

# Hands on with Cloud Computing

25th & 26th -August-2018

---

Created by:  
Ankit Velani, Analyst  
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

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

2. Click on *Create a Free account*

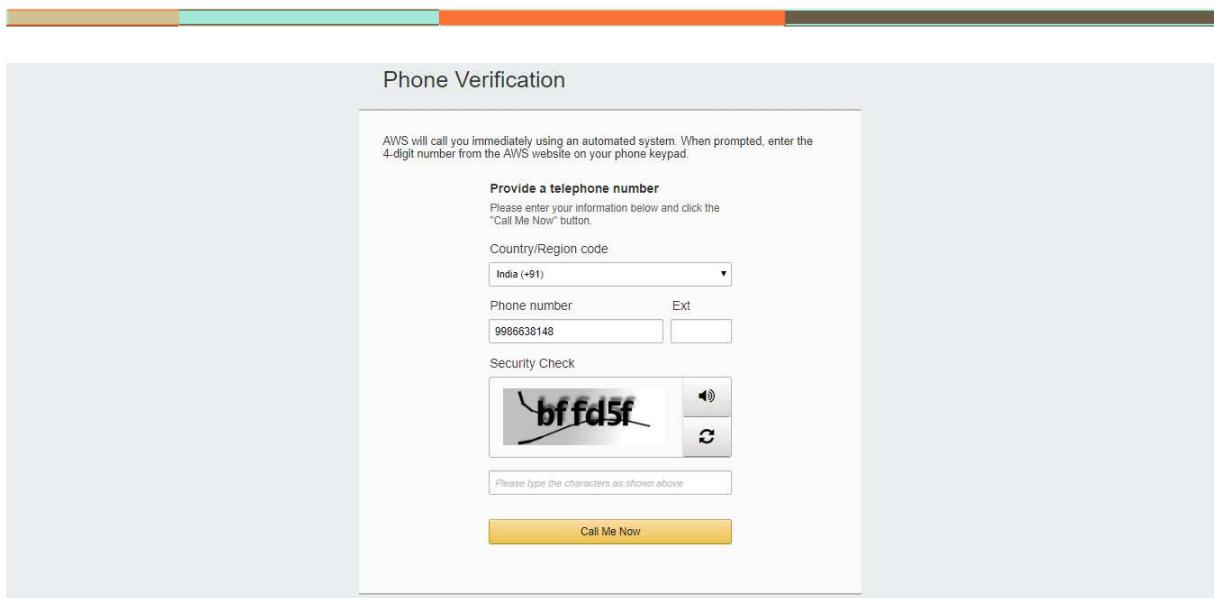
3. Enter **email** address and **password** for AWS account creation.
4. Enter your personal details to complete the AWS account.

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 radio buttons: 'Professional' (selected) and 'Personal'. The 'Full name' field contains 'Ankit Velani'. The 'Phone number' field contains '0988638148'. The 'Country/Region' dropdown is set to 'India'. A note states: '\* If you selected India, your country/region selection cannot be changed after creating the account.' The 'Address' field contains 'Bangalore'. The 'City' field contains 'Bangalore'. The 'State / Province or region' field contains 'Karnataka'. The 'Postal code' field contains '560037'. Below these fields is the 'Amazon Internet Services Pvt. Ltd. Customer Agreement'. It includes a note: 'Customers with an India contact address are now required to contract with Amazon Internet Service Private Ltd. (AISPL). AISPL is the local seller for AWS infrastructure services in India.' There is a checked checkbox next to the text: 'Check here to indicate that you have read and agree to the terms of the AISPL Customer Agreement.' At the bottom right is a yellow 'Create Account and Continue' button.

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, instructions say '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 note in a box says: 'As part of our card verification process we will charge INR 2 on your card when you click the "Secure Submit" button below. This will be refunded once your card has been validated. Your bank may take 3-5 business days to show the refund. Mastercard/Visa customers may be redirected to your bank website to authorize the charge.' The form fields include: 'Credit/Debit card number' (input field), 'Expiration date' (dropdowns showing '08' and '2019'), 'Cardholder's name' (input field), 'Billing address' (input field containing 'Bangalore Bangalore Karnataka 560037 IN'), 'Use my contact address' (radio button selected), 'Use a new address' (radio button), 'Do you have a PAN? ⓘ' (checkboxes for 'Yes' and 'No'), and a 'Secure Submit' button at the bottom.

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.

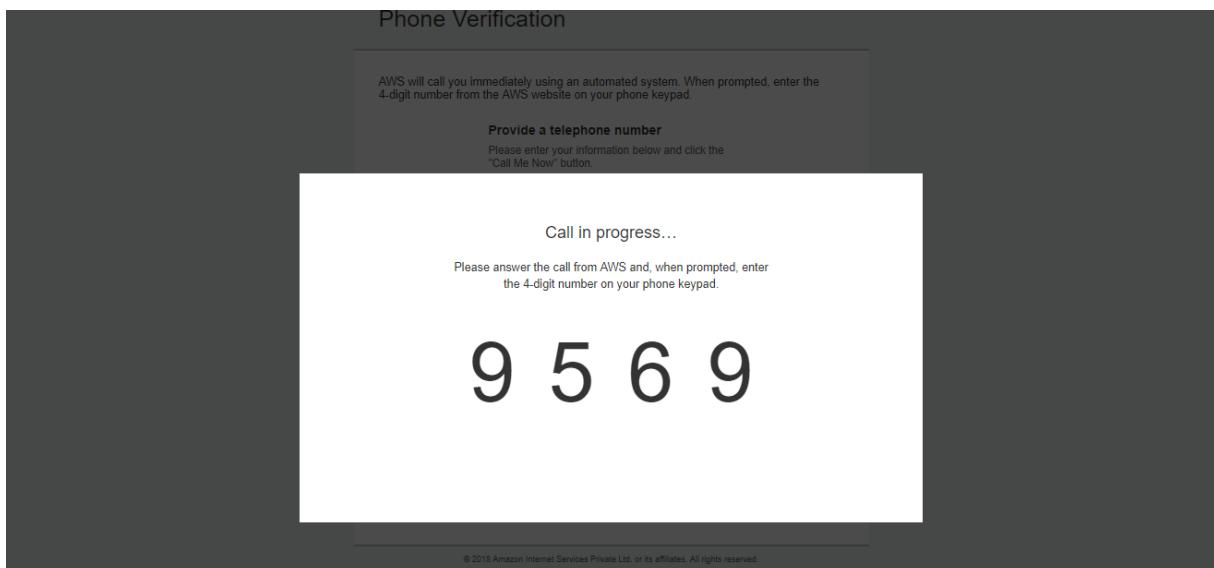
**Provide a telephone number**  
Please enter your information below and click the "Call Me Now" button.

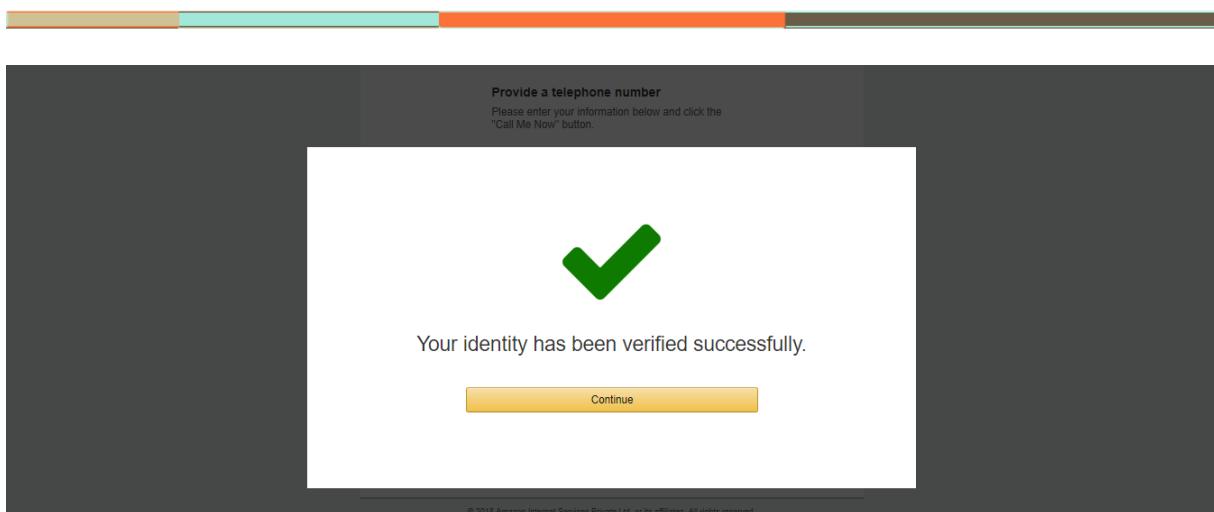
Country/Region code

Phone number Ext

Security Check  


*Please type the characters as shown above*





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

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

12. Click on Continue.

13. Congratulation, you have successfully registered with Amazon Cloud Platform.

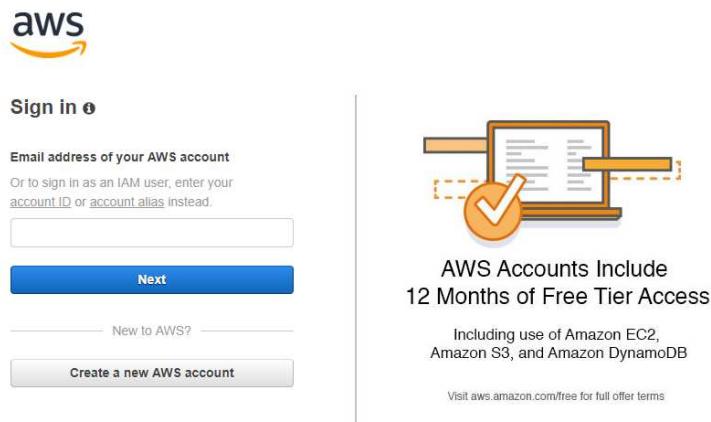


## Setup EC-2 on Amazon Cloud.

### 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://console.aws.amazon.com>  
 Enter your registered email and click on **Next**.



2. Enter your password and click on **Sign in**  
 3. After successfully login, Console landing page will appear.

The screenshot shows the AWS Home Page. In the top navigation bar, there are links for Services, Resource Groups, and Support, along with user information for Ankit SIT and Ohio. The main content area is titled "AWS services" and features a search bar. Below the search bar, there's a section for "Recently visited services" with icons for Billing and EC2. A "Build a solution" section follows, containing six quick-start wizards: Launch a virtual machine (With EC2, ~2-3 minutes), Build a web app (With Elastic Beanstalk, ~6 minutes), Build using virtual servers (With Lightsail, ~1-2 minutes), Connect an IoT device (With AWS IoT, ~5 minutes), Start a development project (With CodeStar, ~5 minutes), and Register a domain (With Route 53, ~3 minutes). To the right, there's a "Helpful tips" section with links to "Manage your costs" and "Create an organization". Below these, an "Explore AWS" section highlights "Machine Learning with Amazon SageMaker" and "Amazon Relational Database Service (RDS)".

#### 4. Click on All Services ,

This screenshot shows the same AWS Home Page as above, but the "All services" section is now expanded. It lists various service categories: Compute (EC2, Lightsail, Elastic Container Service, EKS, Lambda, Batch, Elastic Beanstalk), Storage (S3, EFS, Glacier, Storage Gateway), Management Tools (CloudWatch, AWS Auto Scaling, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Systems Manager, Trusted Advisor, Managed Services), Mobile Services (Mobile Hub, AWS AppSync, Device Farm, Mobile Analytics), AR & VR (Amazon Sumerian), Application Integration (Step Functions, Amazon MQ, Simple Notification Service), and Media Services (Elastic Transcoder). The rest of the page content (Helpful tips, Explore AWS sections) remains the same.

#### 5. Choose EC2 , from category “ Compute “

The screenshot shows the AWS Management Console homepage. At the top, there's a navigation bar with links for Apps, DNI Institute, R Tutorial-Learn, Lesson 1: What is Data Science?, Dataaspirant, Mathematics, Collaborative File, Linear Regression, and user profile information for Ankit, Ohio, and Support.

The main content area is titled "AWS services" and features a search bar. It lists recently visited services and all services under categories:

- Compute**: EC2, Lambda, Batch, Lightsail, Elastic Beanstalk, and others.
- Storage**: S3, EFS, Glacier, Storage Gateway.
- Database**: MySQL, PostgreSQL, Oracle, and others.
- 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.

On the right side, there are "Helpful tips" and "Explore AWS" sections. The "Helpful tips" section includes "Manage your costs" (with a link to start now) and "Create an organization" (with a link to start now). The "Explore AWS" section includes "New Product Announcements" (with a link to learn more) and "Migrate from Oracle to Amazon Aurora" (with a link to view project).

The URL in the address bar is <https://us-east-2.console.aws.amazon.com/console/home?region=us-east-2#>.

6. It jumped to EC2 Dashboard. Click on **Launch Instance**

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with various navigation links like EC2 Dashboard, Instances, AMIs, and Network & Security. The main area displays resource counts: 0 Running Instances, 0 Dedicated Hosts, 0 Volumes, 0 Key Pairs, 0 Elastic IPs, 0 Snapshots, 0 Load Balancers, and 1 Security Groups. Below this, a 'Create Instance' section is shown with a 'Launch Instance' button highlighted in blue. To the right, there are sections for Account Attributes (Supported Platforms: VPC, Default VPC: vpc-f9e9da91), Additional Information (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us), and AWS Marketplace.

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

The screenshot shows the 'Choose an Amazon Machine Image (AMI)' step of the instance creation wizard. The 'Free tier only' checkbox is checked, highlighting the 'Amazon Linux 2 AMI (HVM), SSD Volume Type' option. This option is described as being free for five years, providing a 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. It uses an EBS-backed, AWS-supported image with a root device type of ebs and virtualization type of hvm. There are other options listed below it, such as 'Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type' and 'Red Hat Enterprise Linux 7.5 (HVM), SSD Volume Type'. Each option has a 'Select' button to its right.

8. Select AMI type Ubuntu **Server 14.04 LTS (HVM), SSD Volume Type** - ami-019abc64 and click on Select.
9. Next step, choose instance type. Select the **t2.micro (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 vCPUs, 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 <span style="background-color: #33A356; color: white;">Free tier eligible</span>	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	EBS only	-	Low to Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

T2.micro 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>

10. Click on Review and Launch.

**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** [Edit AMI](#)

**Free tier eligible** Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-7d132e18  
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** [Edit 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** [Edit security groups](#)

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

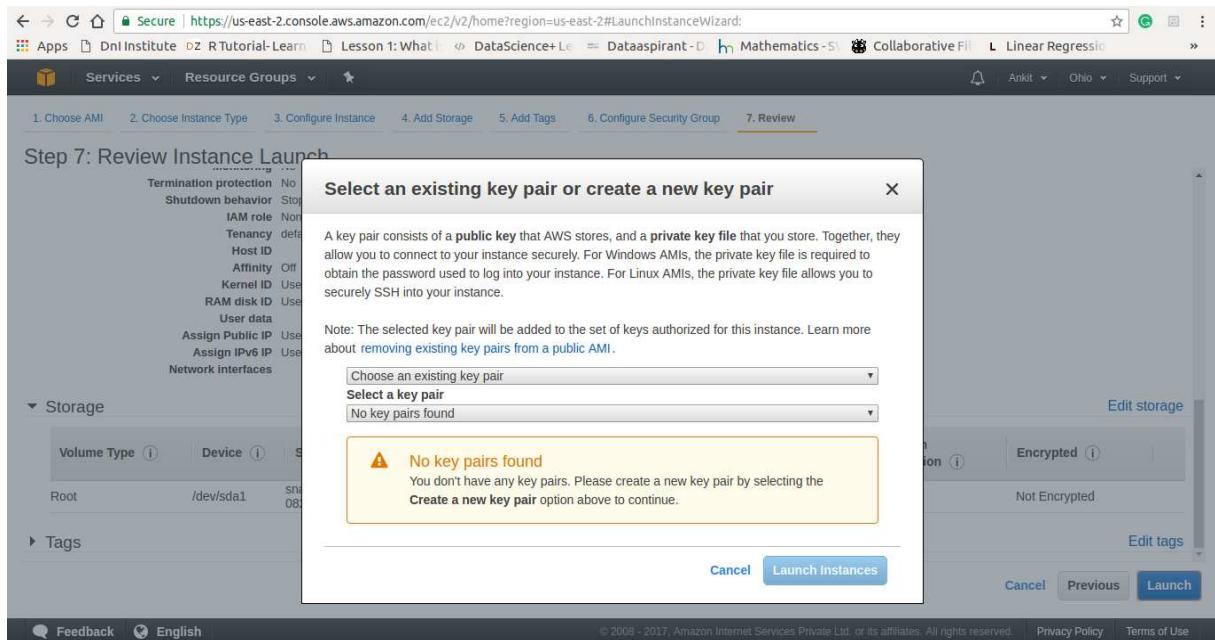
Cancel Previous Launch

11. Click on Launch.

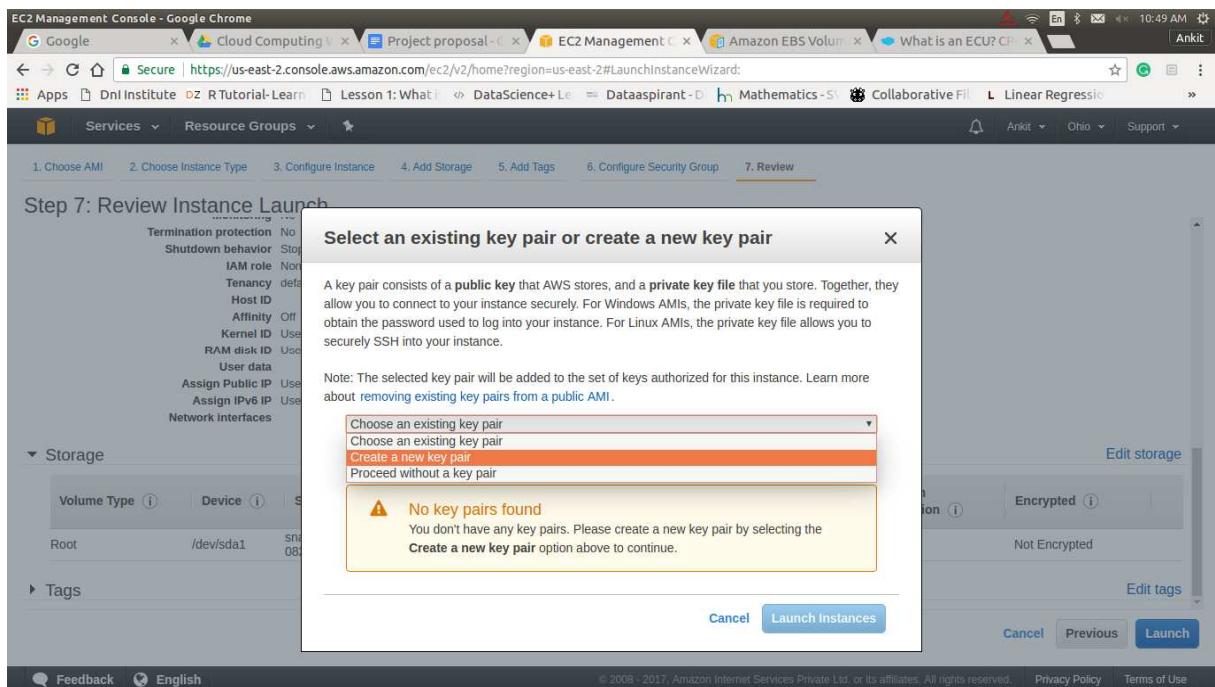
Hands on Cloud Computing Workshop.

