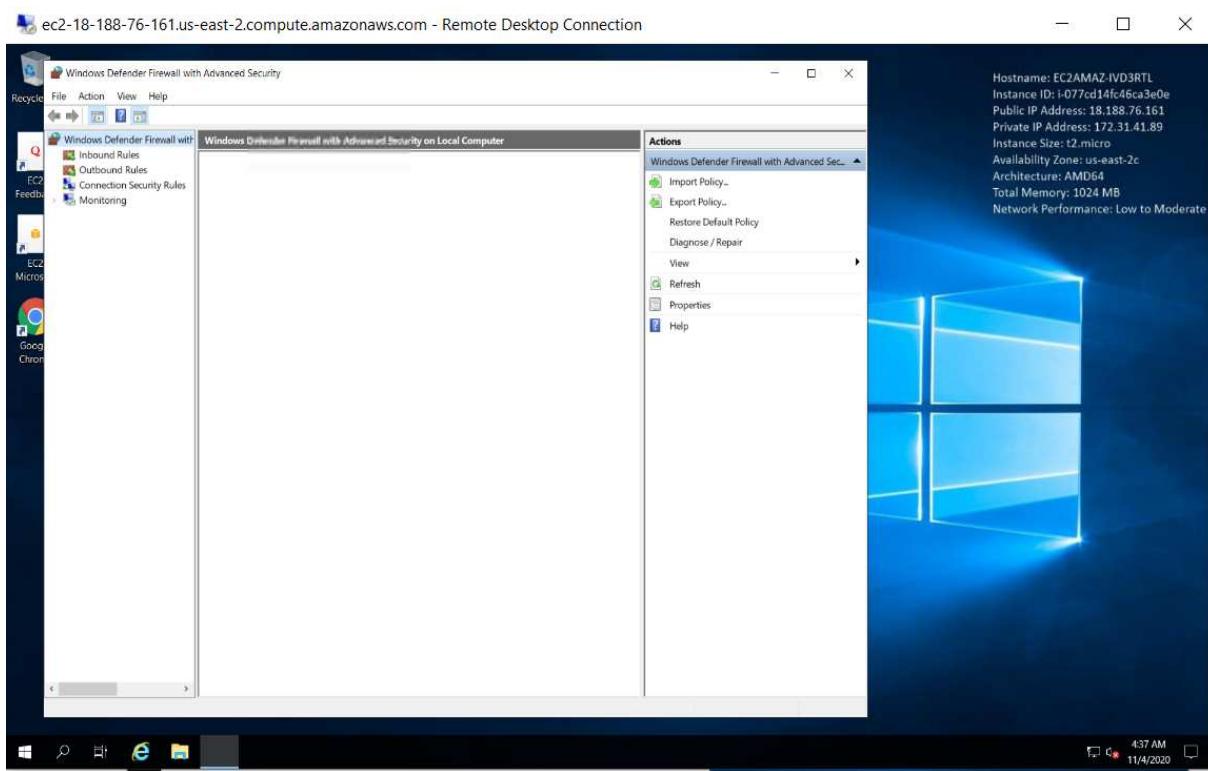


Successfully connected Windows EC2 using Remote Desktop Connection software.

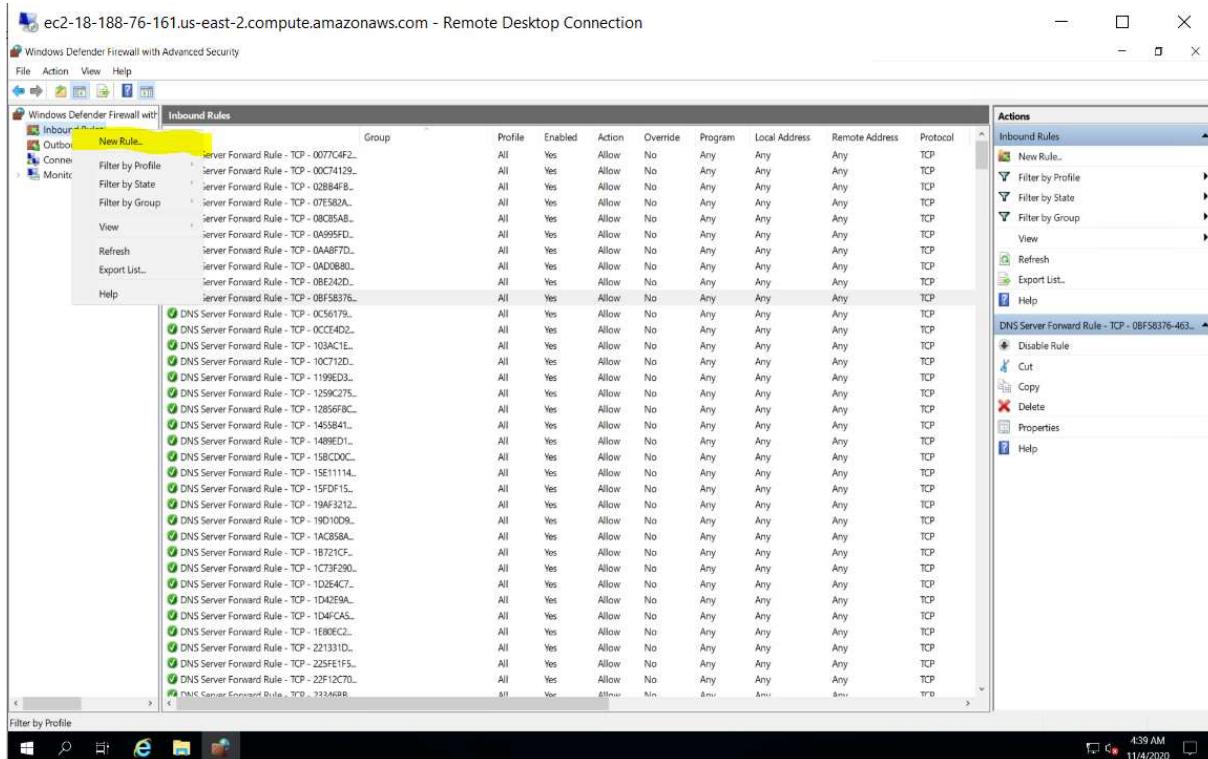
20. Configure Port 80 on windows instance, open Windows Defender Firewall with advance Security features.



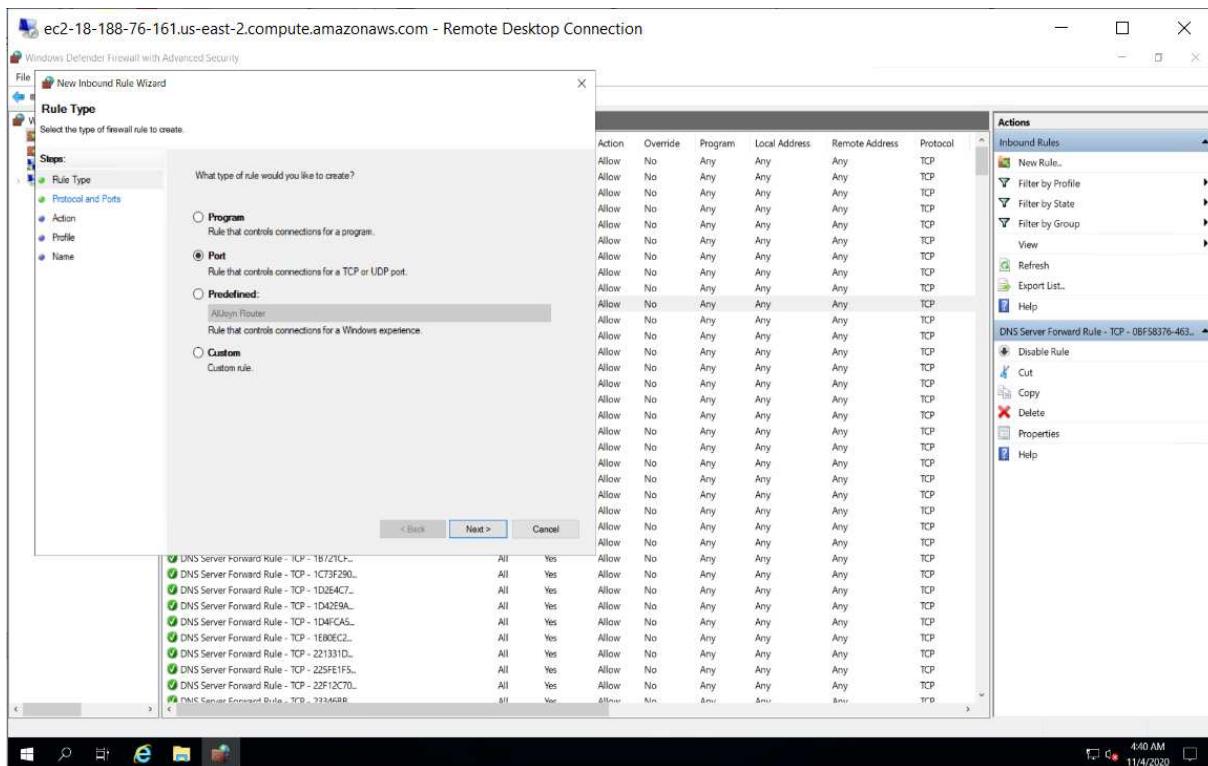
21. Select Inbound Rules tab from screen



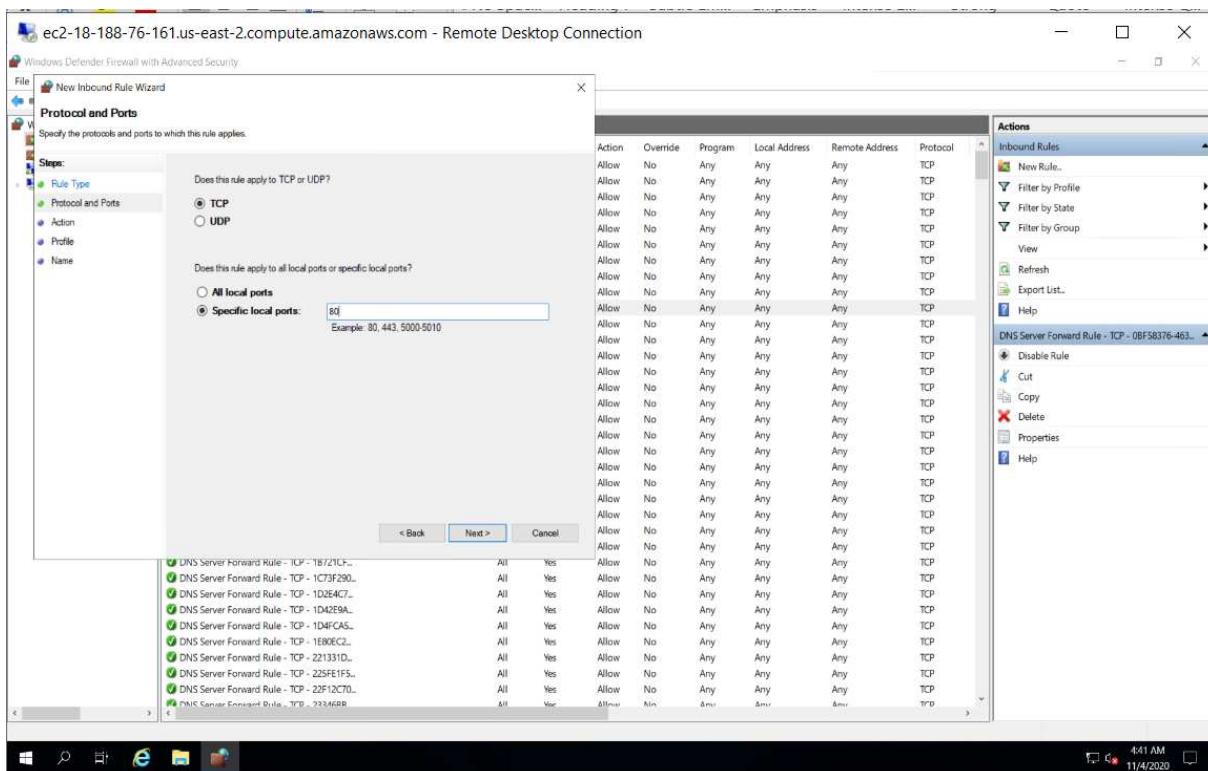
22. Right click on Inbound Rules > New Rule...