Dept. Of MCA – SIT, Tumkuru

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



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



14. Write KeyName and Download Key Pair

Step 7: Review Instance Launch

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.

Create a new key pair

Key pair name: AmazonEC2

Download Key Pair

You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel Launch Instances

### 15. After Download, Click on Launch Instances.

Secure | https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:

Your instances are now launching  
The following instance launches have been initiated: i-0ad2d2f2e17281327 View launch log

Get notified of estimated charges  
Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances  
Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.  
Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can [connect](#) to them from the Instances screen. [Find out](#) how to connect to your instances.

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your instances are launching you can also

- Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
- Create and attach additional EBS volumes (Additional charges may apply)
- Manage security groups

[View Instances](#)

16. Click on [View Instances](#).

17. Back to EC2 Dashboard, under [INSTANCES](#), recently created instance will appear.

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, 'Instances' is selected. In the main content area, there is a table with one row. The row details a single instance:

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

Below the table, there is a detailed view of the instance 'i-0ad2d2f2e17281327'. The 'Description' tab is selected, showing the following information:

- Instance ID: i-0ad2d2f2e17281327
- Public DNS (IPv4): ec2-52-14-56-150.us-east-2.compute.amazonaws.com
- IPv4 Public IP: 52.14.56.150
- IPv6 IPs: -
- Private DNS: ip-172-31-32-230.us-east-2.compute.internal
- Private IPs: 172.31.32.230
- Availability zone: us-east-2c

18. 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.

19. Description of the instance.

The screenshot shows the AWS EC2 Instances page with the same interface as the previous screenshot, but the 'Description' tab is not selected. Instead, the 'Status Checks' tab is selected, showing the following information for the instance 'i-0ad2d2f2e17281327':

- 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-20170619 (ami-019abc64)
- Platform: -
- IAM role: -
- Key pair name: AmazonEC2

On the right side of the screen, there is a large table with many columns, likely showing network interface details or security group associations, which is partially cut off by the right edge of the screenshot.

## 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.

The screenshot shows the AWS EC2 Management Console in Google Chrome. The URL is https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:Step1. The page title is "Step 1: Choose an Amazon Machine Image (AMI)". The navigation bar at the top includes tabs for "Choose AMI", "Choose Instance Type", "Configure Instance", "Add Storage", "Add Tags", "Configure Security Group", and "Review". On the left, there's a sidebar with "Services" and "Resource Groups". The main content area displays a list of Windows AMIs:

AM Name	Description	Type	Select
Microsoft Windows Server 2016 Base Nano - ami-0994b56c	Microsoft Windows 2016 Datacenter Edition Nano. [English] Root device type: ebs Virtualization type: hvm	64-bit	Select
Microsoft Windows Server 2012 R2 Base - ami-ceaa8bab	Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English] Root device type: ebs Virtualization type: hvm	64-bit	Select
Microsoft Windows Server 2012 Base - ami-faa8f9a	Microsoft Windows 2012 Standard edition with 64-bit architecture. [English] Root device type: ebs Virtualization type: hvm	64-bit	Select
Microsoft Windows Server 2008 R2 Base - ami-37ab8a52	Microsoft Windows 2008 R2 SP1 Datacenter edition, 64-bit architecture. [English] Root device type: ebs Virtualization type: hvm	64-bit	Select
Microsoft Windows Server 2003 R2 Base - ami-82af8ee7	(64-bit) / ami-47a68722 (32-bit)		Select

At the bottom, there are links for "Feedback", "English", "© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.", "Privacy Policy", and "Terms of Use".

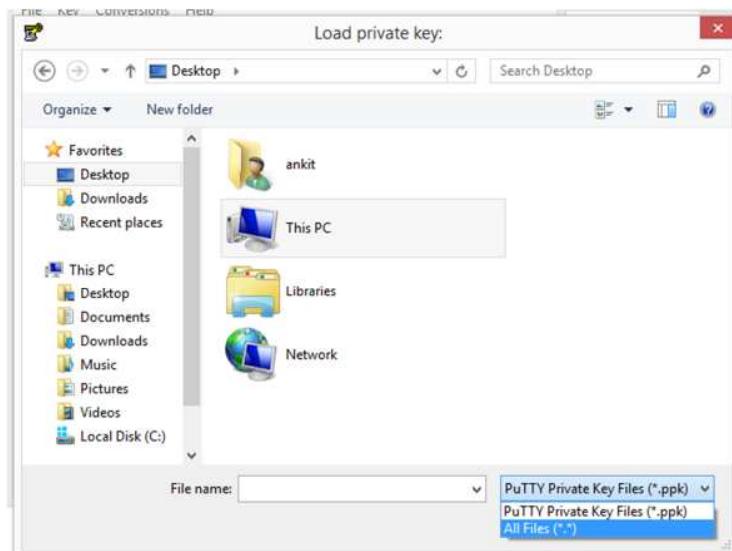
## Converting Encrypted file to ppk (Private Key) to access EC2

Note: Make sure that [Puttygen](#) should be installed in your machine.

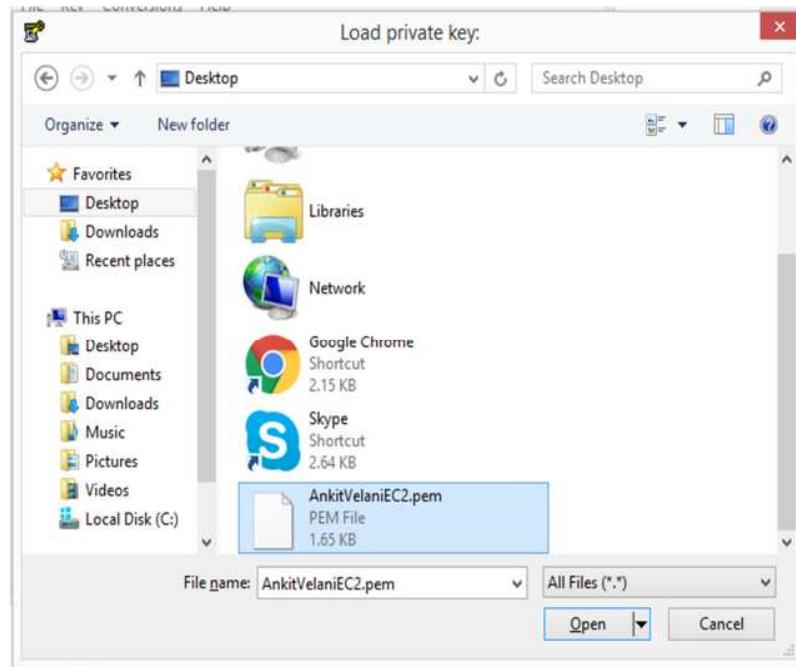
1. Open PuTTygen.



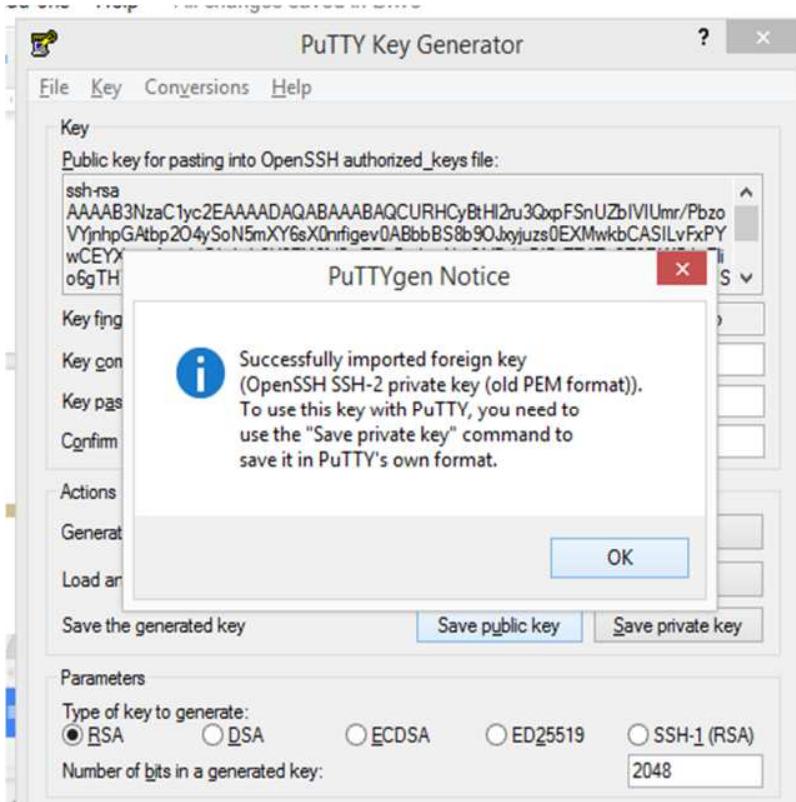
2. Click on **Load**, It gives FileBox , choose file type as **All Files(\*)**



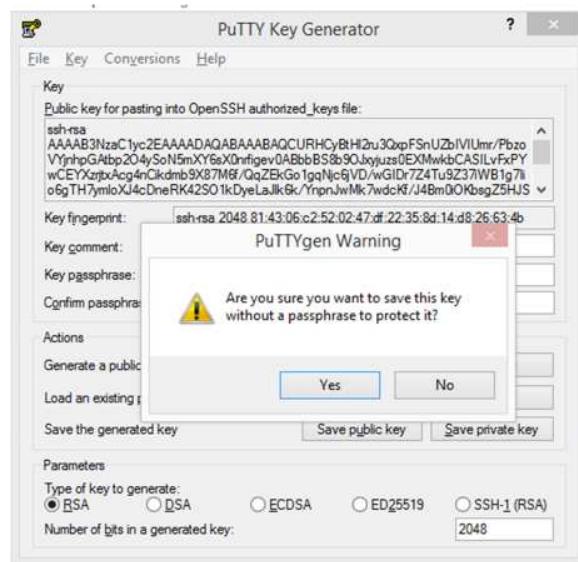
3. Select your .pem file and click on Open.



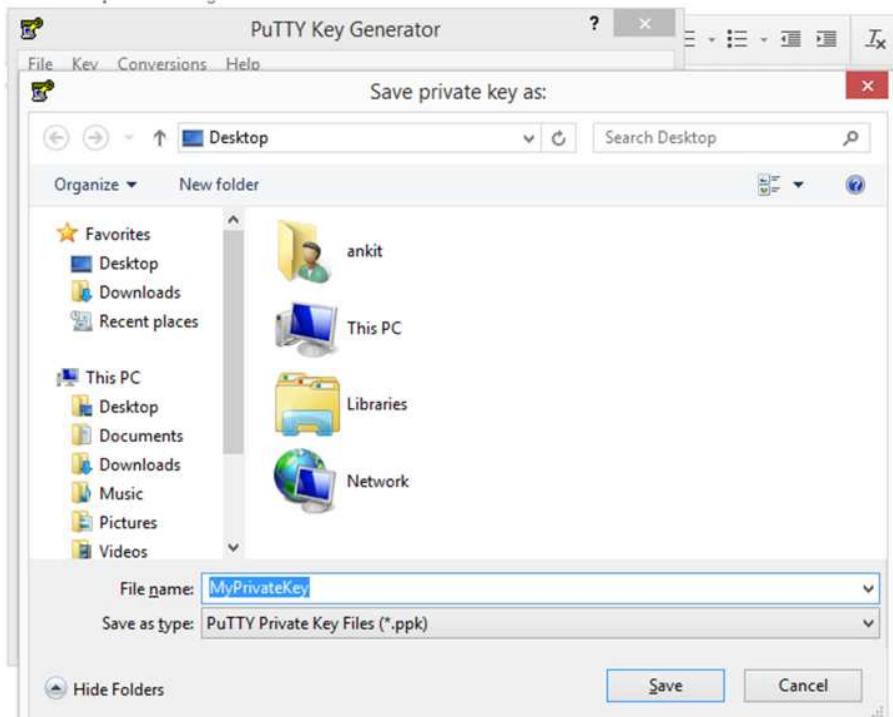
- Click on Ok , if Putty show any notice.



- 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.

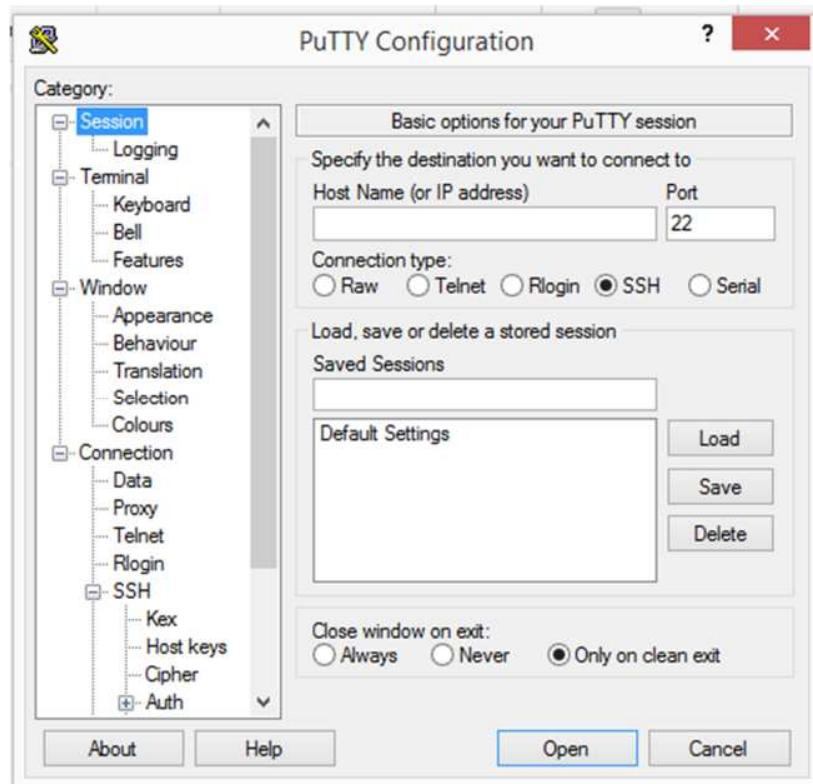
Congratulations :) Successfully created Private key(ppk) file.