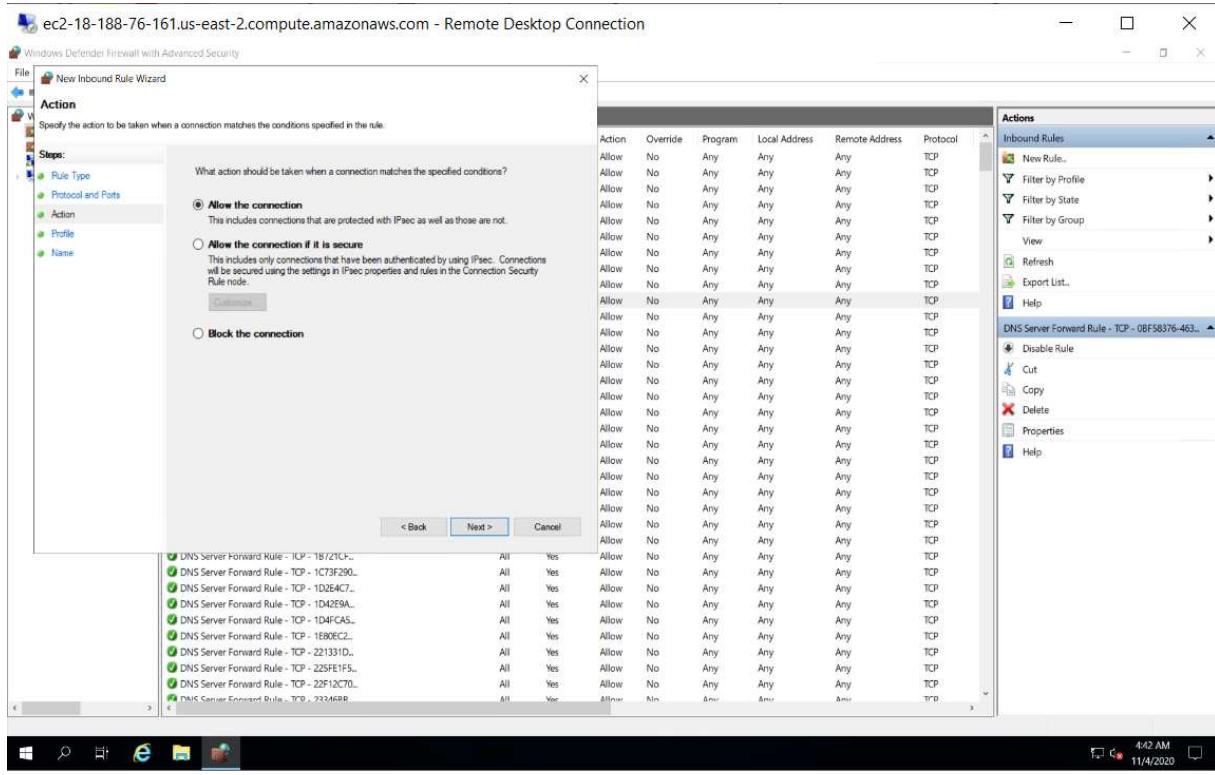
23. Select Port and click on Next



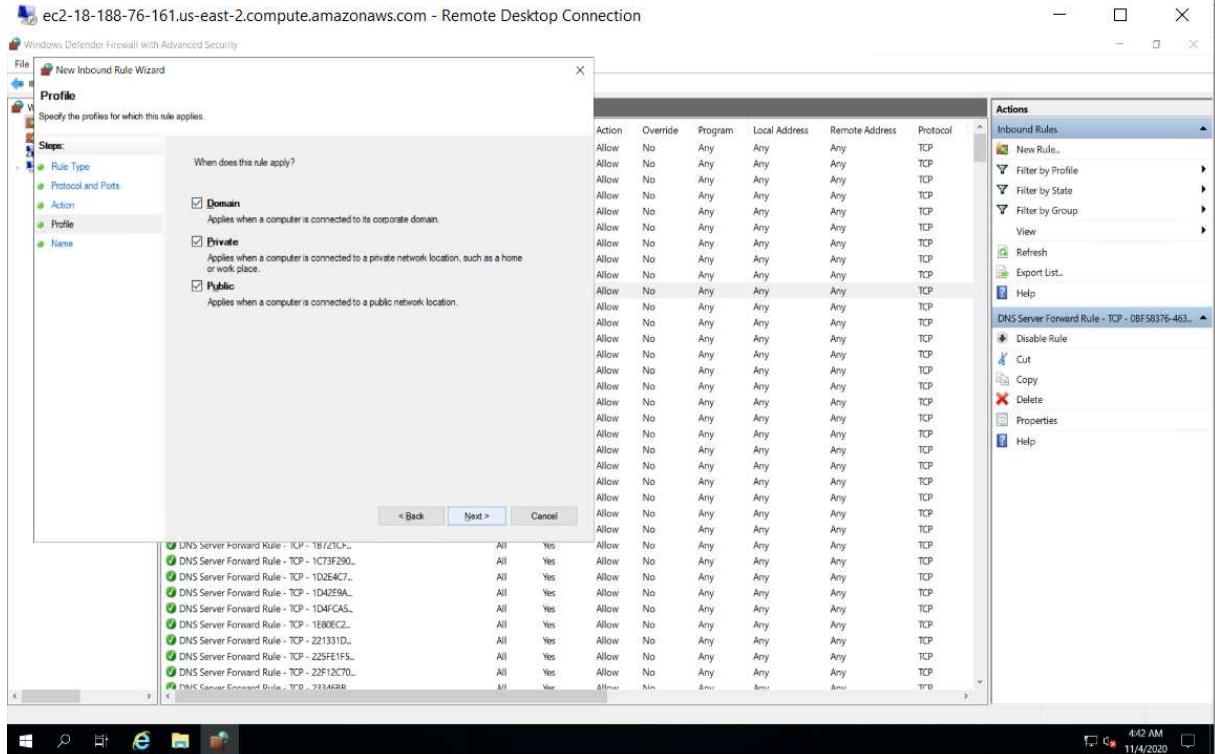
24. Type port number 80 with TCP selection and click on Next



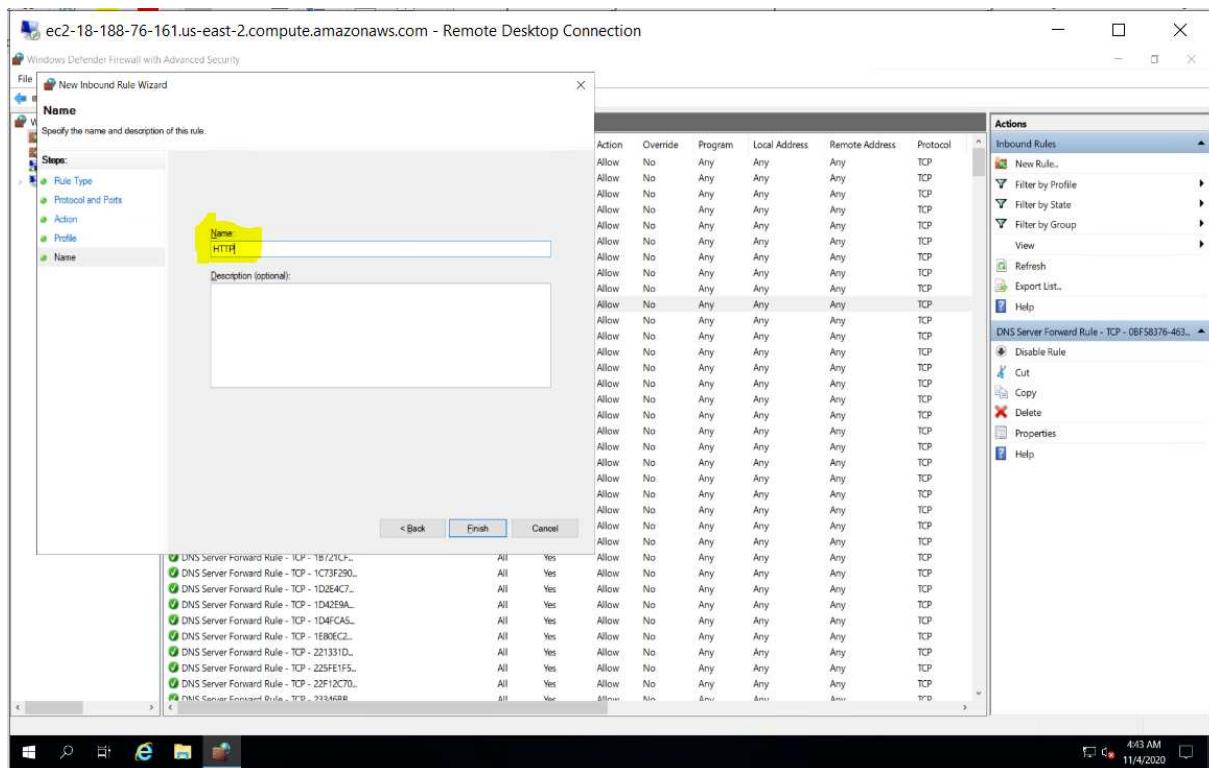
25. Click on Next



26. Click on Next

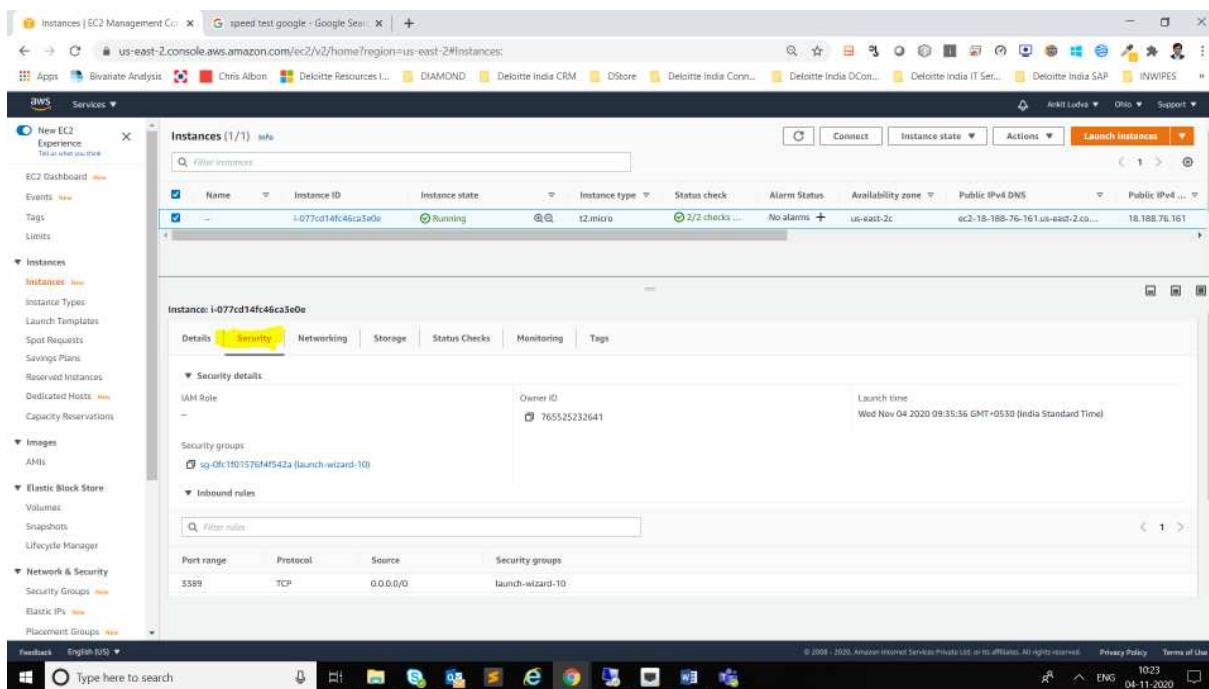


27. Enter Rules name HTTP , click on Finish.



28. Port 80 successfully configured on Windows Firewall configuration.

29. Next Step to configure port 80 on AWS console, select instance and select Security tab.



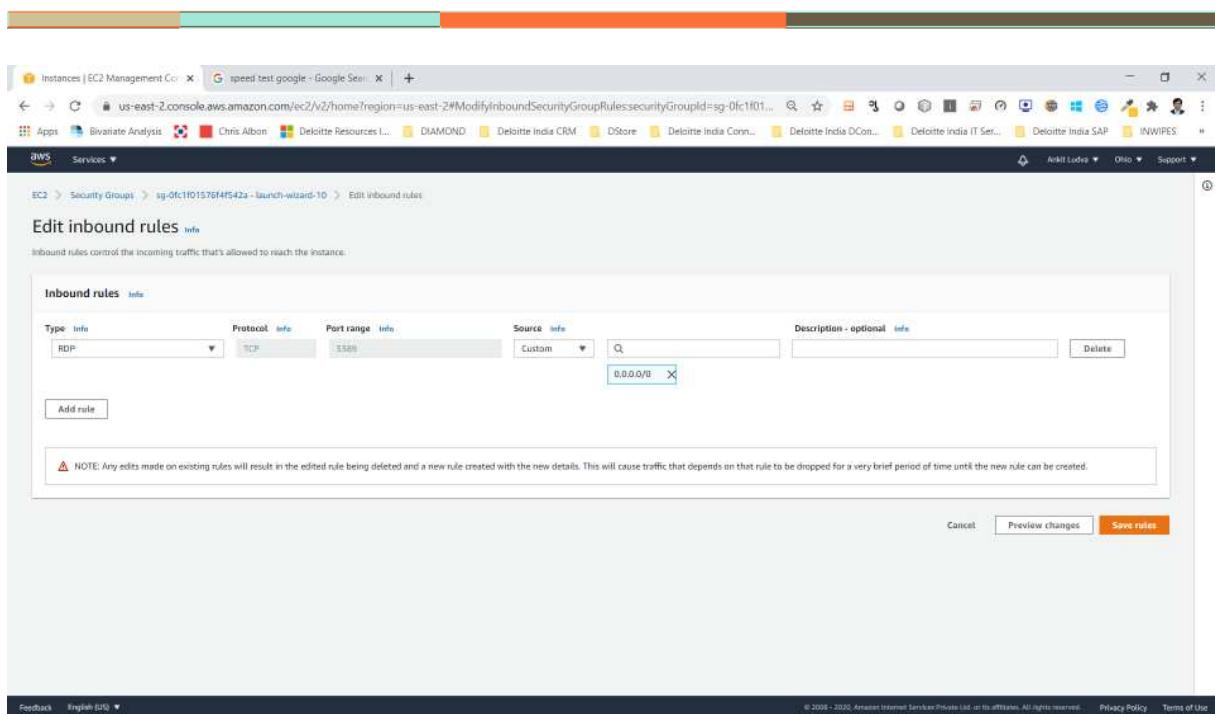
30. Select associate security group

The screenshot shows the AWS EC2 Instances page. A single instance, 'i-077cd14fc66ca3e00', is listed as 'Running'. The 'Security' tab is selected under the instance details. It shows the security group 'sg-0fc1f01576f4f542a - launch-wizard-10' is associated with the instance.

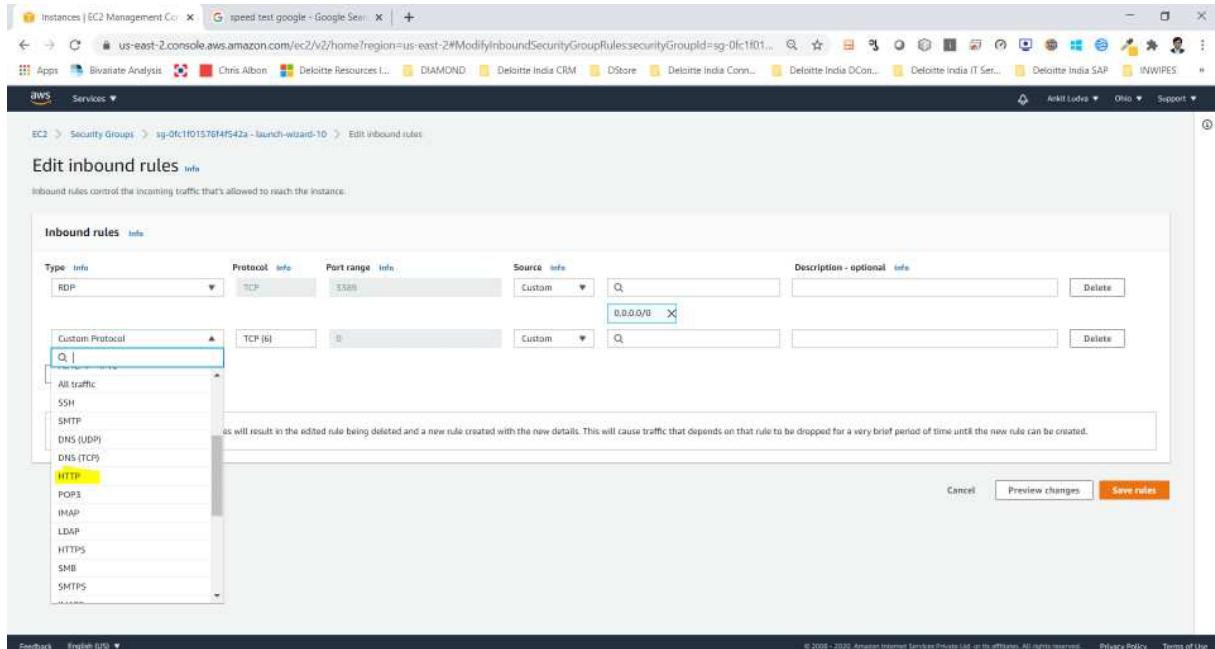
31. Click on Edit inbound rules.

The screenshot shows the AWS Security Groups page for the security group 'sg-0fc1f01576f4f542a - launch-wizard-10'. The 'Inbound rules' tab is selected. One inbound rule is listed: Type: RDP, Protocol: TCP, Port range: 3389, Source: 0.0.0.0/0. A yellow box highlights the 'Edit inbound rules' button.

32. Click on Add rule



33. Select HTTP from the drop down



34. Select Anywhere from Source drop down

The screenshot shows the AWS EC2 Management Console with the URL <https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#ModifyInboundSecurityGroupRulessecurityGroupId=sg-0fc101576f4f542a-launch-wizard-10>Edit inbound rules>. The page title is "Edit inbound rules". The main content area shows an "Inbound rules" table with three rows:

Type	Protocol	Port range	Source	Description - optional
RDP	TCP	3389	Custom	0.0.0.0/0
HTTP	TCP	80	Custom	0.0.0.0/0

A fourth row is being added, with the "Type" set to "RDP", "Protocol" to "TCP", "Port range" to "3389", and "Source" set to "Custom". The "Description - optional" field contains "My IP". A note at the bottom states: "⚠ NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created." At the bottom right are "Cancel", "Preview changes", and a large orange "Save rules" button.

35. Click on save rules.

Congratulation!!!! HTTP port configured successfully from AWS console and Windows firewall configuration.

Setup MySQL on Amazon RDS.

What is Amazon RDS?