## Accessing Amazon EC2 Machine using SSH Client

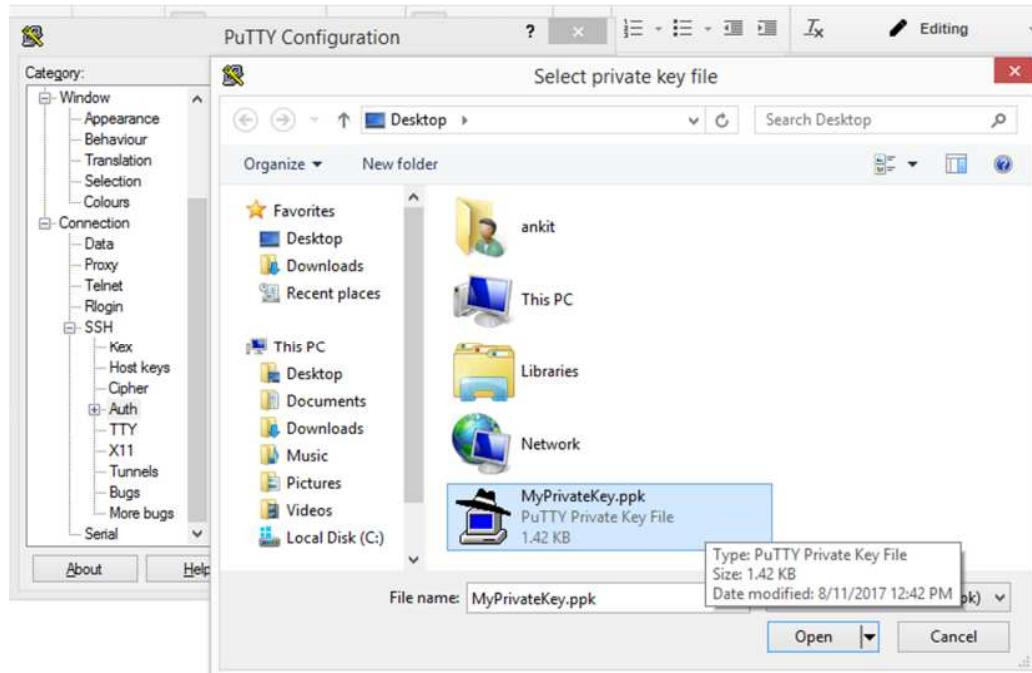
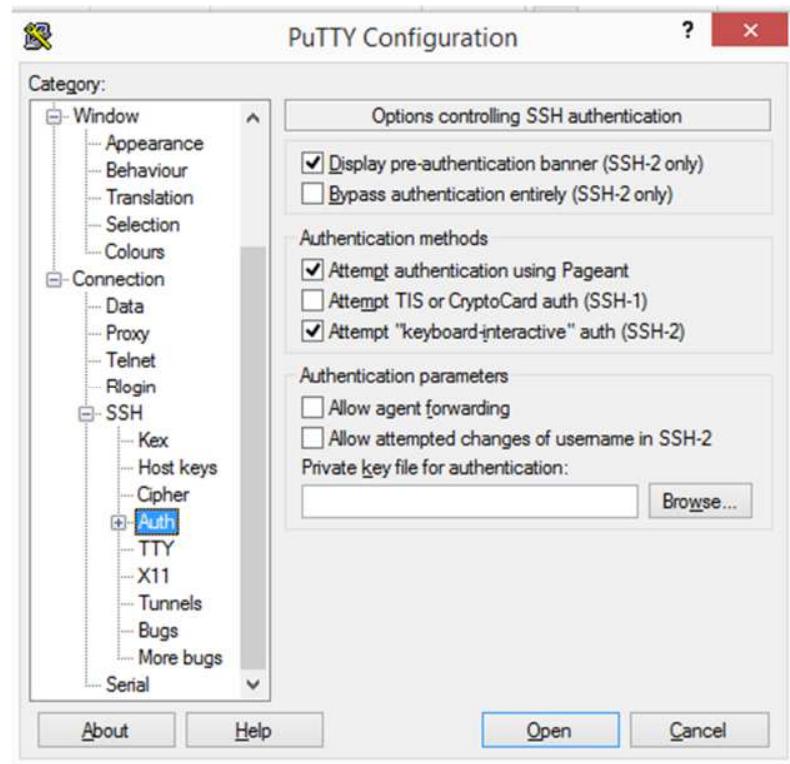
Note: Make sure that Putty ssh client should be installed in your machine.



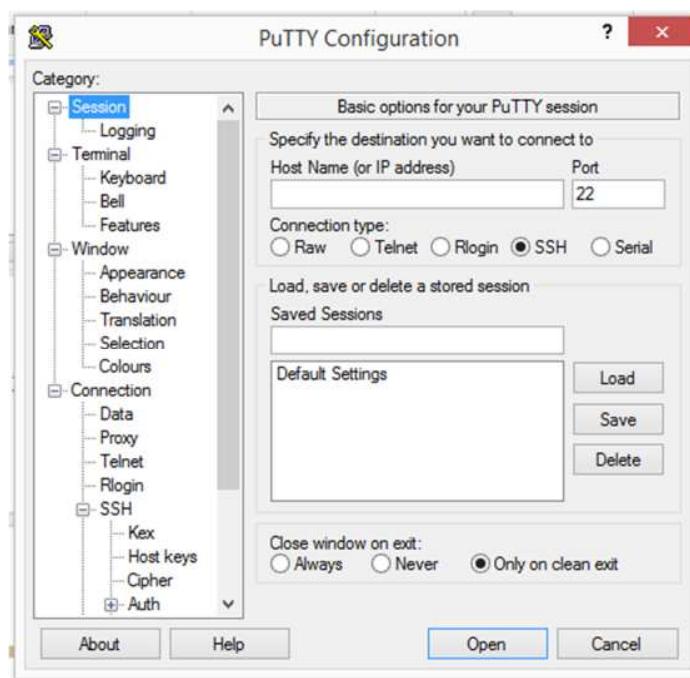
1. Open Putty



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>

7. Click EC2 under Compute Category.
8. 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

9. Click on Running Instances.

**Services** ▾ **Resource Groups** ▾

**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

**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 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** **Status Checks** **Monitoring** **Tags**

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

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

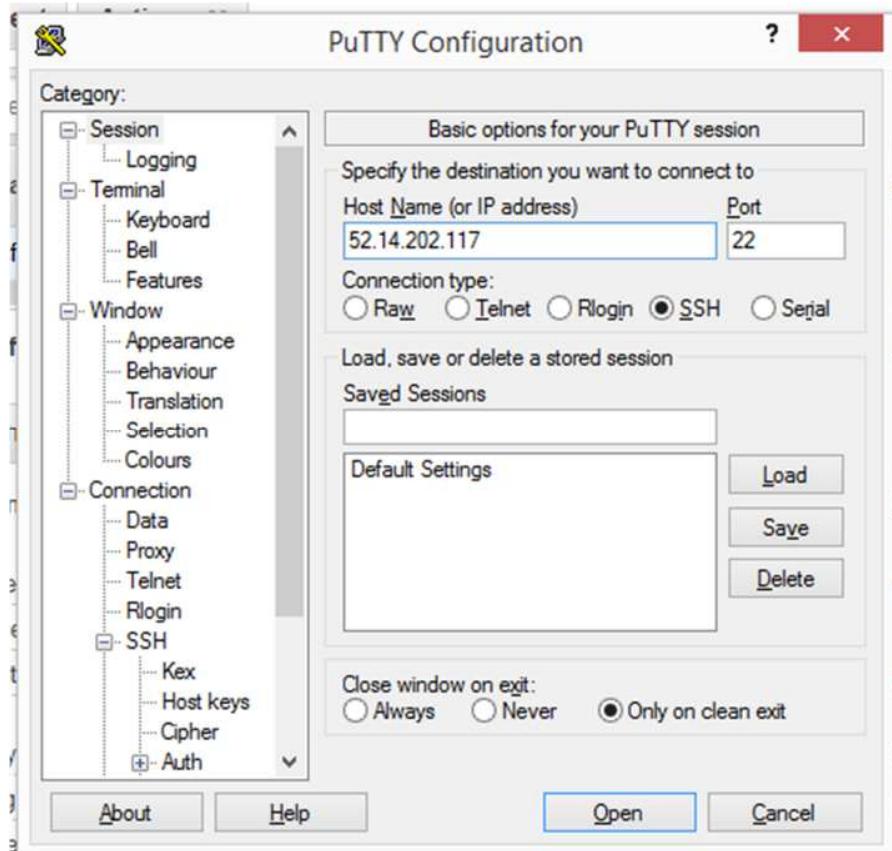
10. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc details.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links: EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (with Instances selected), Spot Requests, Reserved Instances, Dedicated Hosts, IMAGES (with AMIs selected), Bundle Tasks, ELASTIC BLOCK STORE (with Volumes and Snapshots), and NETWORK & SECURITY (with Security Groups). The main content area has tabs: Launch Instance, Connect, Actions, Description, Status Checks, Monitoring, and Tags. The Description tab is active. It displays details for a single instance: 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 this, there's a table with columns: Instance ID, Instance state, Instance type, Availability zone, Security groups, Scheduled events, AMI ID, Public DNS (IPv4), IPv4 Public IP, IPv6 IPs, Private DNS, Private IPs, Secondary private IPs, VPC ID, and Subnet ID. The table shows the same information for the instance.

### 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

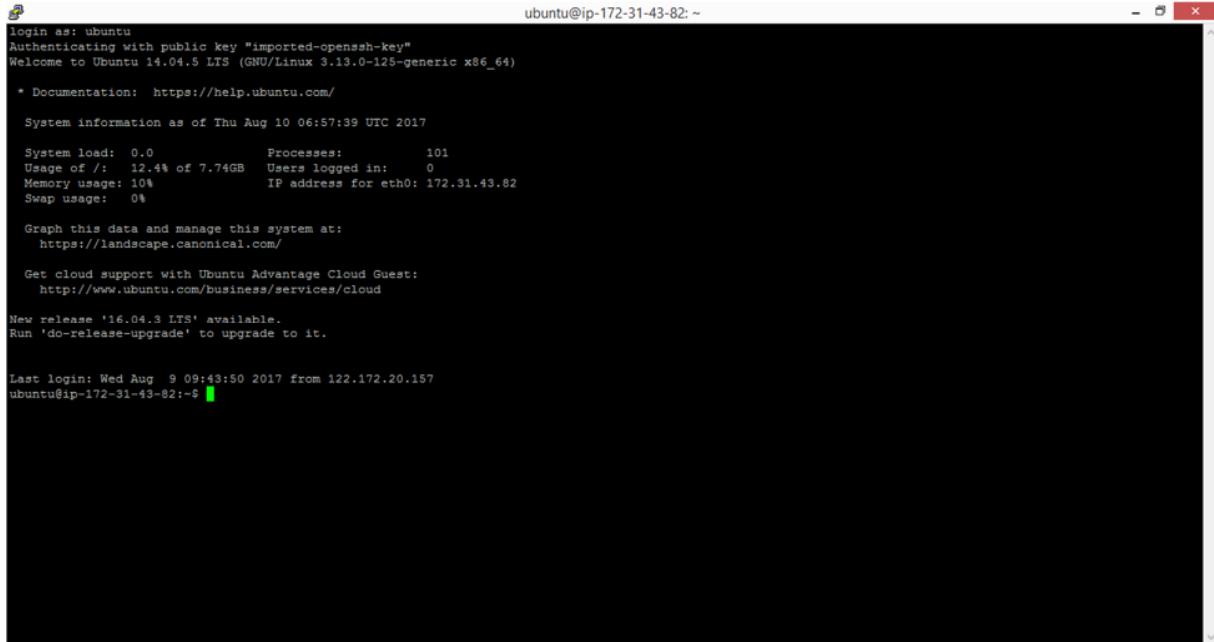
12. Go back to Putty SSH client , enter the Hostname as a Public IP.