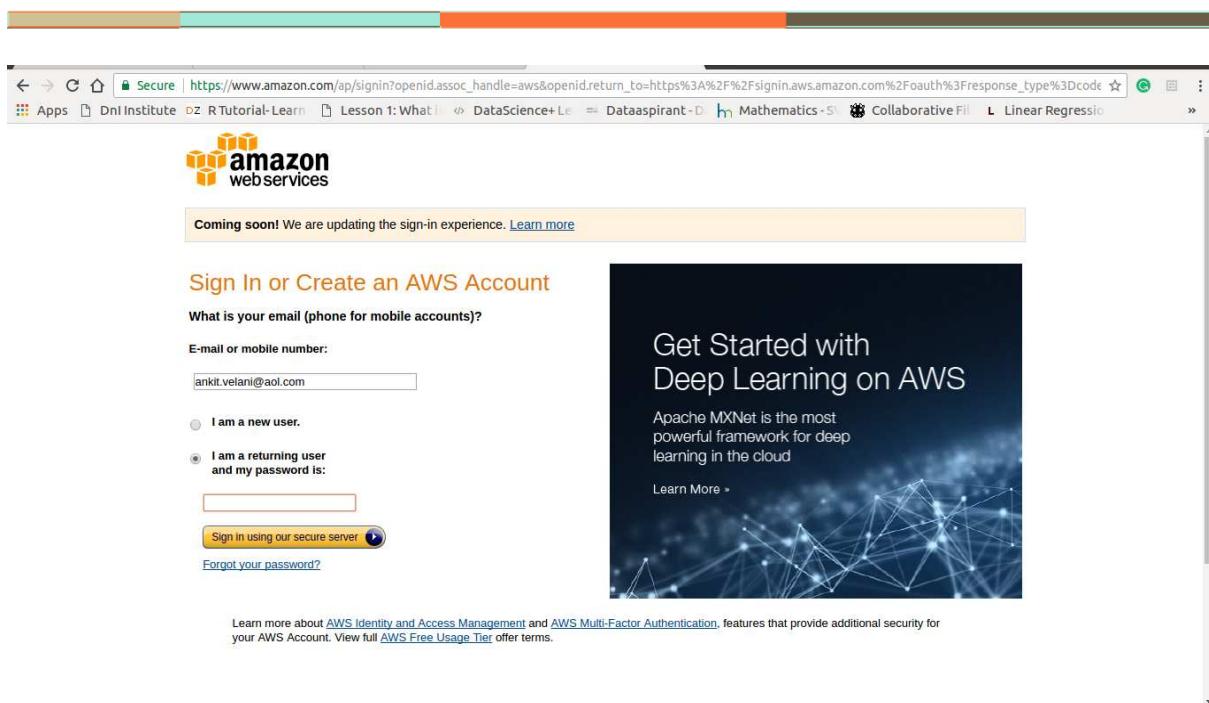
Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

Why would you want a managed relational database service? Because Amazon RDS takes over many of the difficult or tedious management tasks of a relational database.

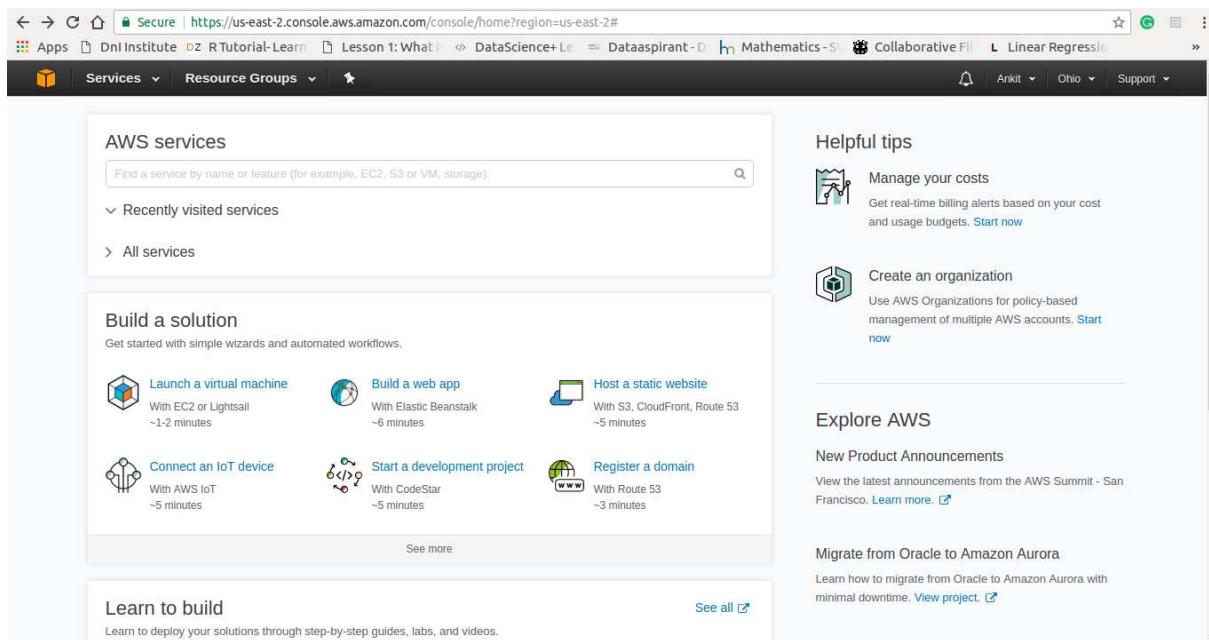
- When you buy a server, you get CPU, memory, storage, and IOPS, all bundled together. With Amazon RDS, these are split apart so that you can scale them independently. So, for example, if you need more CPU, less IOPS, or more storage, you can easily allocate them.
- Amazon RDS manages backups, software patching, automatic failure detection, and recovery.
- In order to deliver a managed service experience, Amazon RDS does not provide shell access to DB instances, and it restricts access to certain system procedures and tables that require advanced privileges.
- You can have automated backups performed when you need them, or create your own backup snapshot. These backups can be used to restore a database, and the Amazon RDS restore process works reliably and efficiently.
- You can get high availability with a primary instance and a synchronous secondary instance that you can failover to when problems occur. You can also use MySQL, MariaDB, or PostgreSQL Read Replicas to increase read scaling.
- You can use the database products you are already familiar with: MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server, and the new, MySQL-compatible Amazon Aurora DB engine (for information, see Aurora on Amazon RDS).
- In addition to the security in your database package, you can help control who can access your RDS databases by using AWS IAM to define users and permissions. You can also help protect your databases by putting them in a virtual private cloud.

Setup MySQL on Amazon RDS.

1. Login to Amazon cloud portal. <https://console.aws.amazon.com>
2. Enter registered email and choose option ‘I am returning user and my password is: ‘.
3. Click on Sign in using our secure server.



4. After successfully login, Console landing page will appear.



5. Click on All Services ,

The screenshot shows the AWS Management Console homepage. The left sidebar lists various AWS service categories: Compute (EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, Batch), Storage (S3, EFS, Glacier, Storage Gateway), and Database (RDS). The main content area displays a grid of service icons and names under categories like Developer Tools, Internet of Things, Contact Center, Management Tools, Game Development, Mobile Services, and more. To the right, there's a "Helpful tips" section with links to "Manage your costs", "Create an organization", and "Explore AWS".

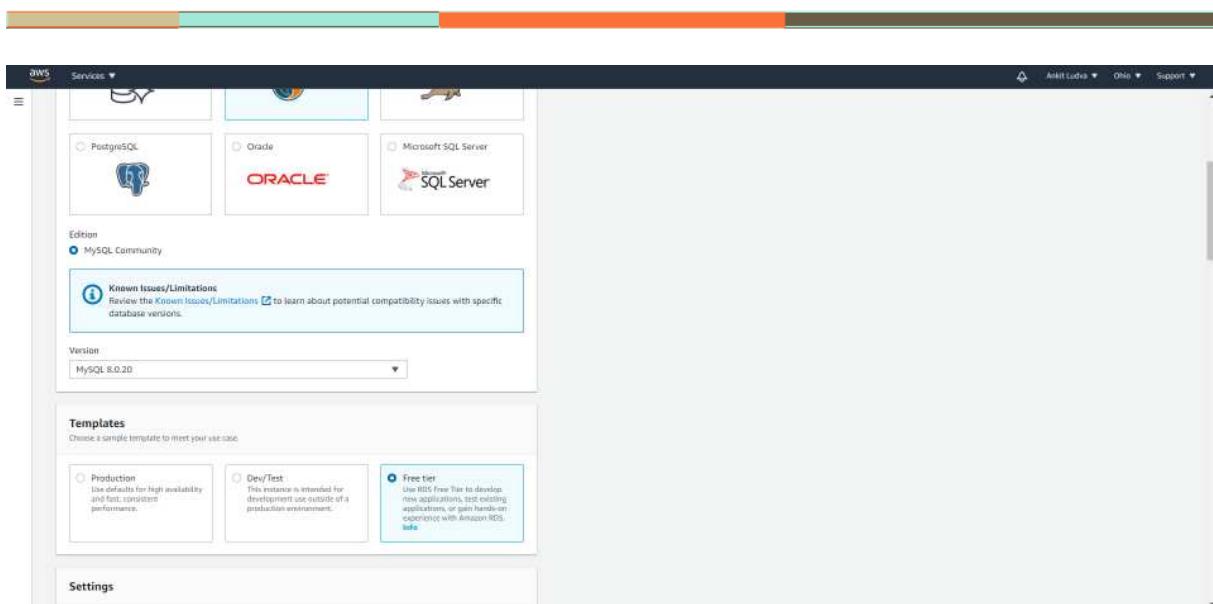
6. Choose RDS, from category “Database”

This screenshot shows the "Database" category selected in the AWS services navigation pane. It includes services like RDS, DynamoDB, ElastiCache, and Redshift. The right side of the screen displays promotional cards for various AWS services: Amazon GameLift, AWS Lambda, Amazon DynamoDB, AWS Marketplace, and a feedback section.

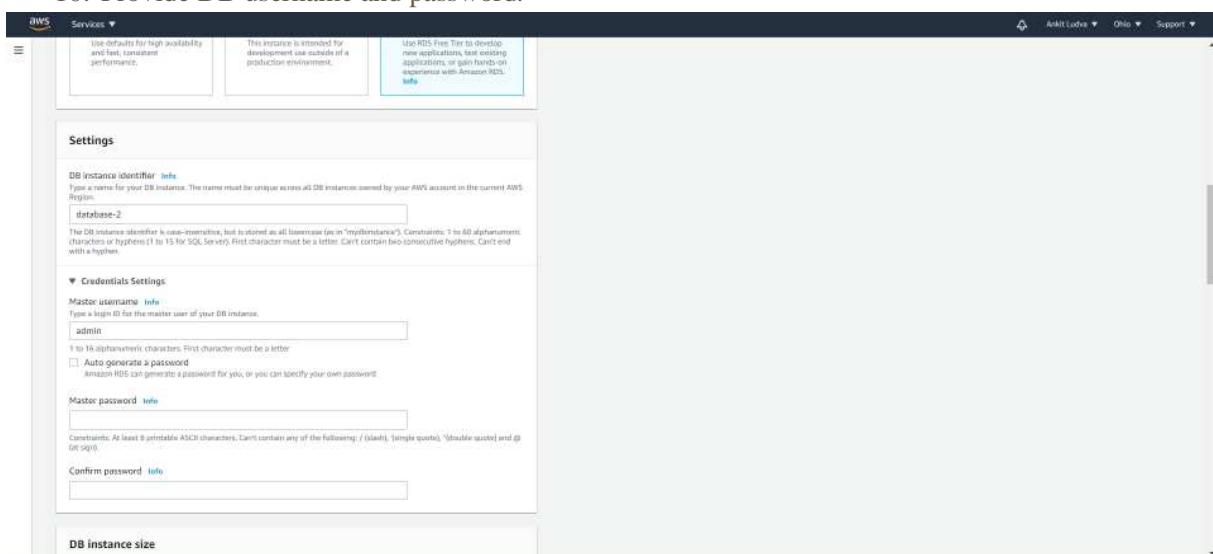
7. Click on RDS.

8. Click on Create database

9. Click on Free tier eligible only , Click on MySQL



10. Provide DB username and password.



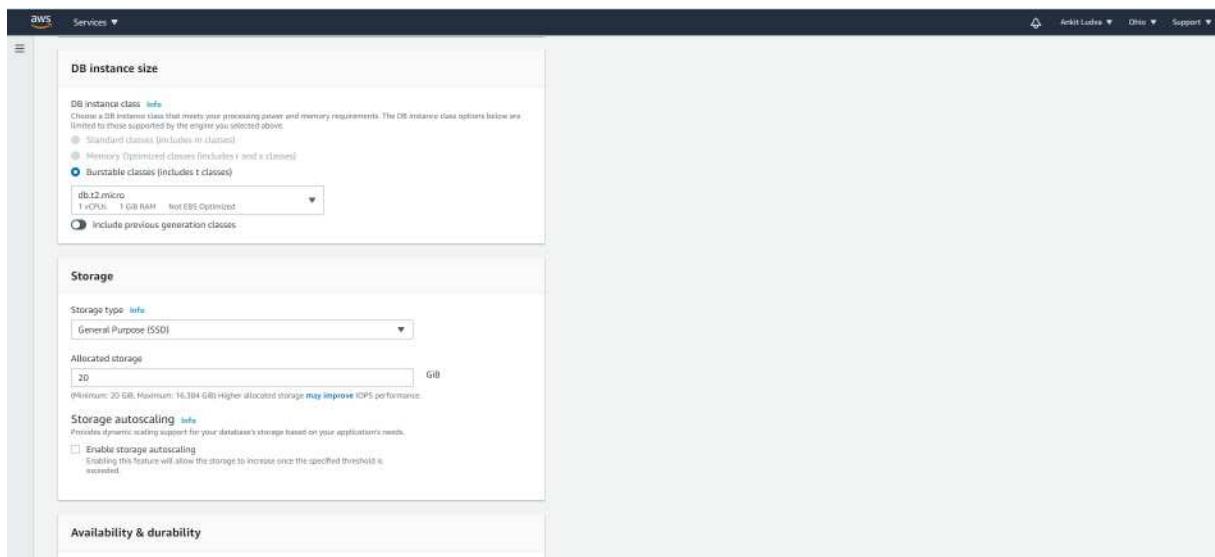
DBInstance Identifier :=====> Database Instance name: **dbserver**

Master Username :=====> root

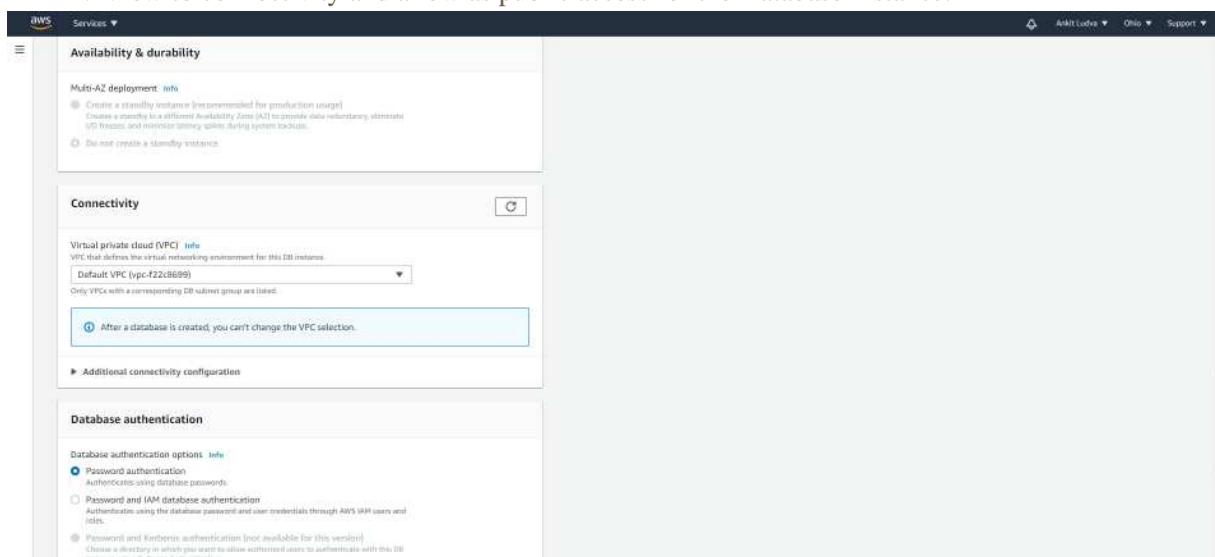
Master Password :=====> mypassword

Confirm Password :=====> mypassword

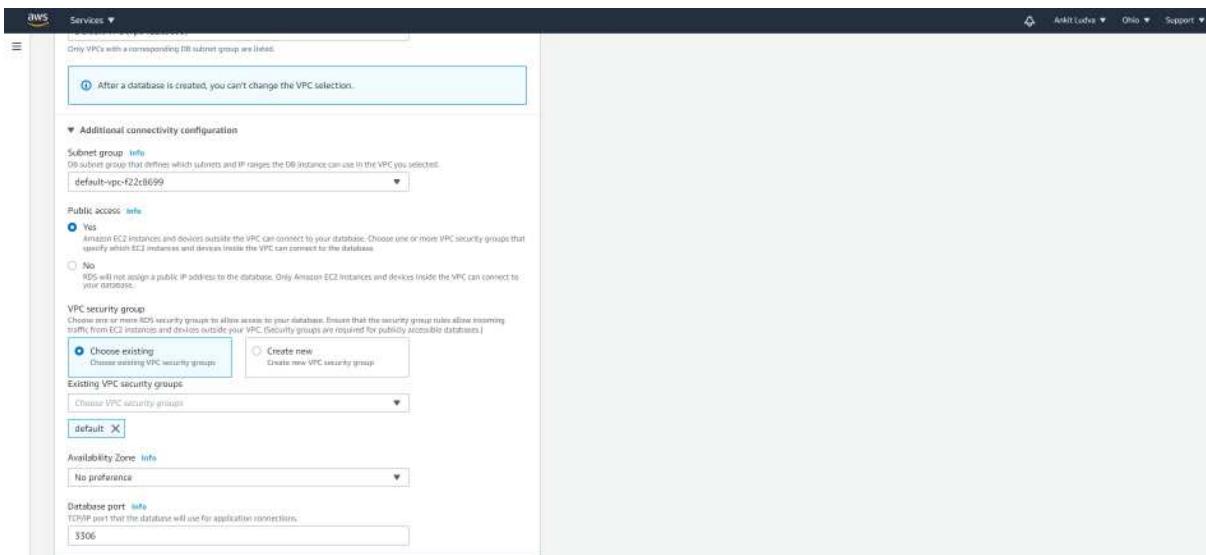
11. Free tier Instance configuration



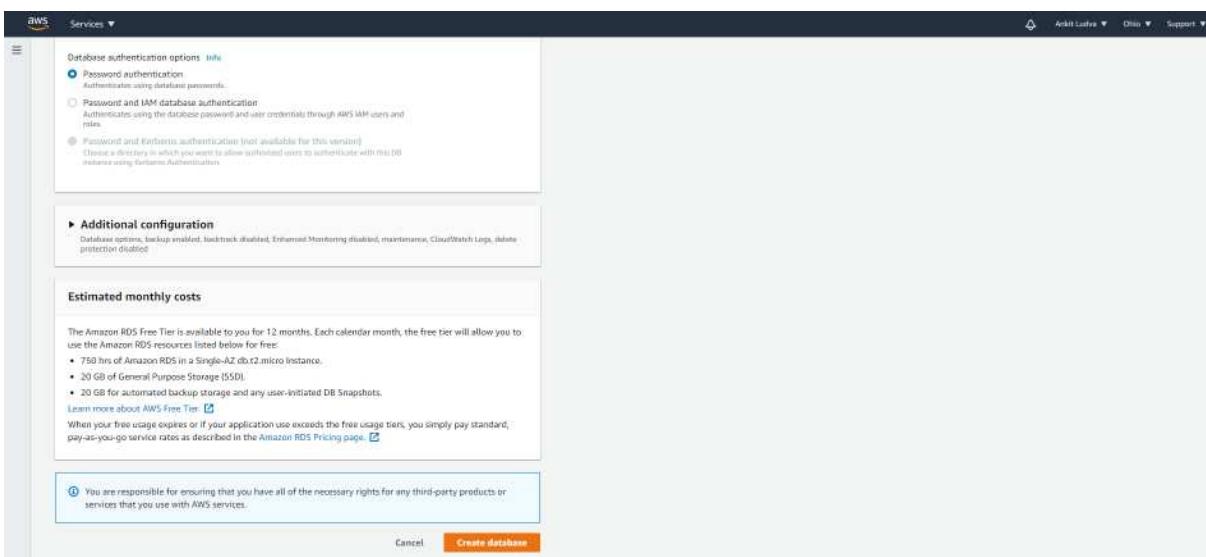
12. View to connectivity and allow as public access for the Database instance.



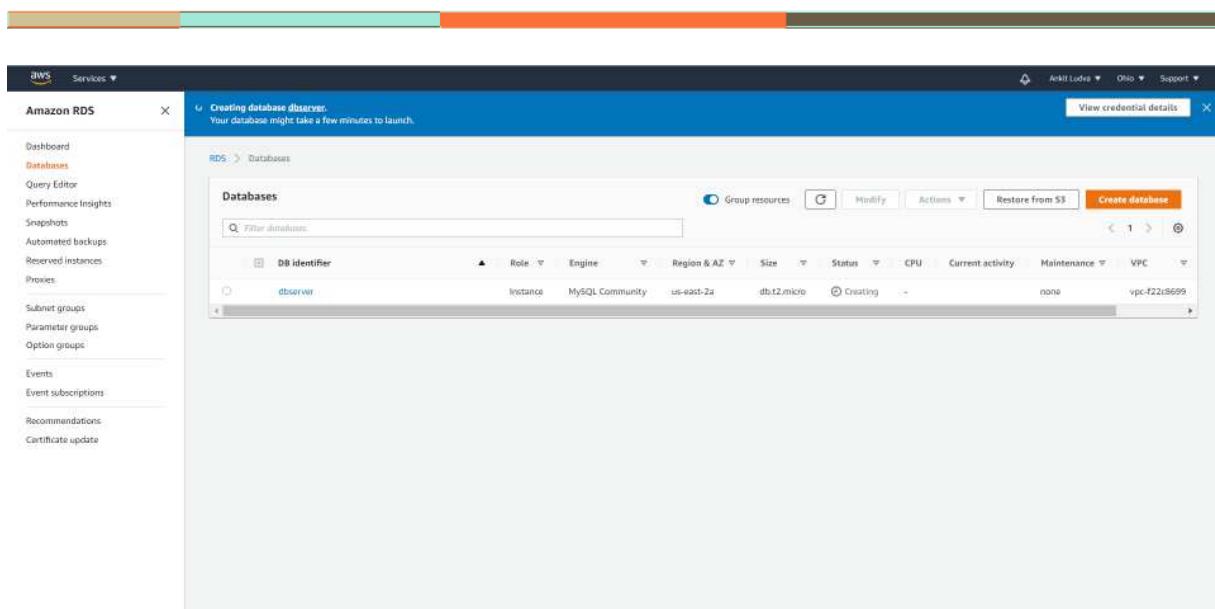
13. Click on Additional connectivity configuration, and select public access YES.