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:~$
```

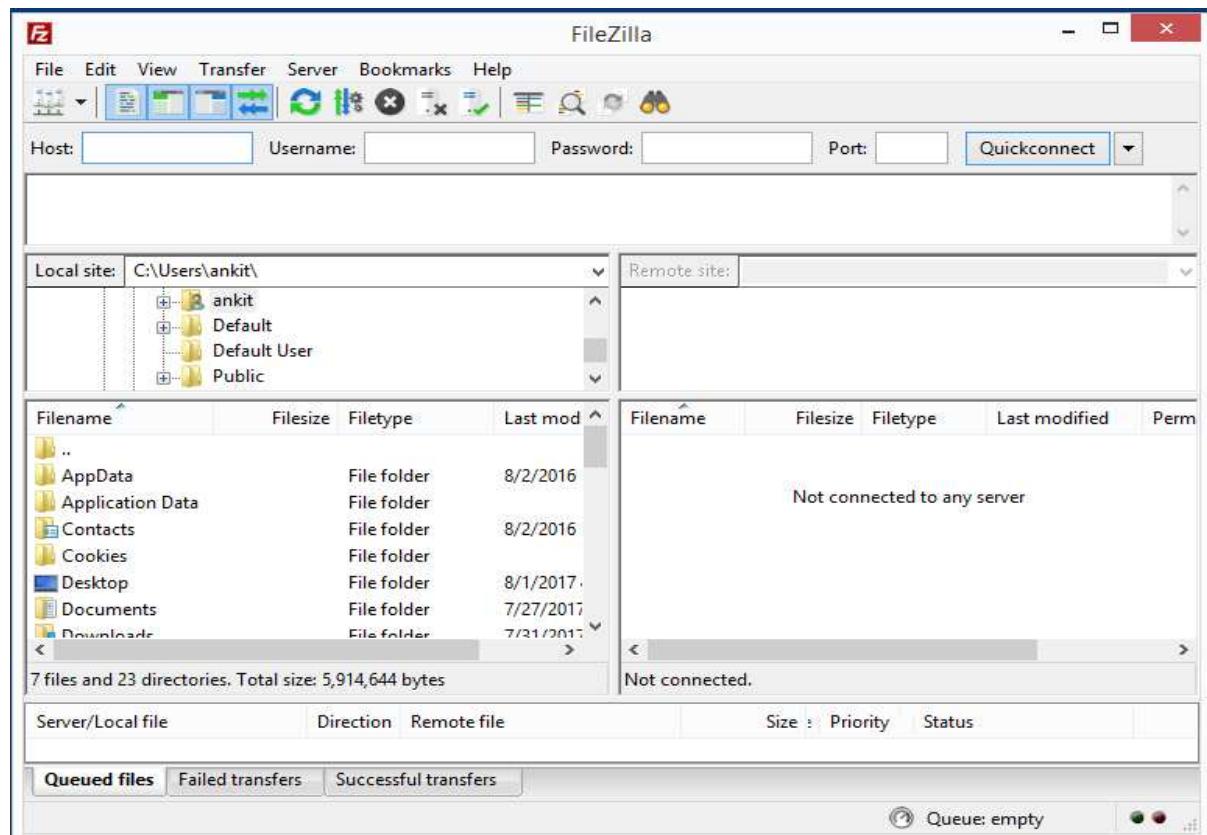
Congratulations :) you are connected to virtual machine using Putty SSH Client.

## Accessing Amazon EC2 Machine using FileManager

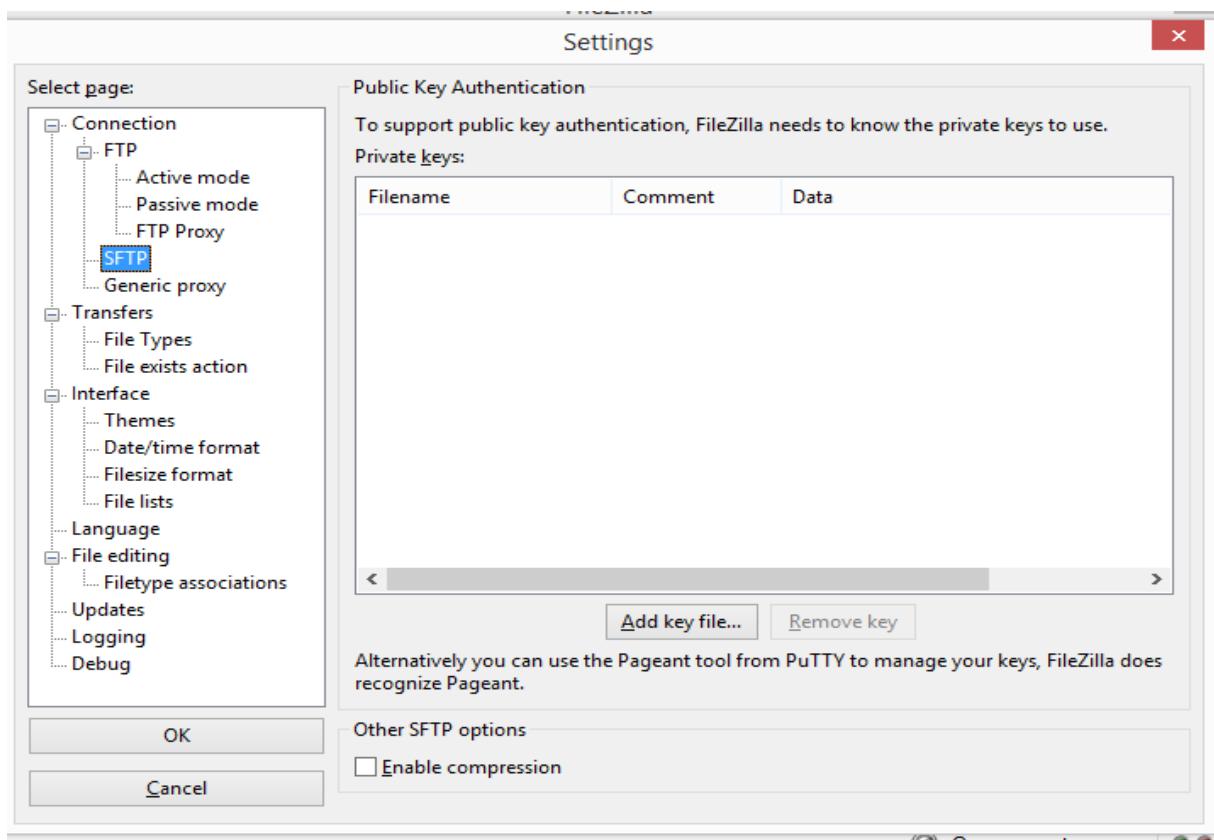
Note: Make sure that any one file manager (FileZilla, WinScp ..etc ) should be installed in your machine.



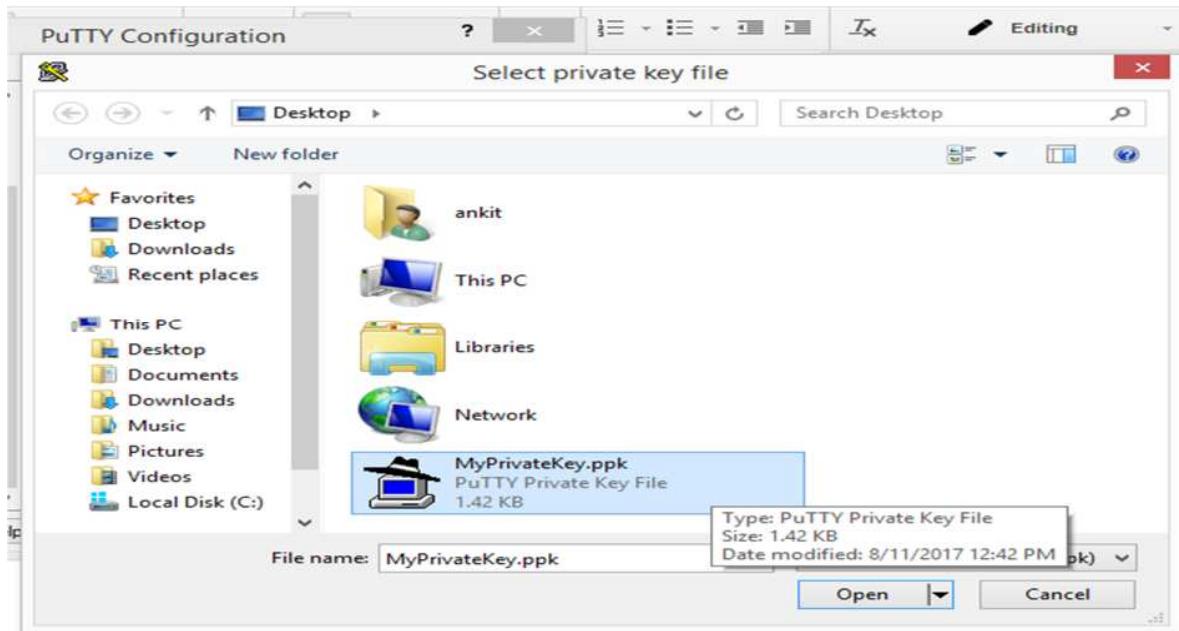
1. Open FileZilla.



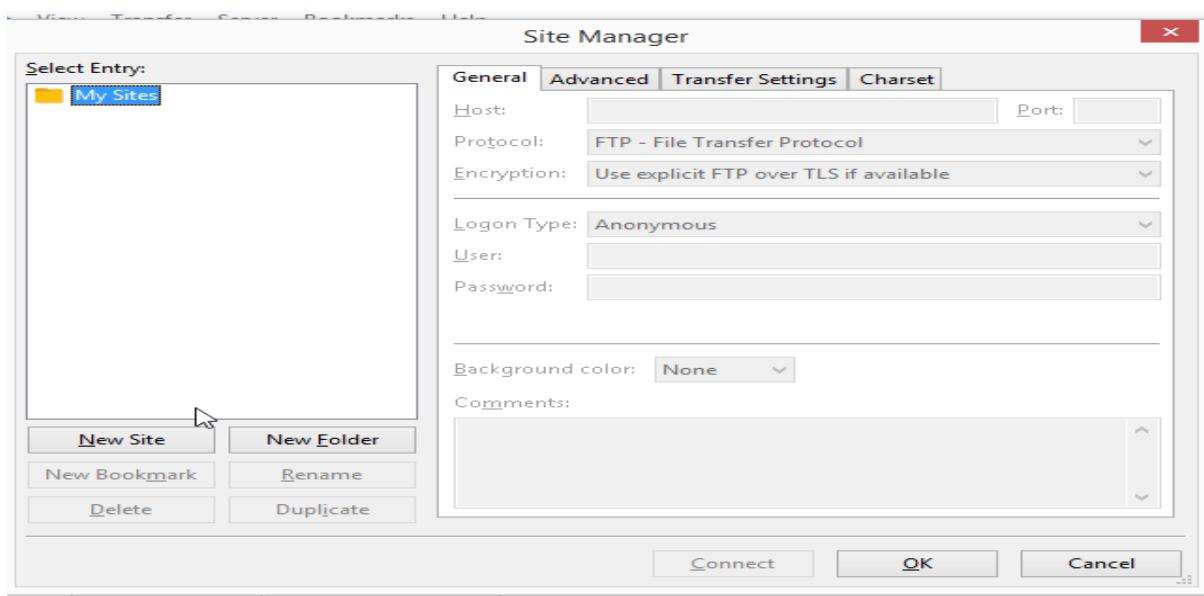
2. Go to **Edit Menu > Connection > SFTP.**



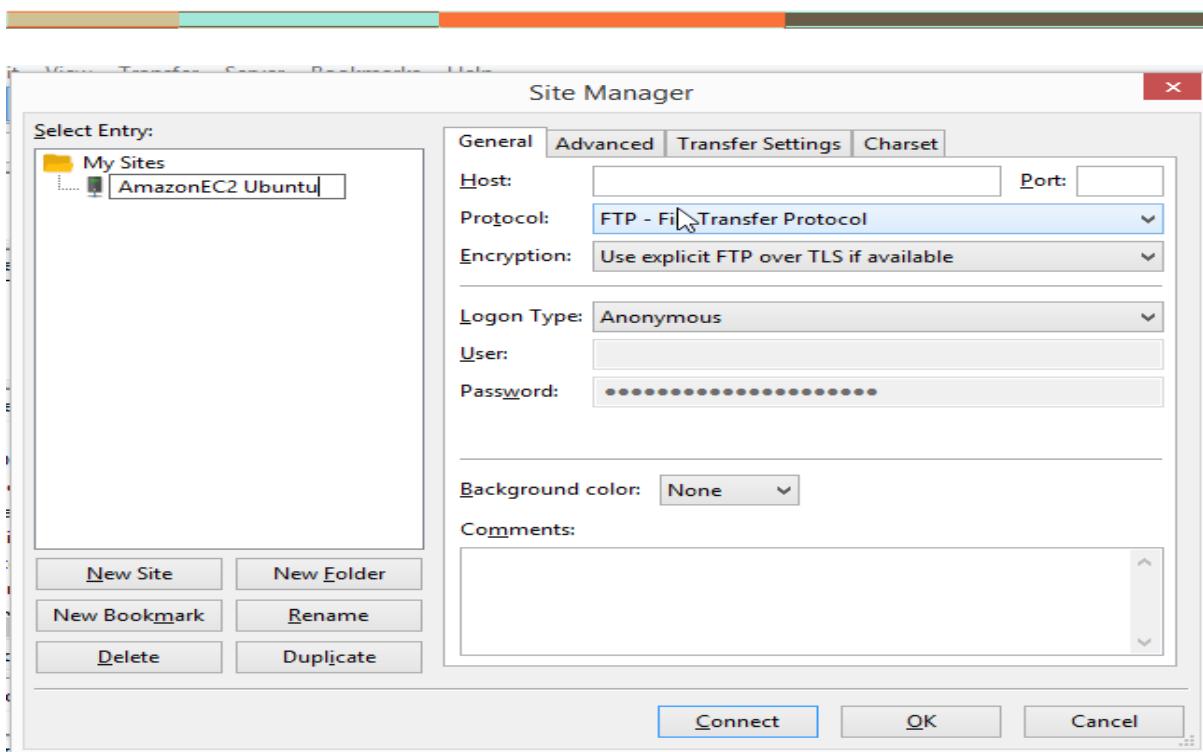
3. Click on **Add key file...**



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>

AWS services

- 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
  - Internet of Things
    - AWS IoT
    - AWS Greengrass
  - Contact Center
    - Amazon Connect
  - Management Tools
    - CloudWatch
    - CloudFormation
    - CloudTrail
    - Config
    - OpsWorks
    - Service Catalog
  - 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](#)

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**

**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	-
Flavor IDs		Private DNS	in-177-31-43-87.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 highlighted. 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. 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

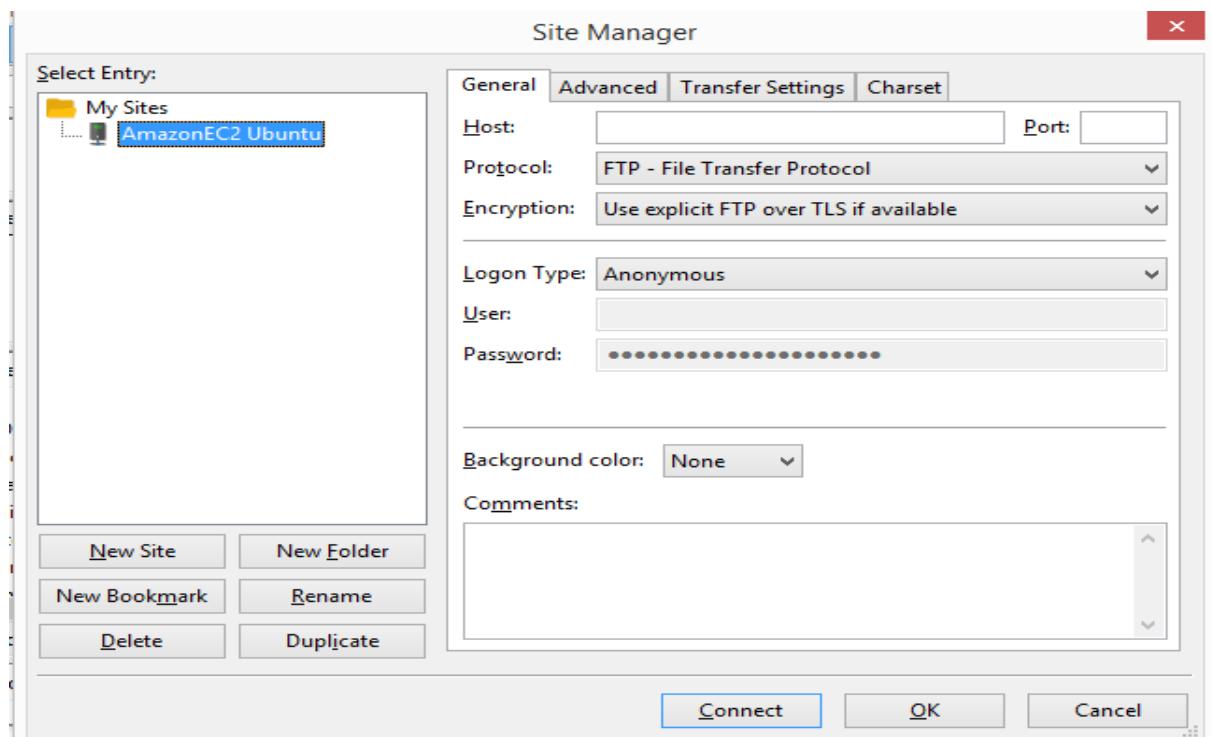
**Description**

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-43301076)	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.

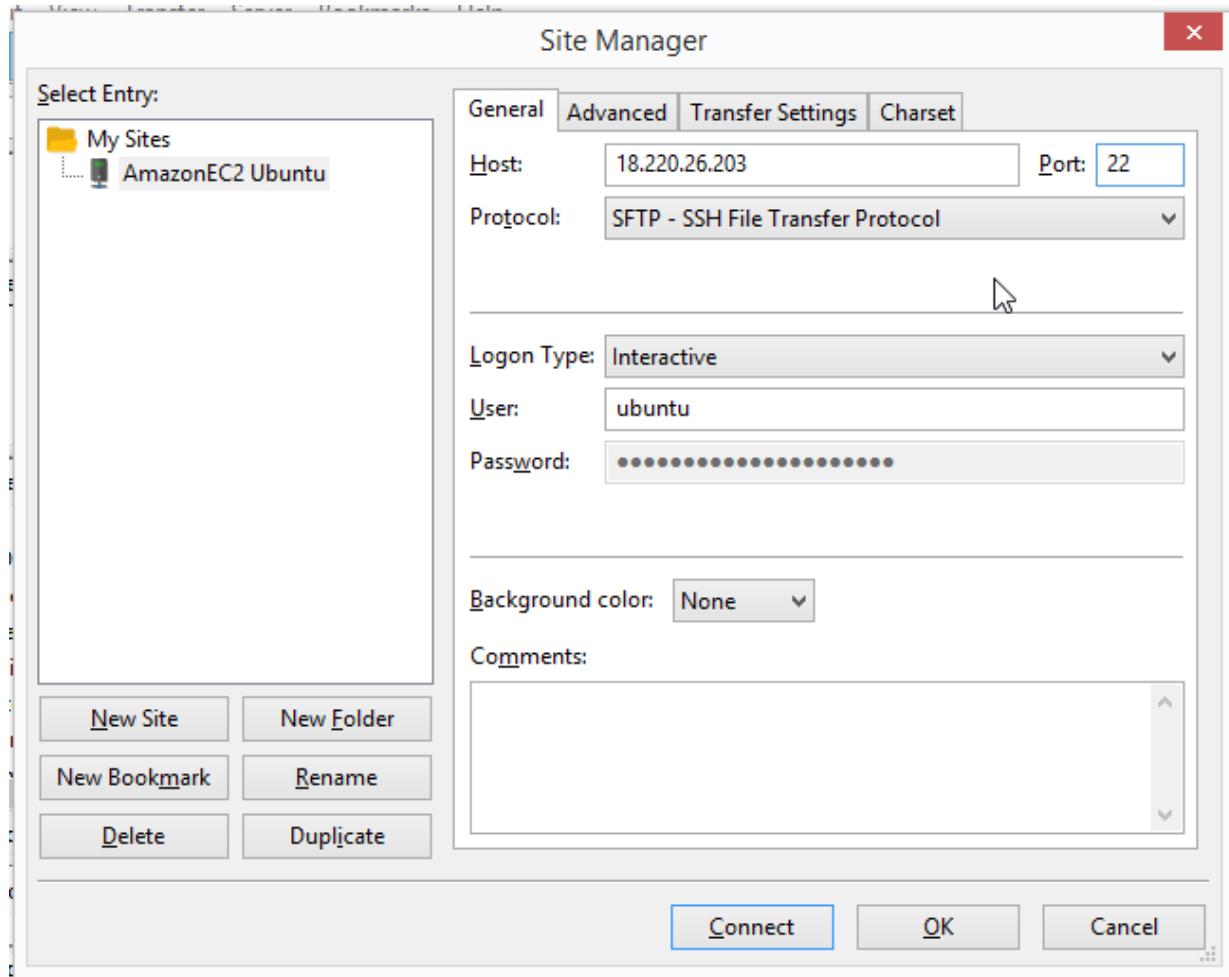
Host : Public IP

Port : 22

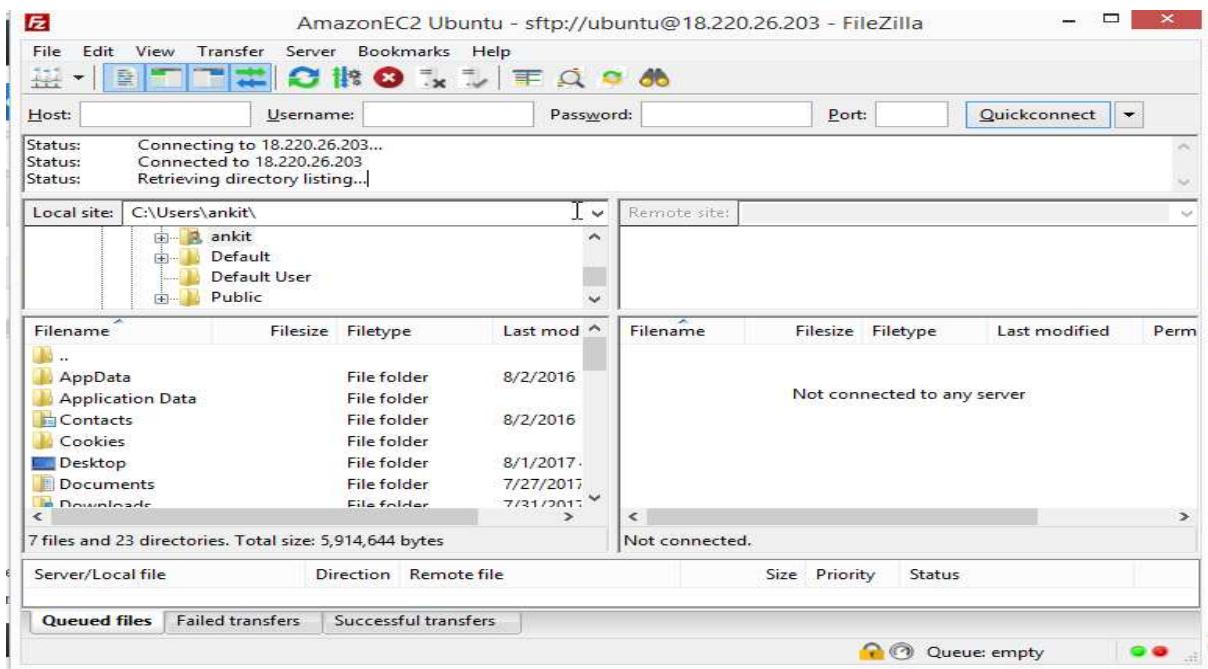
Protocol : SFTP

Logon Type : Interactive

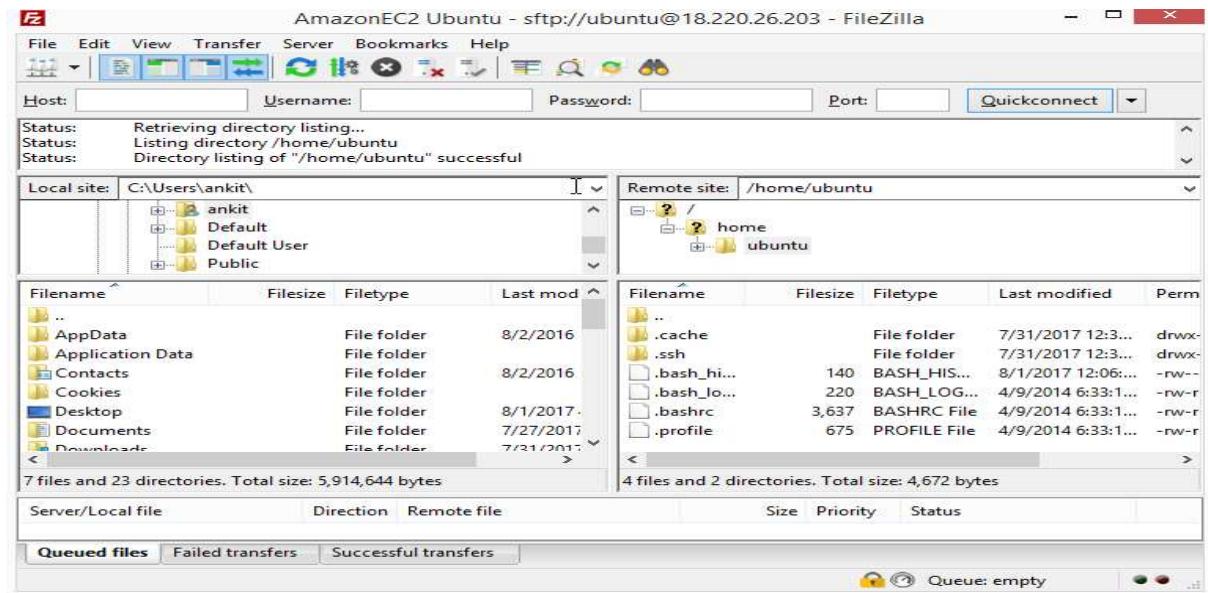
User : ubuntu



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.

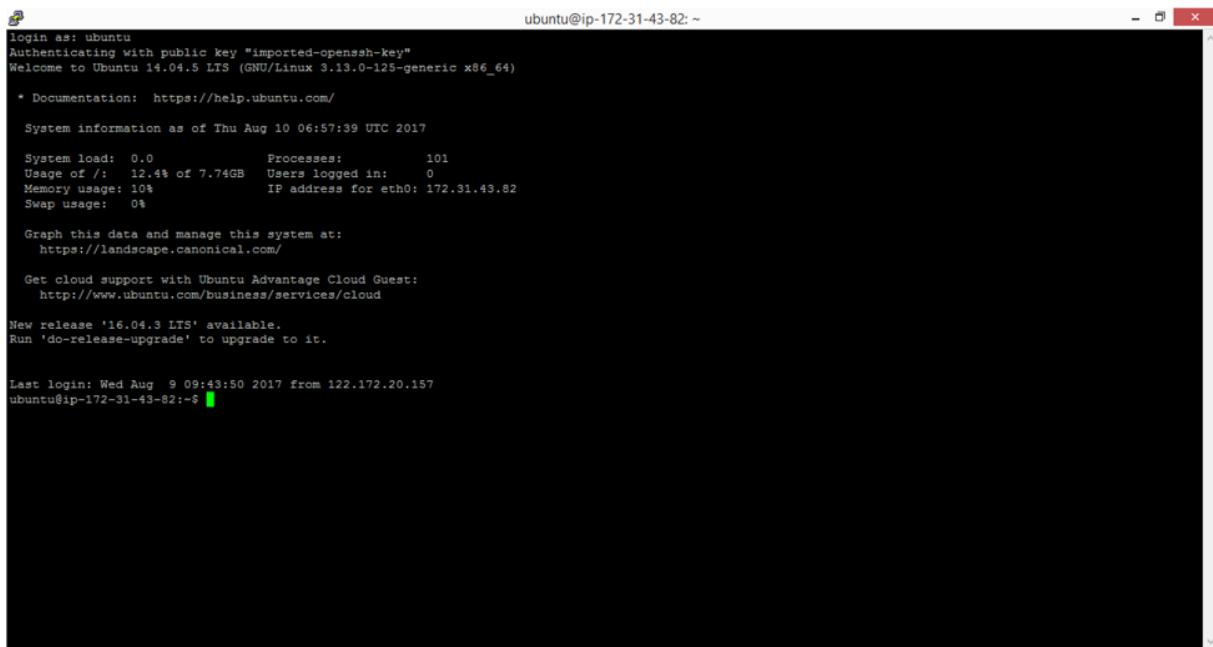


Note: You are accessing Amazon EC Virtual machine using FileZilla file manager, it's easy to download, upload and file manipulation.

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

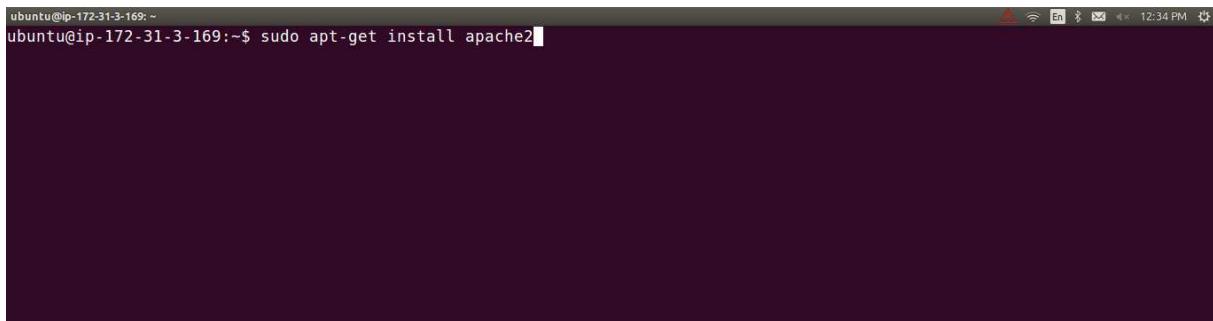
 System load: 0.0          Processes:           101
 Usage of /: 12.4% of 7.74GB   Users logged in: 0
 Memory usage: 10%          IP address for eth0: 172.31.43.82
 Swap usage: 0%
 
 Graph this data and manage this system at:
 https://landscape.canonical.com/
 
 Get cloud support with Ubuntu Advantage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

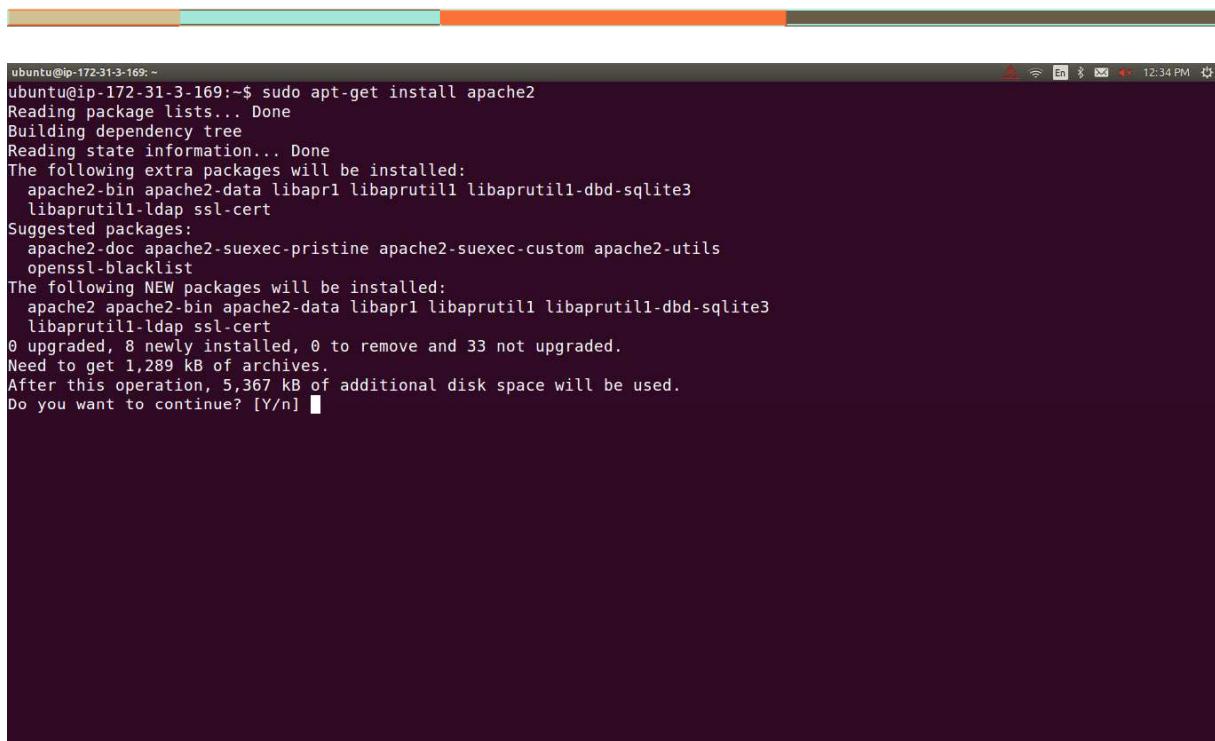
1. Execute command