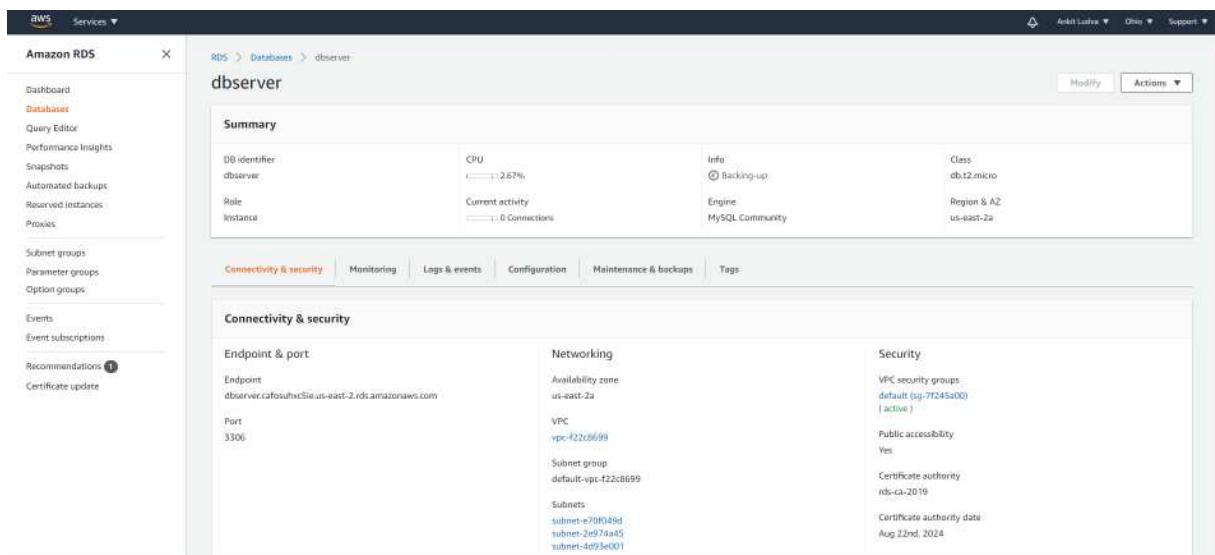
14. Scroll down and Click on Create database



15. After few minutes MySQL RDS instance will be available for use.



16. Once it available, select DB instance to view the endpoints and other configurations.



Congratulation :) Successfully created MySQL Database on Amazon RDS.

Working with RDS MySQL

1. Jump to RDS dashboard.

The screenshot shows the AWS RDS MySQL dashboard. On the left, a sidebar lists options like Dashboard, Instances, Clusters, Performance Insights, Snapshots, Reserved instances, Subnet groups, Parameter groups, Option groups, Events, Event subscriptions, and Recommendations. The main area displays Amazon Aurora information, a 'Create database' button, and a 'Resources' section showing DB Instances (1/40), Allocated storage (20.00 GB/100.00 TB), Reserved instances (0/40), Snapshots (33), and Event subscriptions (0/20). To the right, there's an 'Additional information' panel with links to various RDS resources and a 'Database Preview Environment' section.

2. Click on DB Instances.

The screenshot shows the AWS RDS Instances page. The sidebar is identical to the previous dashboard. The main area shows a table titled 'Instances (1)'. It contains one row for 'db-server', which is listed under the 'DB instance' column. Other columns include Engine (MySQL), Status (available), Current activity (1.19%), Maintenance (none), Class (db.t2.micro), and VPC (vpc-f9e9da9). There are buttons for 'Instance actions', 'Restore from S3', and 'Create database'.

Summary

DB identifier	CPU	Info	Class
dbserver	2.67%	Backing-up	db.t2.micro
Role	Current activity	Engine	Region & AZ
Instance	0 Connections	MySQL Community	us-east-2a

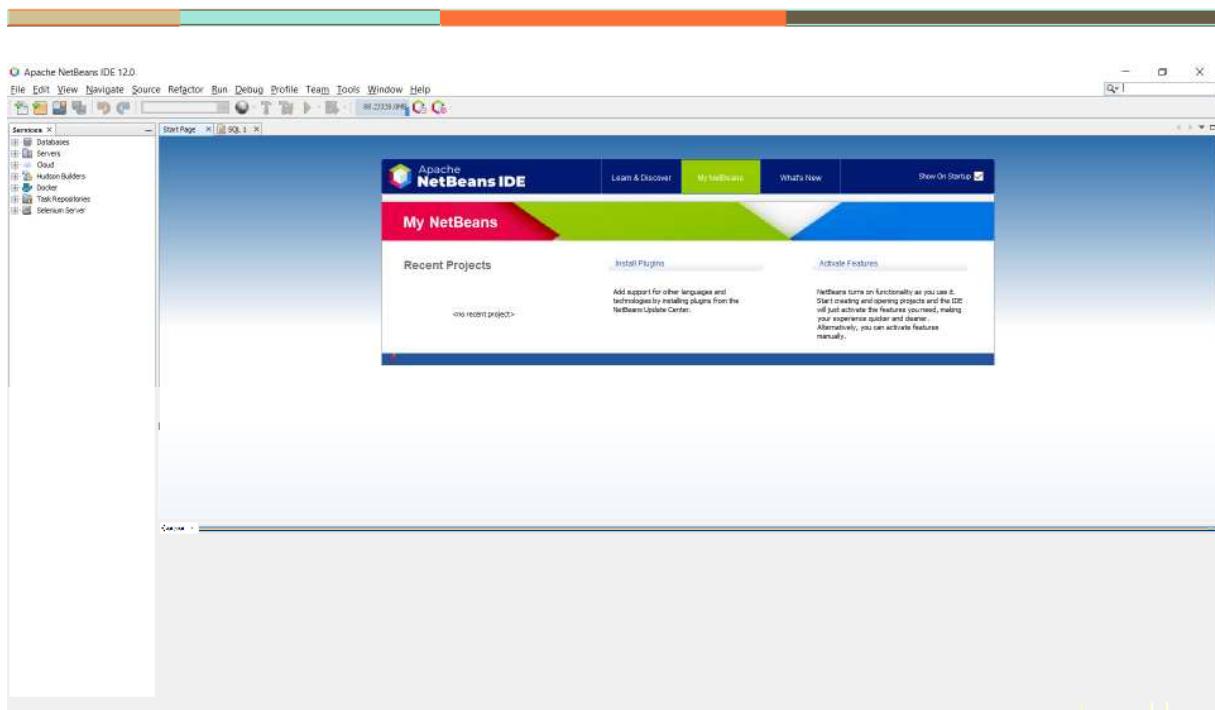
Connectivity & security

Endpoint & port	Networking	Security
Endpoint dbserver.cafoсхc5ie.us-east-2.rds.amazonaws.com	Availability zone us-east-2a	VPC security groups default (sg-7f245a00) (active)
Port 3306	VPC vpc-f22c8699	Public accessibility Yes
	Subnet group default-vpc-f22c8699	Certificate authority rds-ca-2019
	Subnets subnet-e70f049d subnet-2e974a45 subnet-4d93e001	Certificate authority date Aug 22nd, 2024

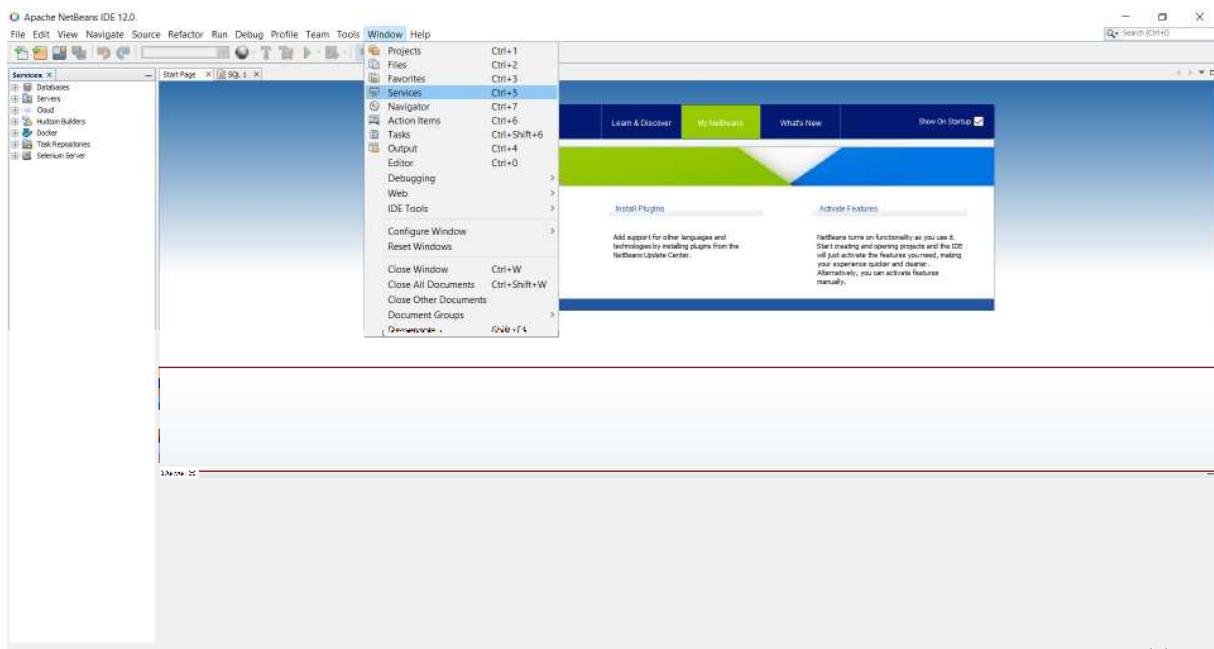
3. Endpoint of the DB Instances help us to access MySQL.

Endpoint & port	Networking	Security
Endpoint dbserver.cafoсхc5ie.us-east-2.rds.amazonaws.com	Availability zone us-east-2a	VPC security groups default (sg-7f245a00) (active)
Port 3306	VPC vpc-f22c8699	Public accessibility Yes
	Subnet group default-vpc-f22c8699	Certificate authority rds-ca-2019
	Subnets subnet-e70f049d subnet-2e974a45 subnet-4d93e001	Certificate authority date Aug 22nd, 2024

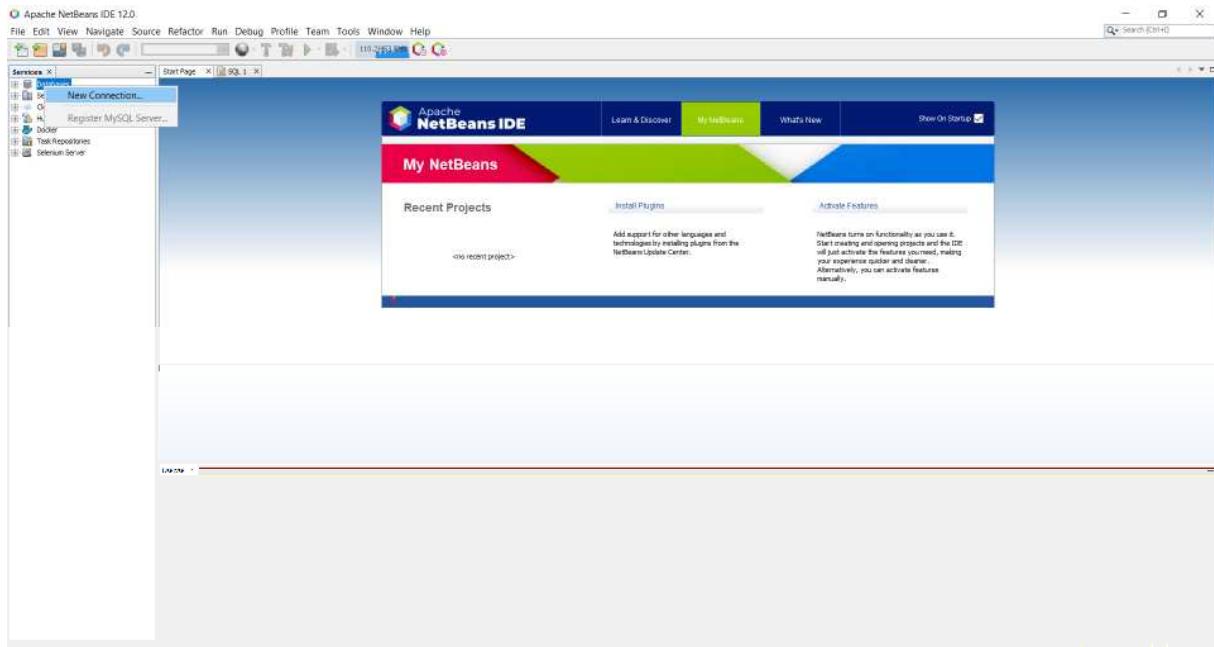
4. We can use this Endpoint to connect with MySQL Database in any programming language or tools like MySQL Workbench, SqlWorkbenchJ, in NetBeans or Eclipse..Etc.
5. We will try to connect with NetBeans. Open **NetBeans**.



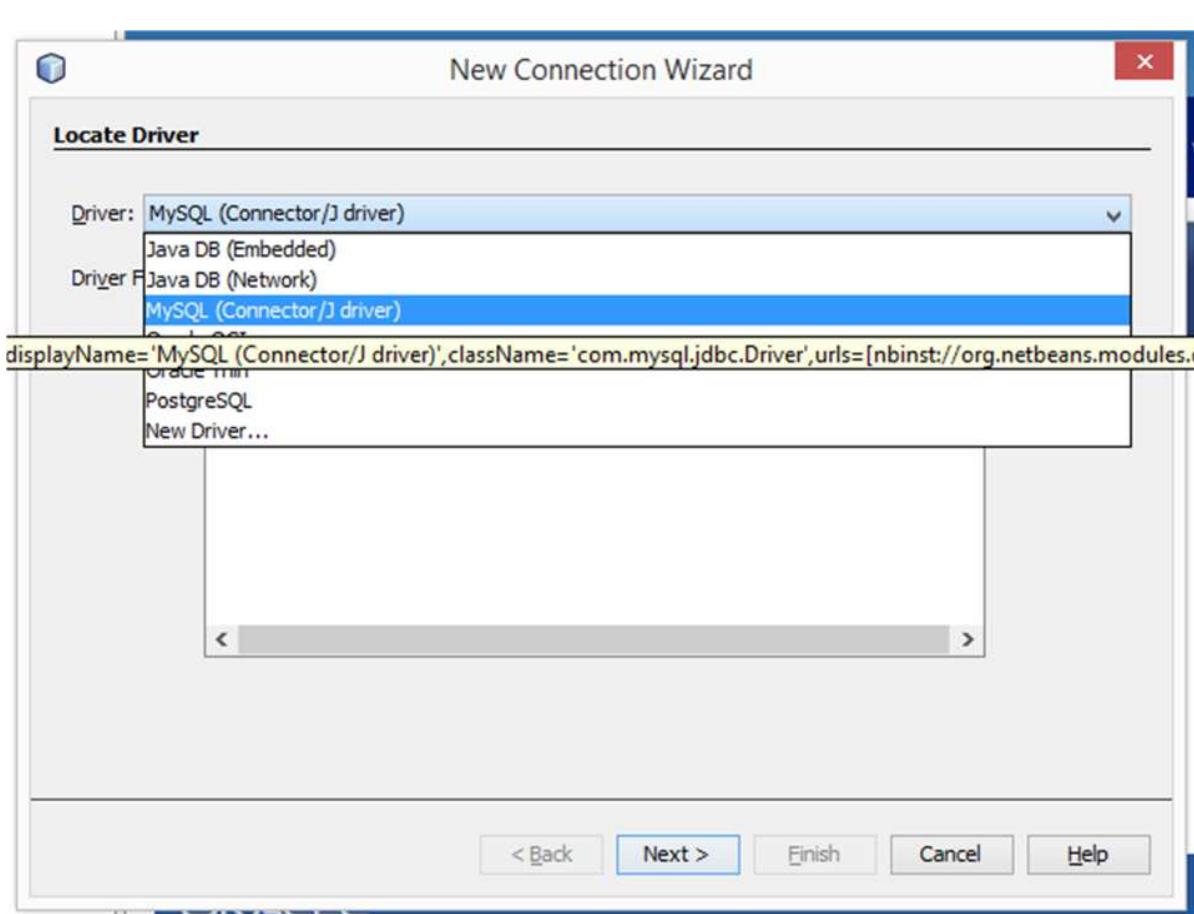
6. Go to Windows > Services



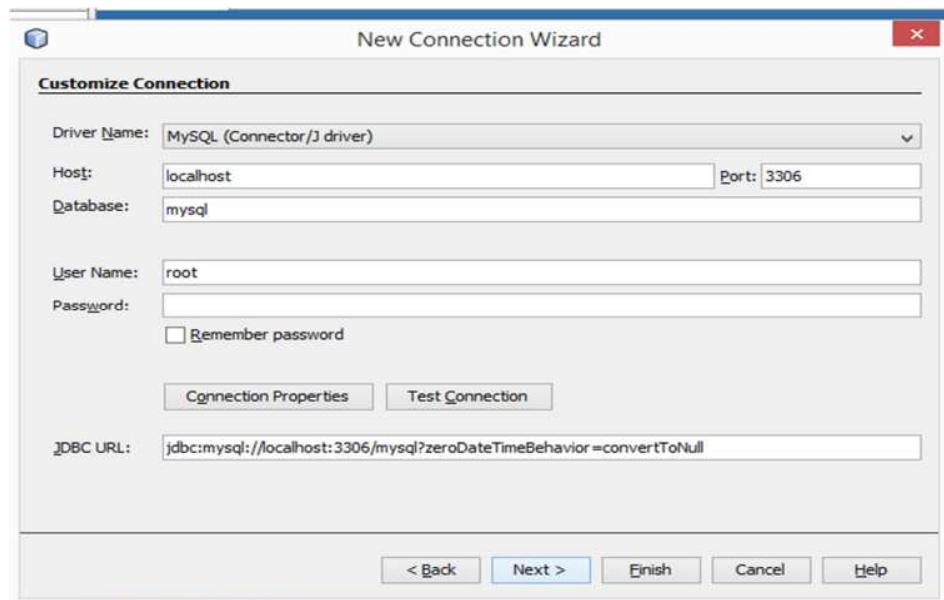
7. Right click on Databases > New Connection.



8. Select MySQL Driver from dropdown.



9. Click on Next.



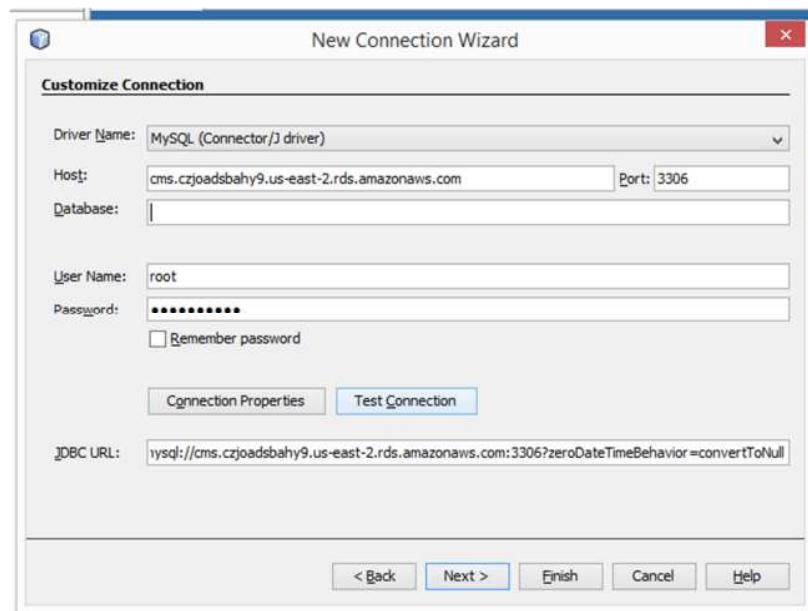
10. Provide connection details.

Host : Paste Endpoint of MySQL RDS.

Database: provide **database name** if you have already created otherwise **blank**.