```
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 'Do you want to continue? [Y/n]' is at the bottom.

```
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'. The cursor is at the end of the command.

A screenshot of a terminal window showing the output of the 'apt-get install' command. The terminal is dark-themed. The output shows the process of reading package lists, building a dependency tree, and determining packages to install. It lists extra packages like libmcrypt4, php5-cli, etc., and suggested packages like php-pear, libmcrypt-dev, etc. It also shows the size of the download (4,950 kB) and the disk space required (20.8 MB). The user is prompted with 'Do you want to continue? [Y/n]'. The cursor is at the end of the question.

```
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.

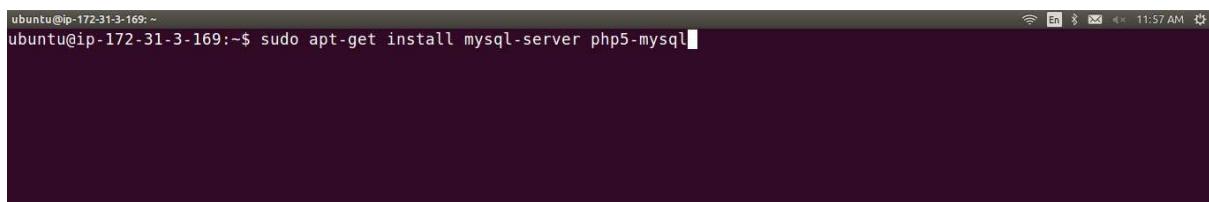
**sudo service apache2 restart**

Congratulations :) PHP is installed on Ubuntu EC2 machine.

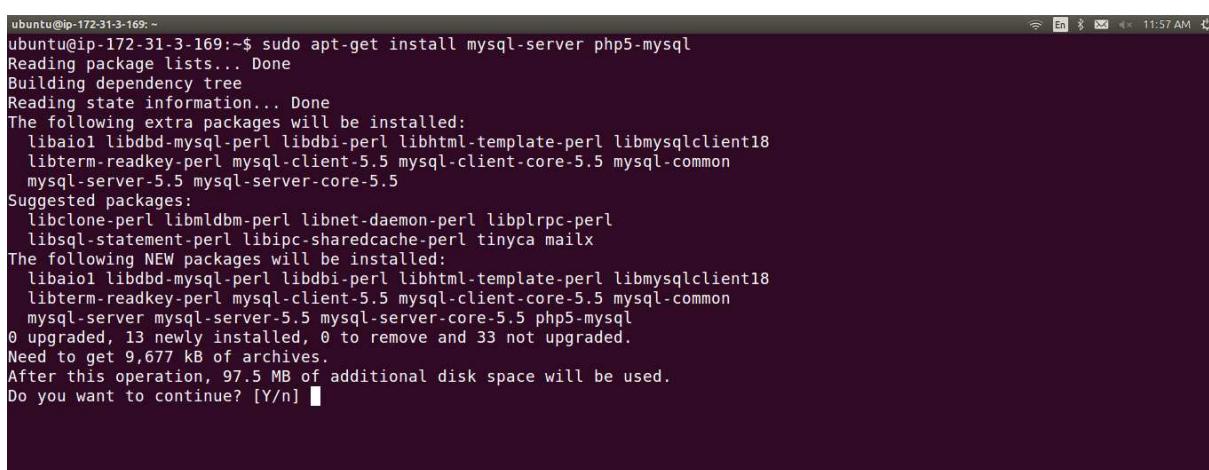
## Configuring MySQL on Ubuntu 14

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

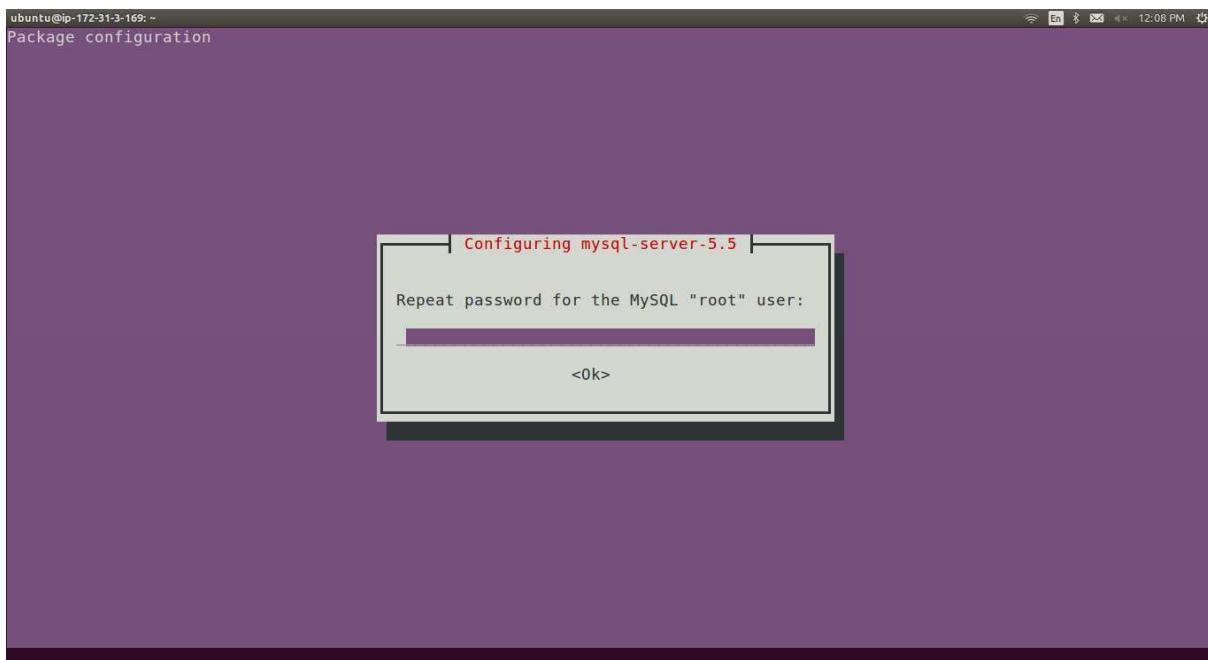
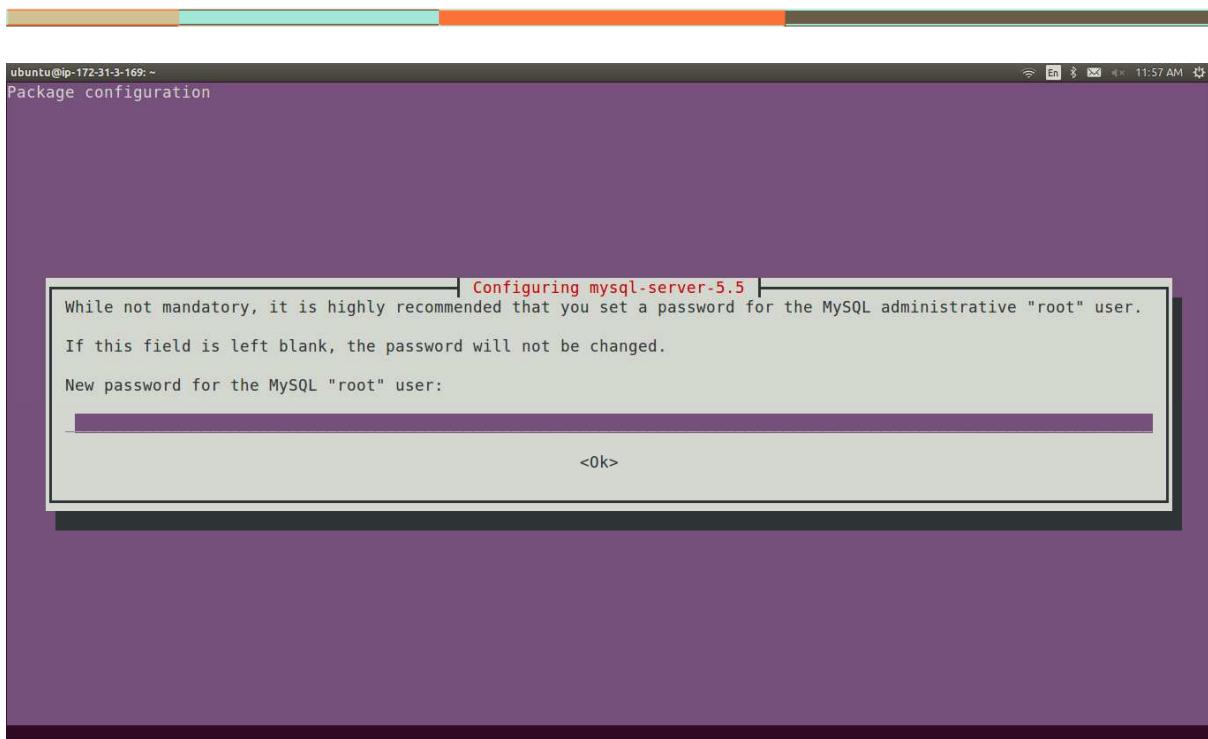
3. Next, MySQL will ask username for root user

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

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

The screenshot shows the AWS Management Console homepage. The left sidebar lists various AWS services under categories like Compute, Storage, and Database. The 'Compute' category is expanded, showing EC2, Lambda, and Batch. The 'Storage' category is also expanded, showing S3, EFS, and Glacier. The 'Database' category is partially visible. On the right, there's a 'Helpful tips' section with links to 'Manage your costs' and 'Create an organization'. Below that is an 'Explore AWS' section with links to 'New Product Announcements' and 'Migrate from Oracle to Amazon Aurora'.

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

The screenshot shows the EC2 Dashboard. The left sidebar has sections for EC2 Dashboard, Instances, Images, and Elastic Block Store. The main area is titled 'Resources' and displays statistics: 1 Running Instances, 0 Dedicated Hosts, 2 Volumes, 1 Key Pairs, 0 Placement Groups, 2 Elastic IPs, 0 Snapshots, 0 Load Balancers, and 3 Security Groups. Below this is a promotional message about Amazon Lightsail. A 'Create Instance' section with a 'Launch Instance' button is present. The right side of the dashboard shows 'Account Attributes' (Supported Platforms: VPC, Default VPC: vpc-bd2d6dd4, Resource ID length management) and 'Additional Information' (Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, Contact Us). At the bottom, there's an 'AWS Marketplace' section and a footer with links to Feedback, English, Privacy Policy, and Terms of Use.

4. Click on Running Instances.

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 (selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security, and 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 Tags tab is active, showing the instance ID and public DNS.

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

The screenshot shows the AWS Instance details page for instance i-0fcfc4fc3afe2f71e. At the top, it displays the instance ID and Public DNS. Below is a table with various instance details. The "Security groups" field contains the value "launch-wizard-1, view inbound rules". The "Tags" tab is active. The URL in the address bar ends with "#SecurityGroups:groupId=sg-906393f8;sort=groupId".

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

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

The screenshot shows the AWS EC2 Management Console interface. The left sidebar navigation includes: EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (Instances, Spot Requests, Reserved Instances, Dedicated Hosts), IMAGES (AMIs, Bundle Tasks), ELASTIC BLOCK STORE (Volumes, Snapshots), and NETWORK & SECURITY (Security Groups, Elastic IPs, Placement Groups, Key Pairs). The main content area displays a table of security groups, with 'sg-906393f8' selected. Below the table, the 'Security Group: sg-906393f8' details are shown, with the 'Inbound' tab selected. The inbound rule listed is:

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0

### 7. Click on Inbound

This screenshot is identical to the previous one, showing the AWS EC2 Management Console with the 'Inbound' tab selected for the security group 'sg-906393f8'. The 'Edit' button is highlighted in the table, indicating the next step in the process.

### 8. Click on Edit

The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with various services like EC2 Dashboard, Instances, Images, and Security Groups. The 'Security Groups' section is currently selected. In the main area, there's a search bar at the top with 'Group ID: sg-906393f8'. 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 in the center. It has fields for Type (SSH), Protocol (TCP), Port Range (22), and Source (Custom, 0.0.0.0/0). At the bottom of the modal, there's an 'Add Rule' button, a note about edits causing traffic drops, and 'Cancel' and 'Save' buttons.

### 9. Click on Add Rule

This screenshot is similar to the previous one, but the 'Add Rule' button in the 'Edit inbound rules' dialog is highlighted with a red box. The 'HTTP' protocol is selected in the dropdown menu on the left side of the dialog. The rest of the interface is identical to the first screenshot.

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. The first row has 'SSH' as the type, 'TCP' as the protocol, '22' as the port range, and 'Custom' (0.0.0.0/0) as the source. The second row has 'HTTP' as the type, 'TCP' as the protocol, '80' as the port range, and 'Anywhere' as the source. An 'Add Rule' button is visible. A note at the bottom of the dialog says: '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.

The screenshot shows the AWS EC2 Management Console displaying the details for a running instance. The instance ID is i-0fcfc4fc3afe2f71e, it is an t2.micro type in us-east-2c zone, and its state is running. The Public DNS is listed as ec2-52-14-202-117.us-east-2.compute.amazonaws.com. The instance has an IPv4 Public IP of 52.14.202.117 and a Private IP of 172.31.43.82. The AMI ID is ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-13291026). Other details shown include Availability zone (us-east-2c), Security groups (launch-wizard-1), and Scheduled events (No scheduled events).

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.



## Setup MySQL on Amazon RDS.

### What is Amazon RDS?

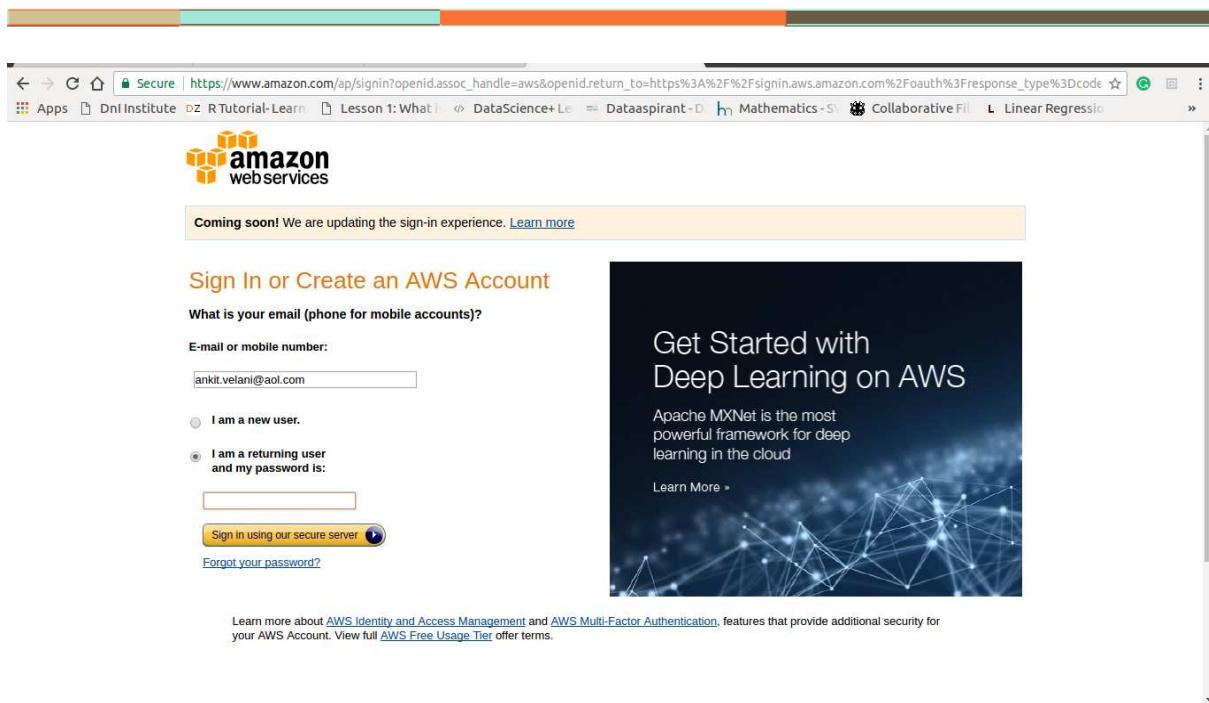
Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

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.

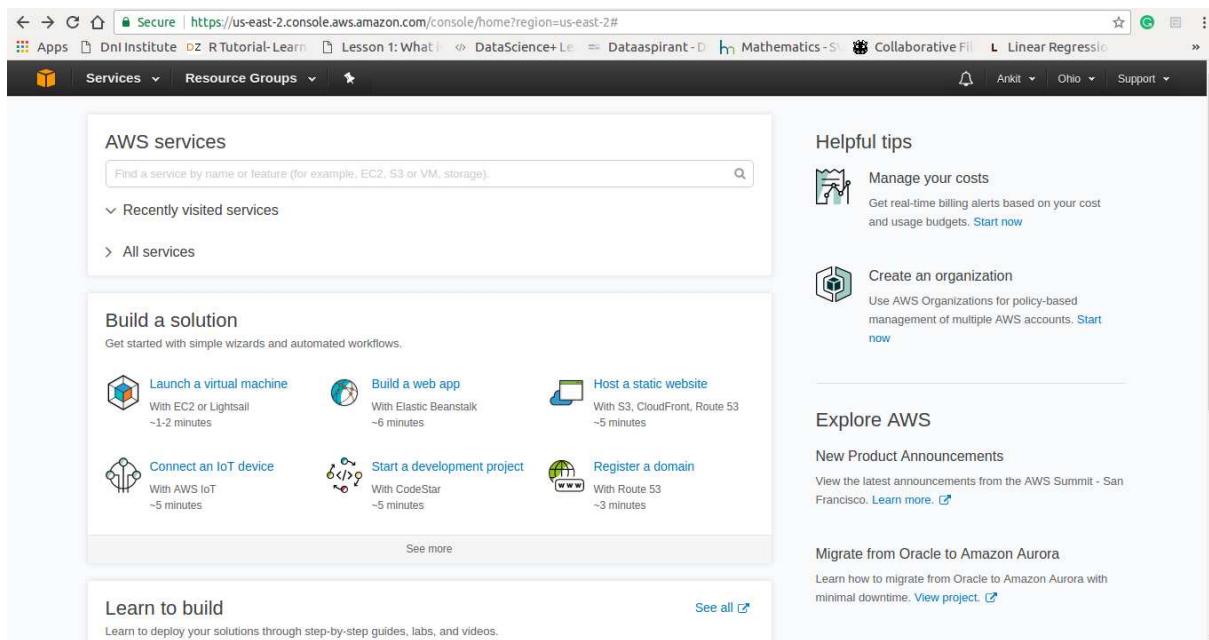
- When you buy a server, you get CPU, memory, storage, and IOPS, all bundled together. With Amazon RDS, these are split apart so that you can scale them independently. So, for example, if you need more CPU, less IOPS, or more storage, you can easily allocate them.
- Amazon RDS manages backups, software 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.



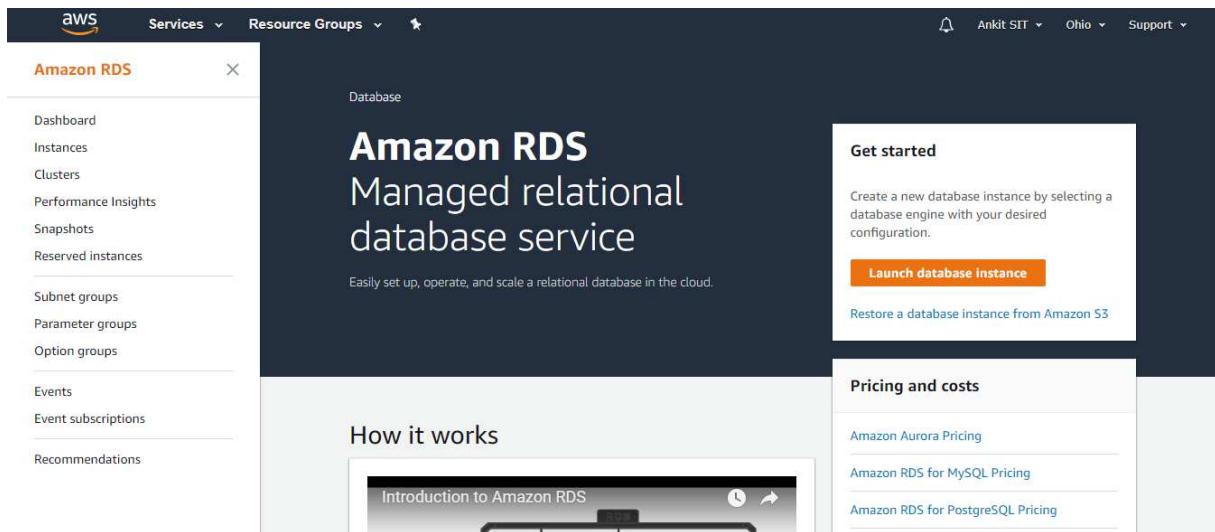
5. Click on All Services ,

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, and Mobile Services. To the right, there's a "Helpful tips" section with links to "Manage your costs", "Create an organization", and "Explore AWS" sections.

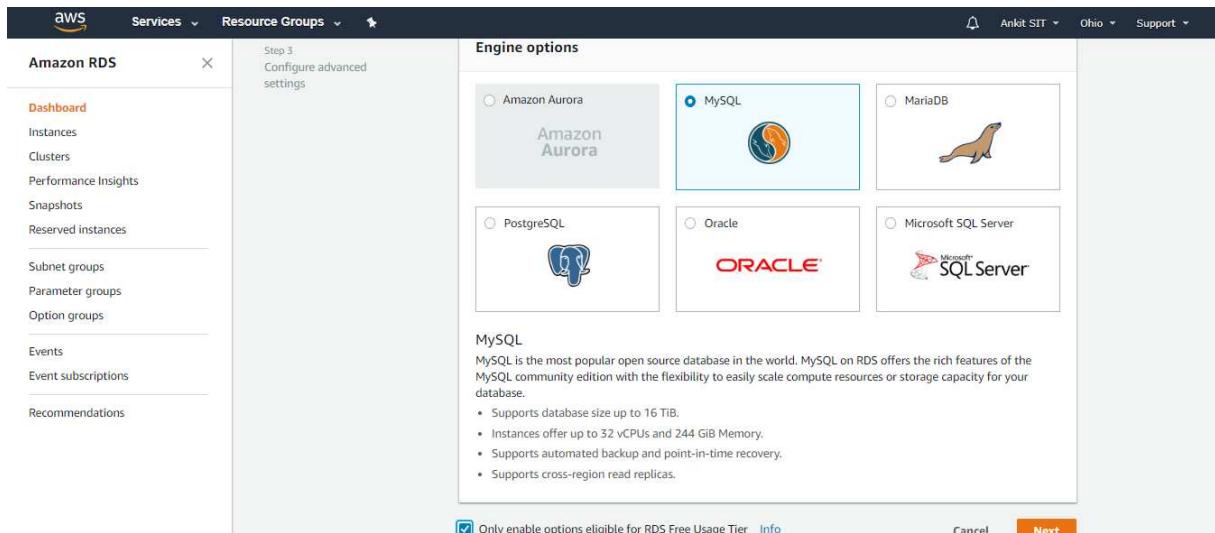
## 6. Choose RDS, from category “ Database”

This screenshot shows a different view of the AWS Management Console homepage, focusing on the "Database" category. It lists services like RDS, DynamoDB, ElastiCache, and Redshift under the "Database" heading. Other service categories visible include Storage, Management Tools, Game Development, Mobile Services, Security, Identity & Compliance, Application Services, Messaging, Analytics, and Business Productivity. The right side features a "Helpful tips" section with links to various AWS services and resources.

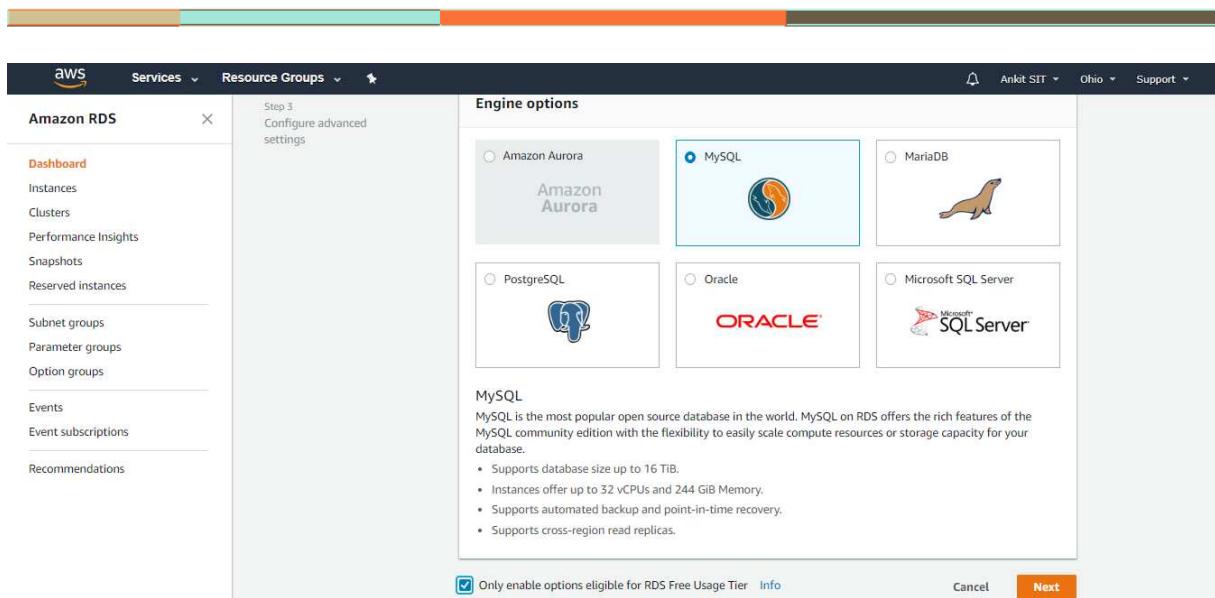
7. Click on RDS.