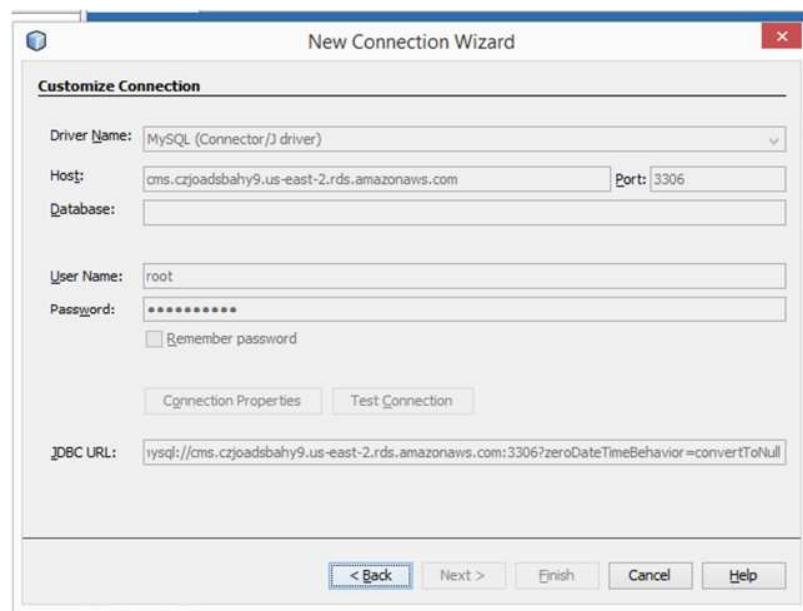
Username: Database **Username (root)**

Password: Database **Password (mypassword)**

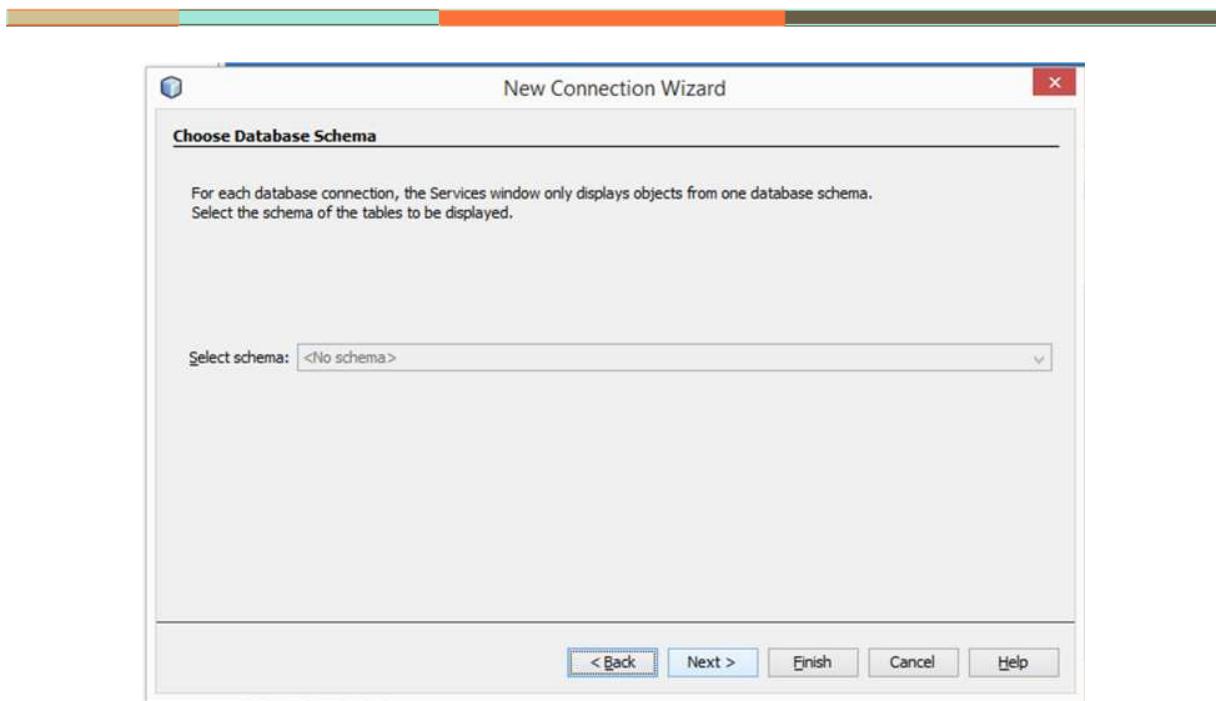


11. Click on Test Connection.

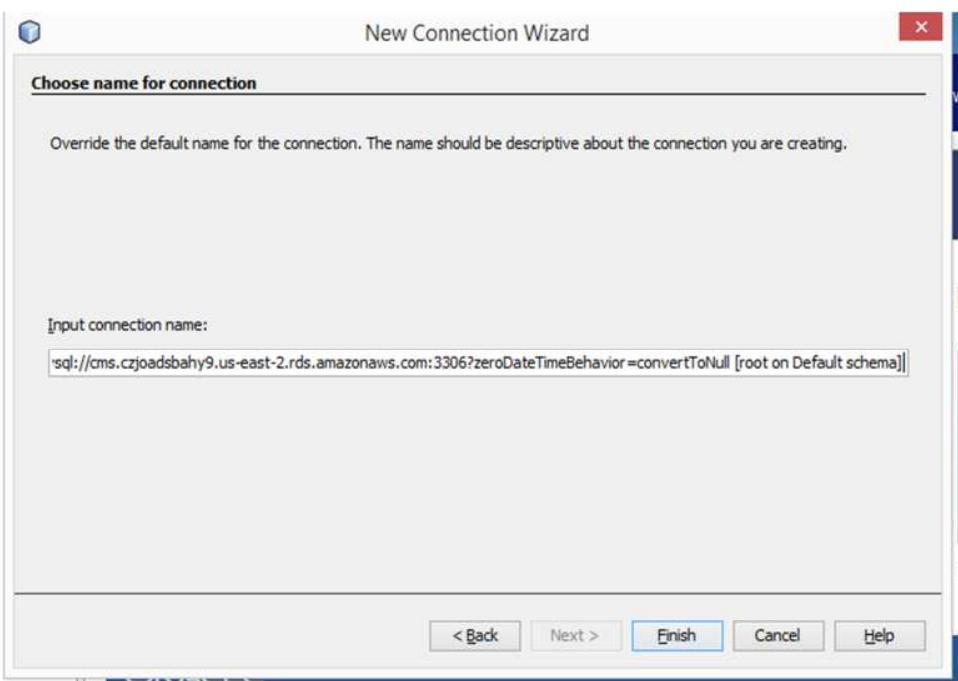
12. If Connection is successfully then click on Next.



13. Select Schema, We do not have any schema yet.

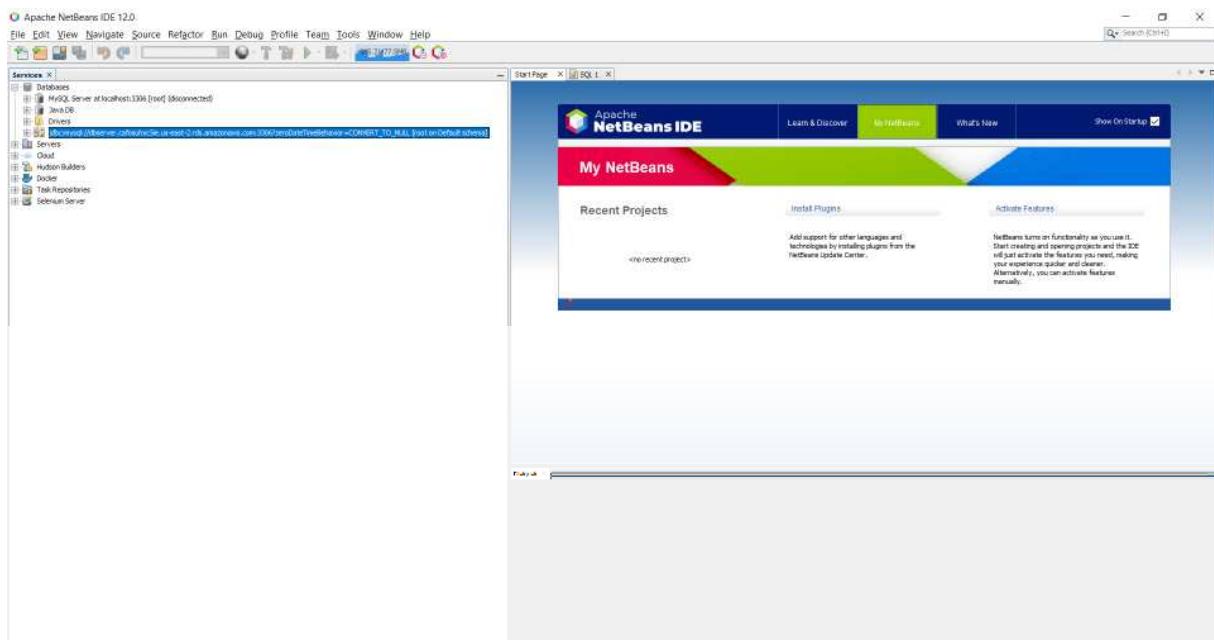


14. Click on Next.

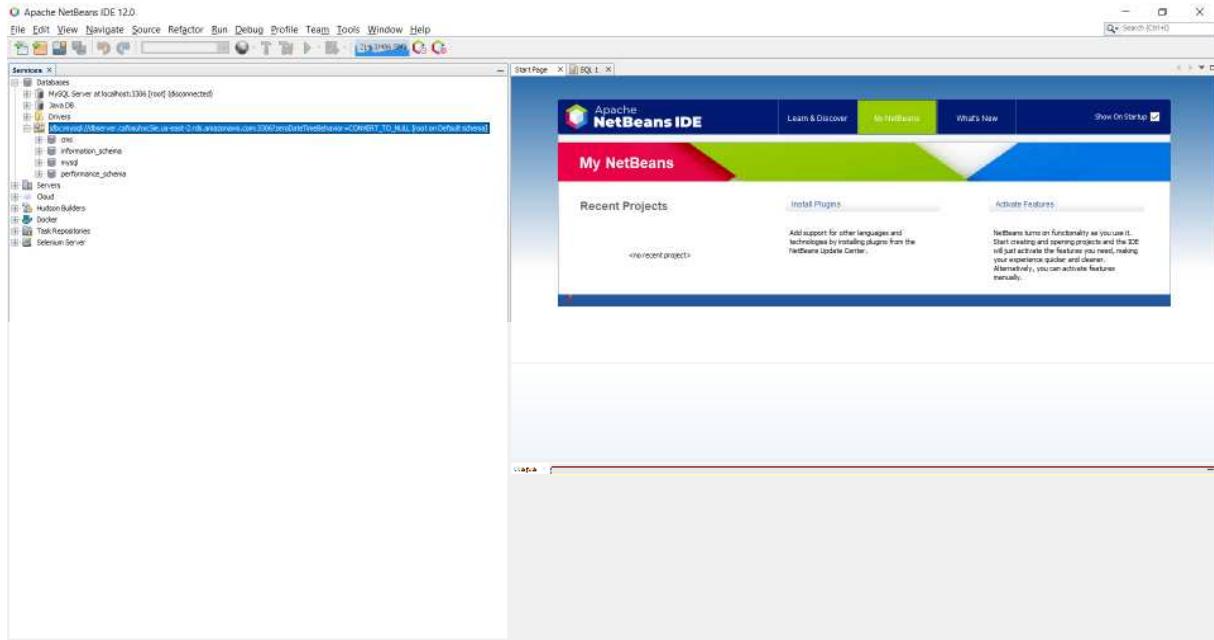


15. Click on Finish.

16. Now, Expand the Databases in NetBeans.



17. Expand MySQL RDS Connection String.



18. Database name will appear in the list. Now it's easy for us to create database, tables and also query on tables of MySQL RDS.

Congratulation :) Successfully Connected NetBeans with Amazon RDS.