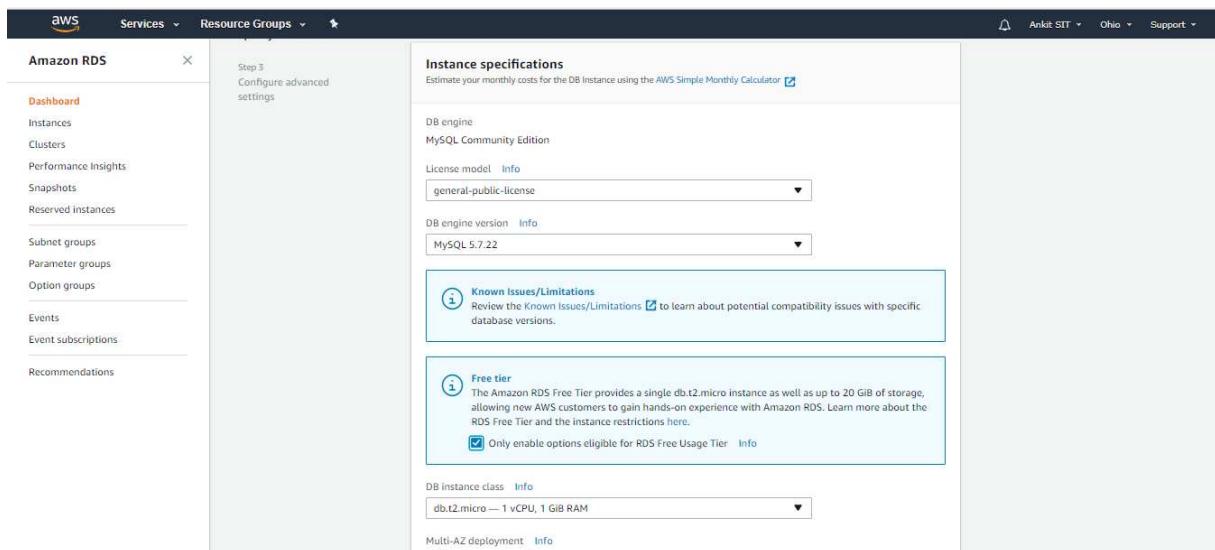
8. Click on Launch database Instance



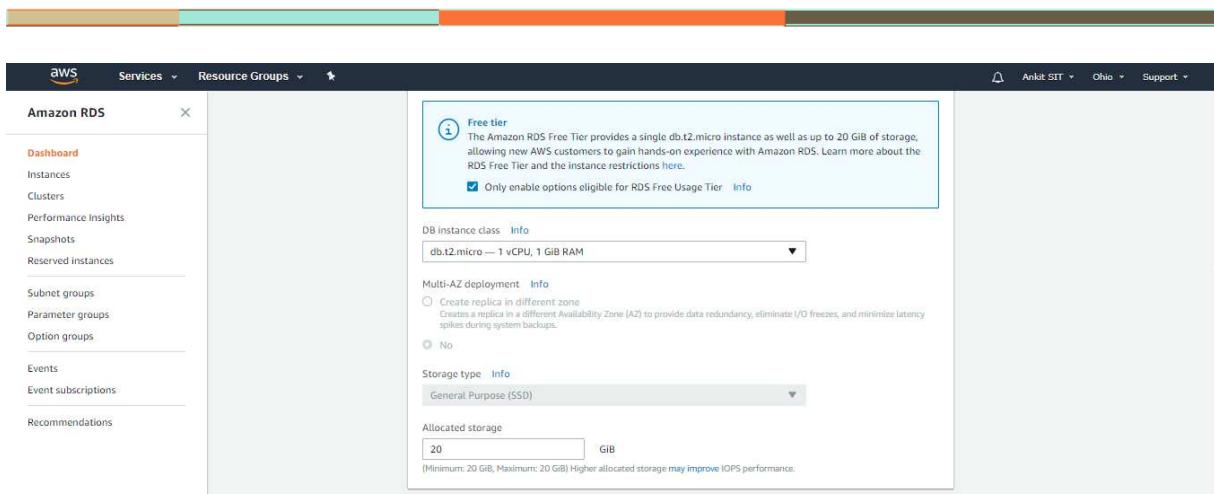
9. Click on Free tier eligible only , Click on MySQL



10. Click on **Next**.



11. Select the Only “show options that are eligible for RDS Free Tier”



## 12. Free tier Instance configuration

## 13. Scroll down the page and specify the database configuration

**Settings**

DB instance identifier [Info](#)  
Specify a name that is unique for all DB instances owned by your AWS account in the current region.

DB instance identifier is case insensitive, but stored as all lower-case, as in "mydbinstance". Must contain from 1 to 63 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Master username [Info](#)  
Specify an alphanumeric string that defines the login ID for the master user.

Master Username must start with a letter. Must contain 1 to 16 alphanumeric characters.

Master password [Info](#) Confirm password [Info](#)

Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "\", or "@".

[Cancel](#) [Previous](#) [Next](#)

## 14. Specify the Database settings

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

Master Username :=====> root

Master Password :=====> mypassword

Confirm Password :=====> mypassword

## 15. Click on Next.

## 16. It allow us to do any advanced configuration.

The image displays three sequential screenshots from the AWS RDS 'Create database' wizard, illustrating the configuration of a new database instance.

**Screenshot 1: Configure advanced settings (Step 3)**

This screen shows the 'Network & Security' section. It includes fields for selecting a VPC (Default VPC (vpc-f9e9dd91)), Subnet group (default), Public accessibility (No), and Availability zone (No preference). It also provides options for creating or choosing VPC security groups.

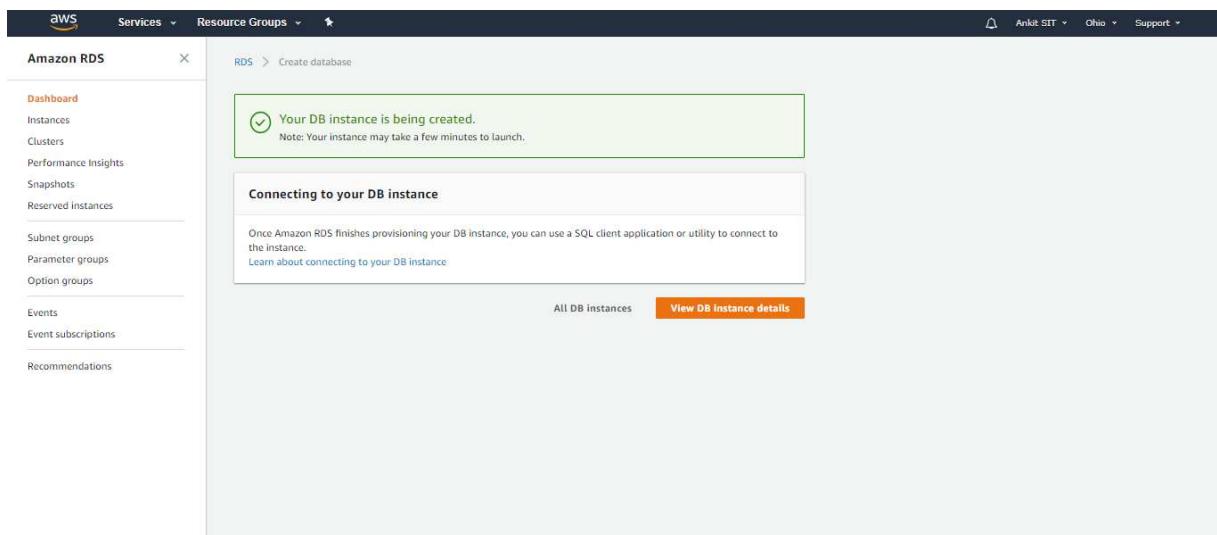
**Screenshot 2: Database options (Step 4)**

This screen shows the 'Database options' section. It includes fields for Database name (db1name), Port (3306), DB parameter group (default.mysql5.7), Option group (default:mysql-5-7), and IAM DB authentication (Disable). It also includes an 'Encryption' section with an 'Encryption' button.

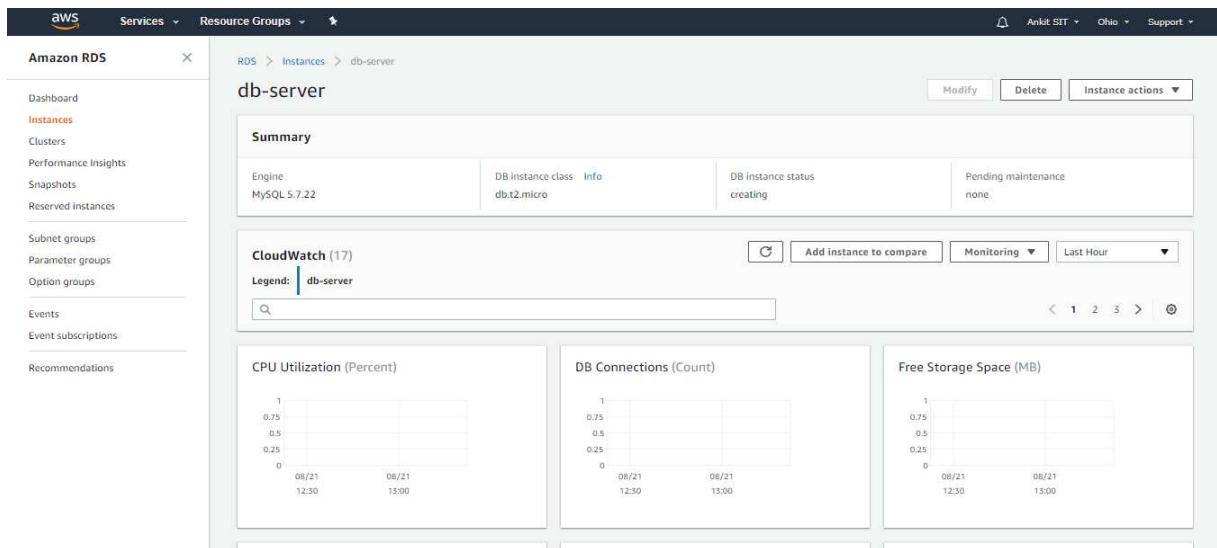
**Screenshot 3: Maintenance (Step 5)**

This screen shows the 'Maintenance' section. It includes fields for Auto minor version upgrade (Enable auto minor version upgrade) and Maintenance window (Select window). It also includes a note about IAM permissions and a 'Create database' button at the bottom.

**17. Click on Create database**



**18. Click to View Your DB Instance.**



Congratulation :) Successfully created MySQL Database on Amazon RDS.

## Working with RDS MySQL

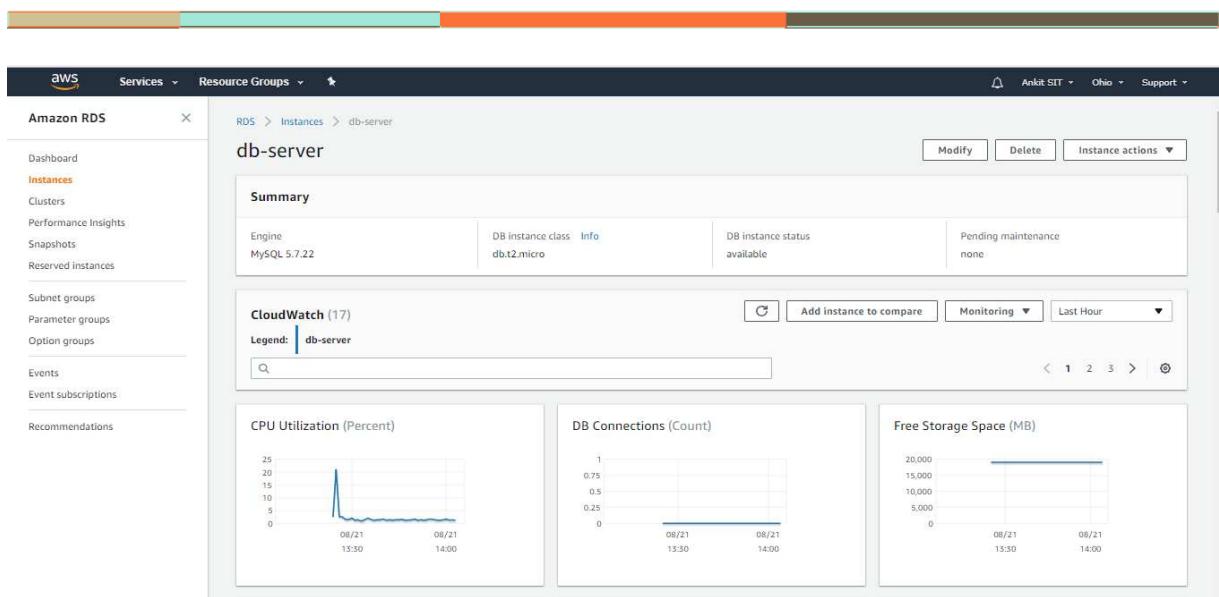
1. Jump to RDS dashboard.

The screenshot shows the AWS RDS MySQL dashboard. On the left, a sidebar lists options like Dashboard, Instances, Clusters, Performance Insights, etc. The main area displays Amazon Aurora information and a summary of RDS resources in the US East (Ohio) region. It shows 1 DB Instance, 33 Snapshots, and various Parameter and Option groups. A prominent orange 'Create database' button is located at the bottom of the main content area.

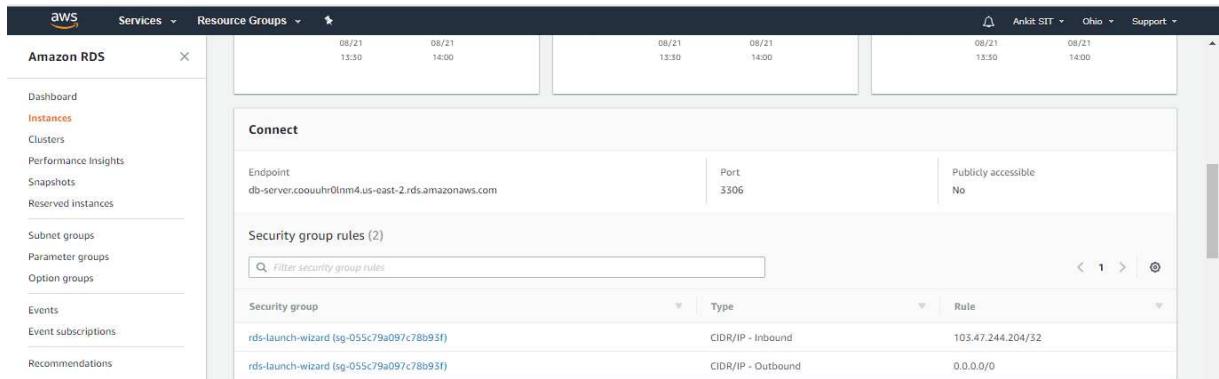
2. Click on DB Instances.

The screenshot shows the AWS RDS Instances page. The sidebar has 'Instances' selected. The main table lists one MySQL instance named 'db-server', which is currently available. The table includes columns for DB instance, Engine, Status, CPU, Current activity, Maintenance, Class, and VPC.

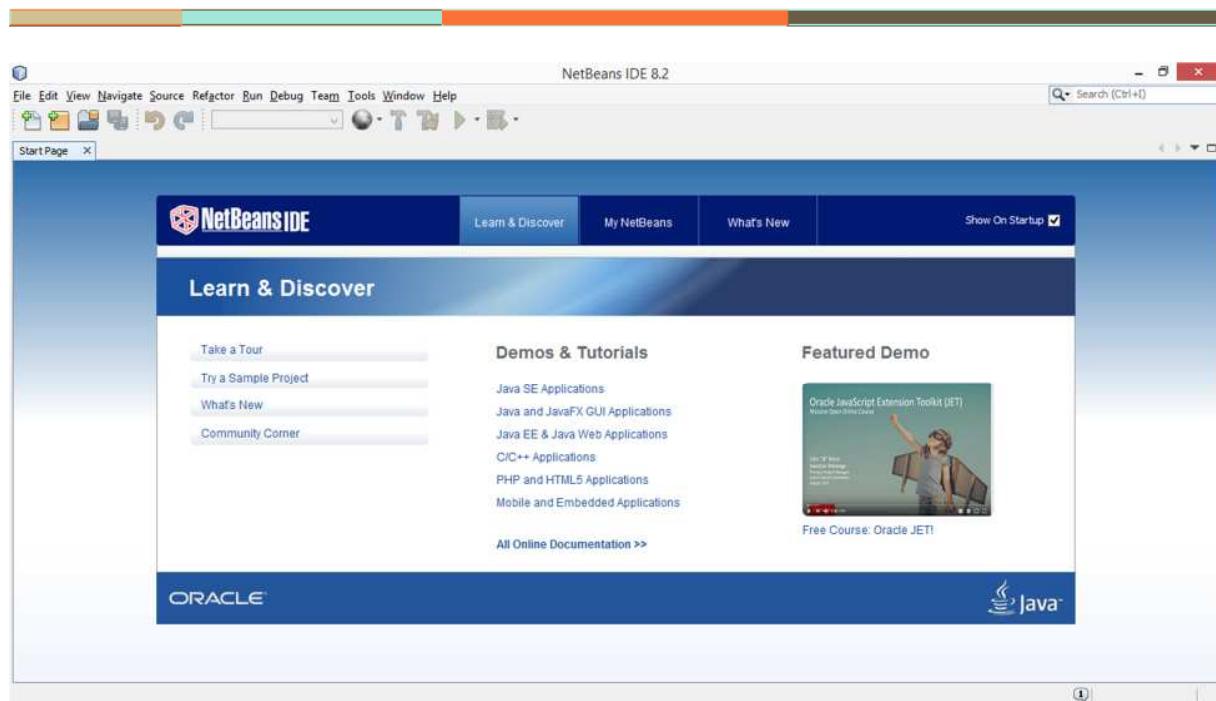
DB instance	Engine	Status	CPU	Current activity	Maintenance	Class	VPC
db-server	MySQL	available	1.19%	0 Connections	none	db.t2.micro	vpc-f9e9da9



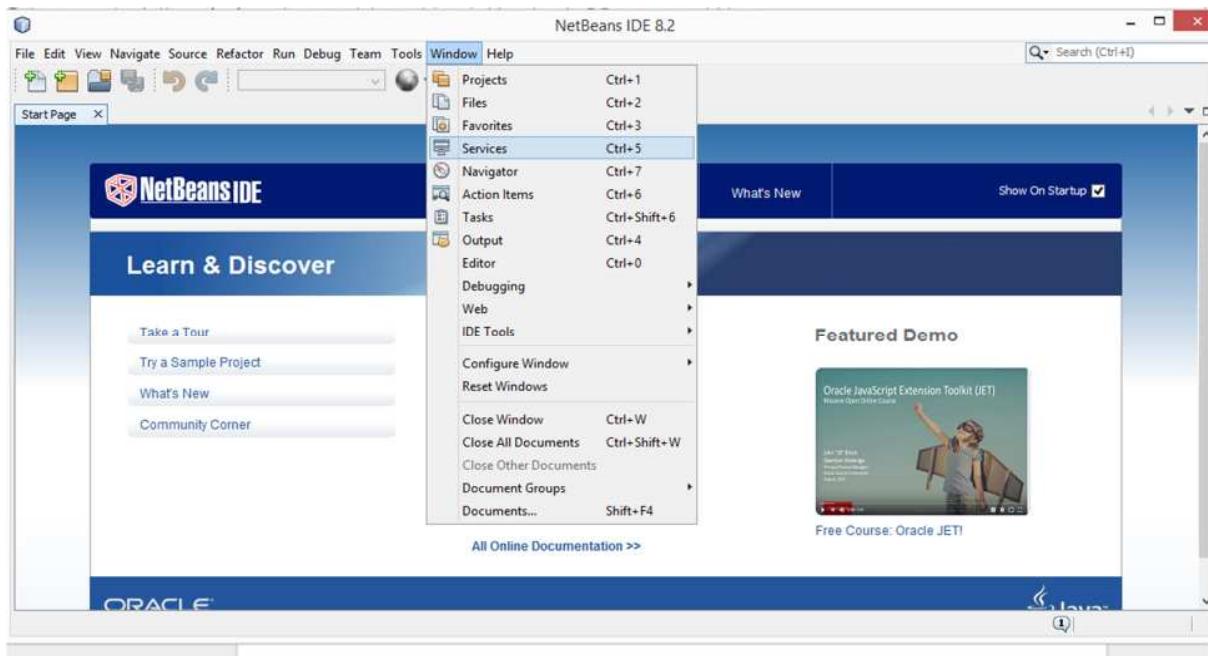
### 3. Endpoint of the DB Instances help us to access MySQL.



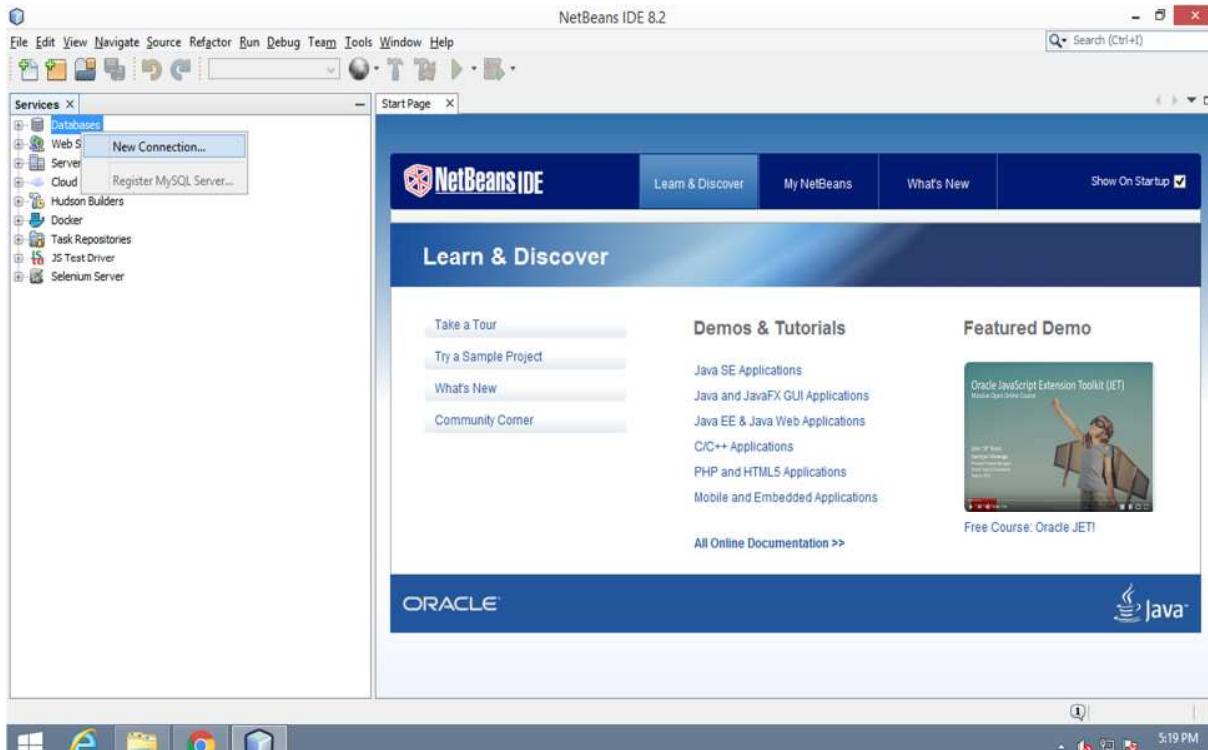
4. We can use this Endpoint to connect with MySQL Database in any programming language or tools like MySQL Workbench, SqlWorkbenchJ, in NetBeans or Eclipse..Etc.
5. We will try to connect with NetBeans. Open **NetBeans**.



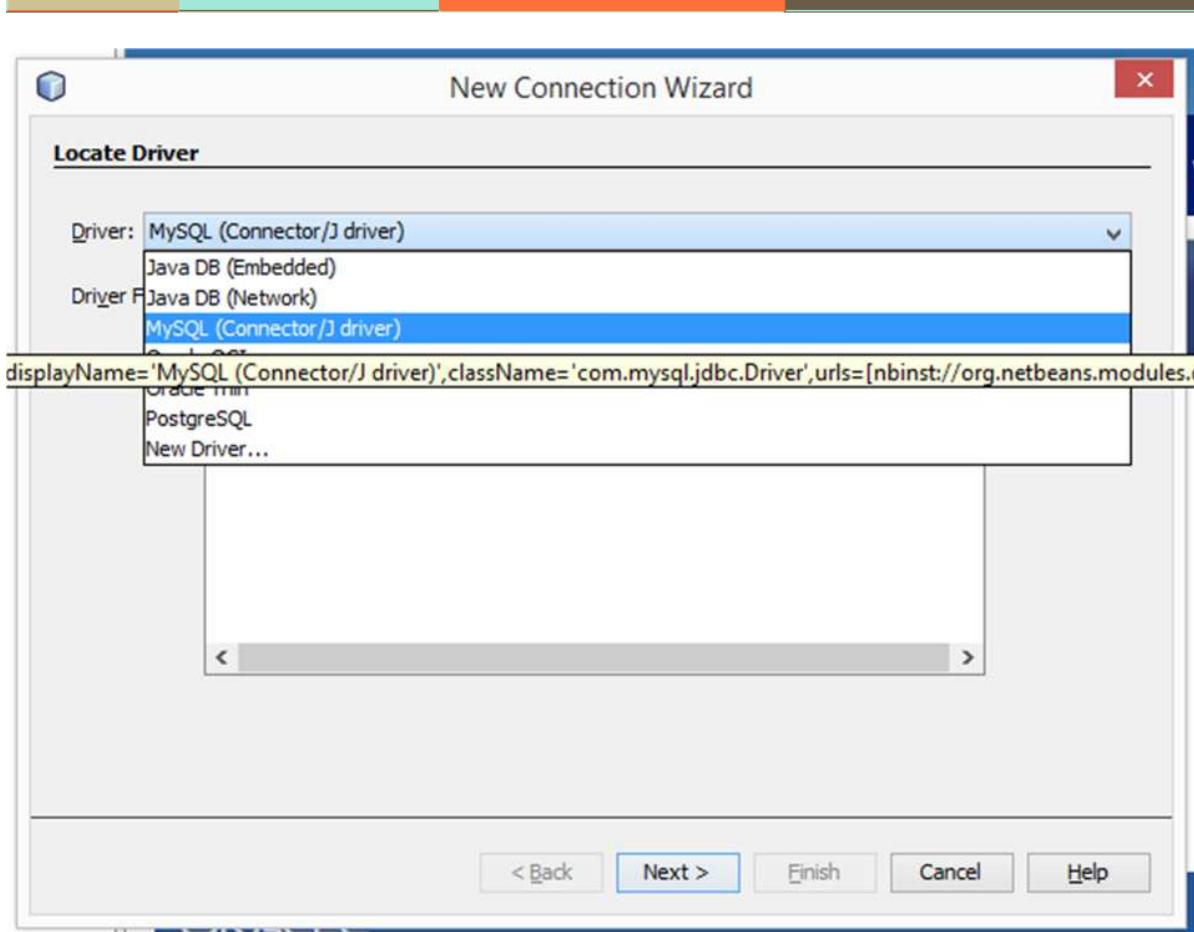
6. Go to Windows > Services



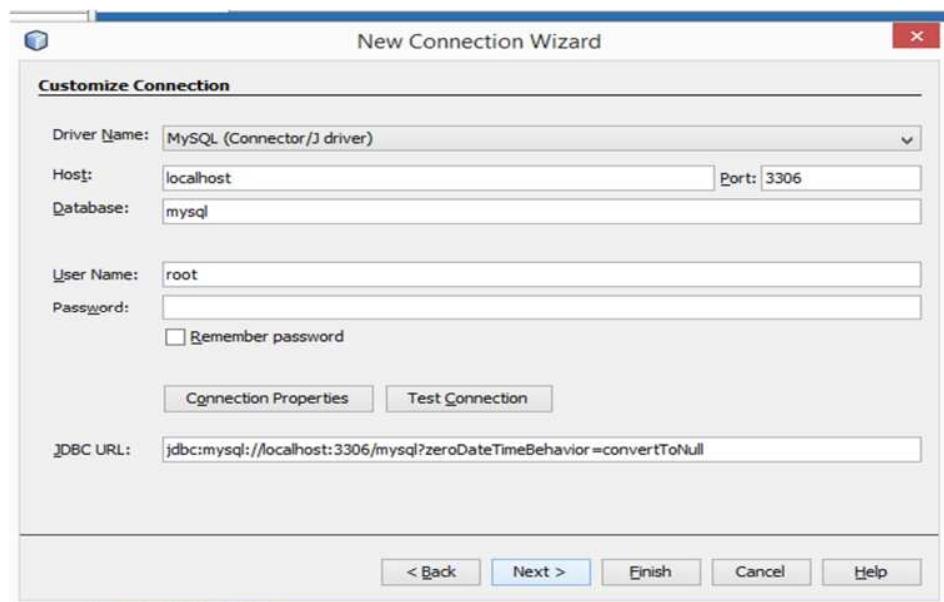
7. Right click on Databases > New Connection.



8. Select MySQL Driver from dropdown.



9. Click on Next.



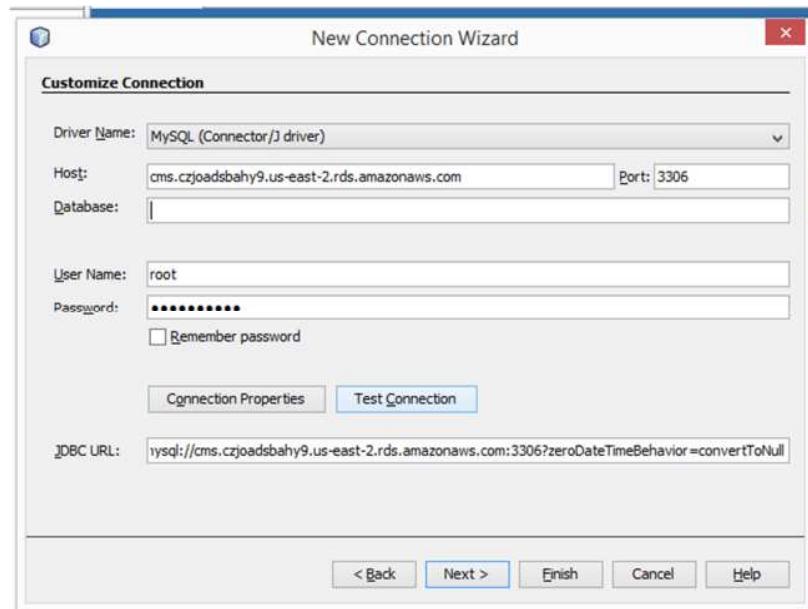
10. Provide connection details.

Host : Paste **Endpoint** of MySQL RDS.

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

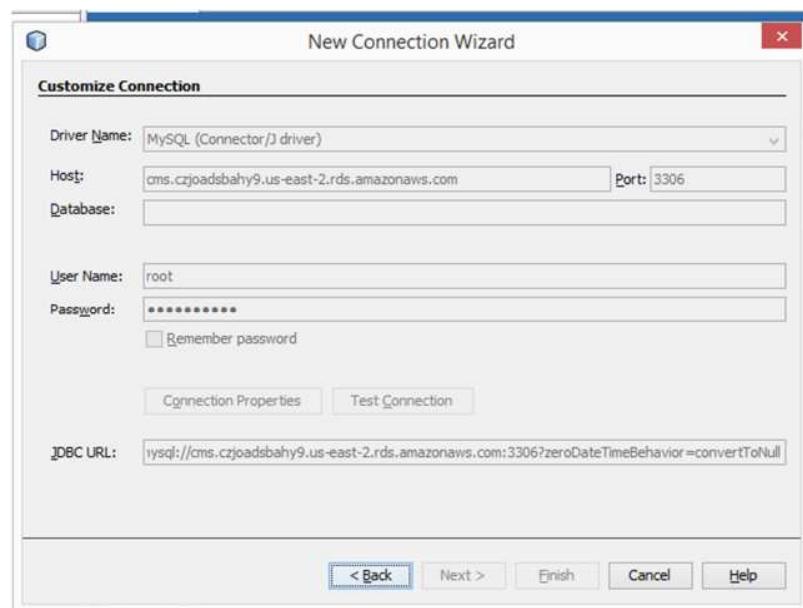
Username: Database **Username (root)**

Password: Database **Password (mypassword)**

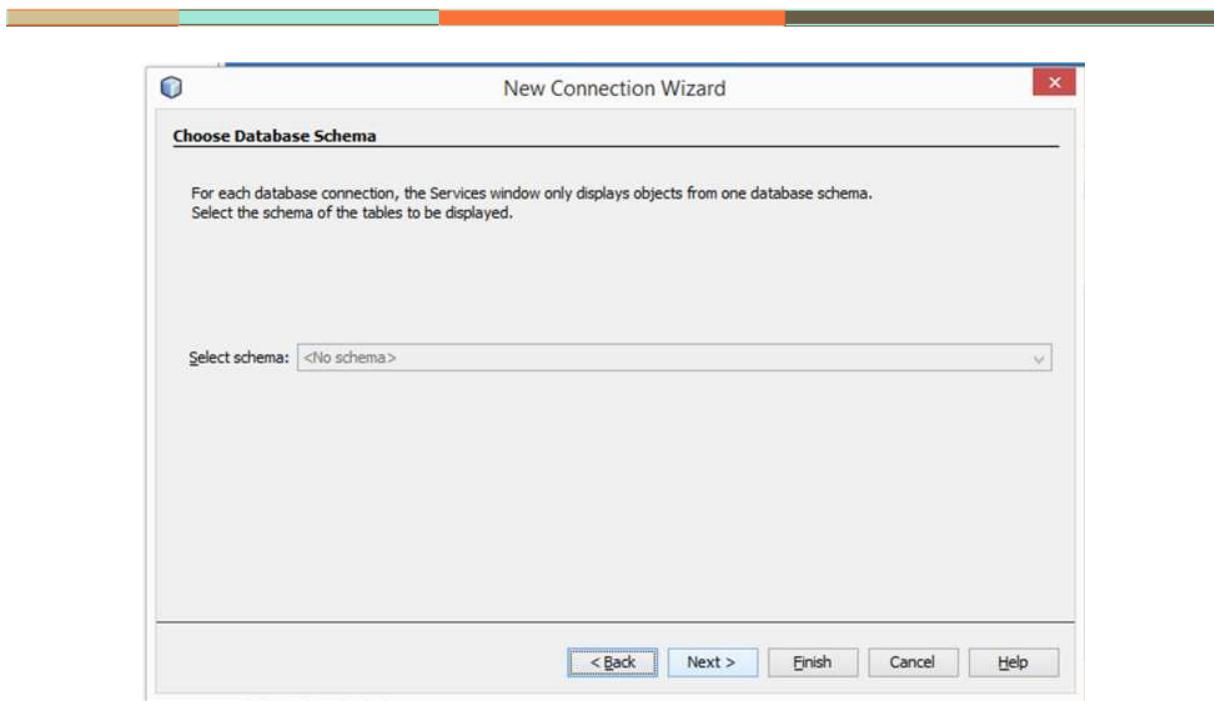


11. Click on Test Connection.

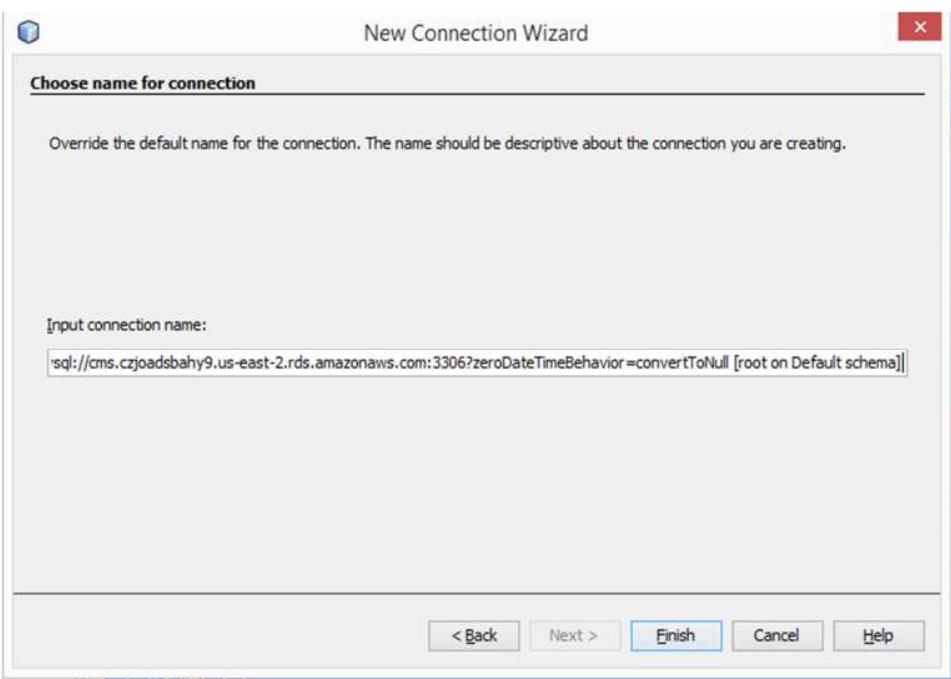
12. If Connection is successfully then click on Next.



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

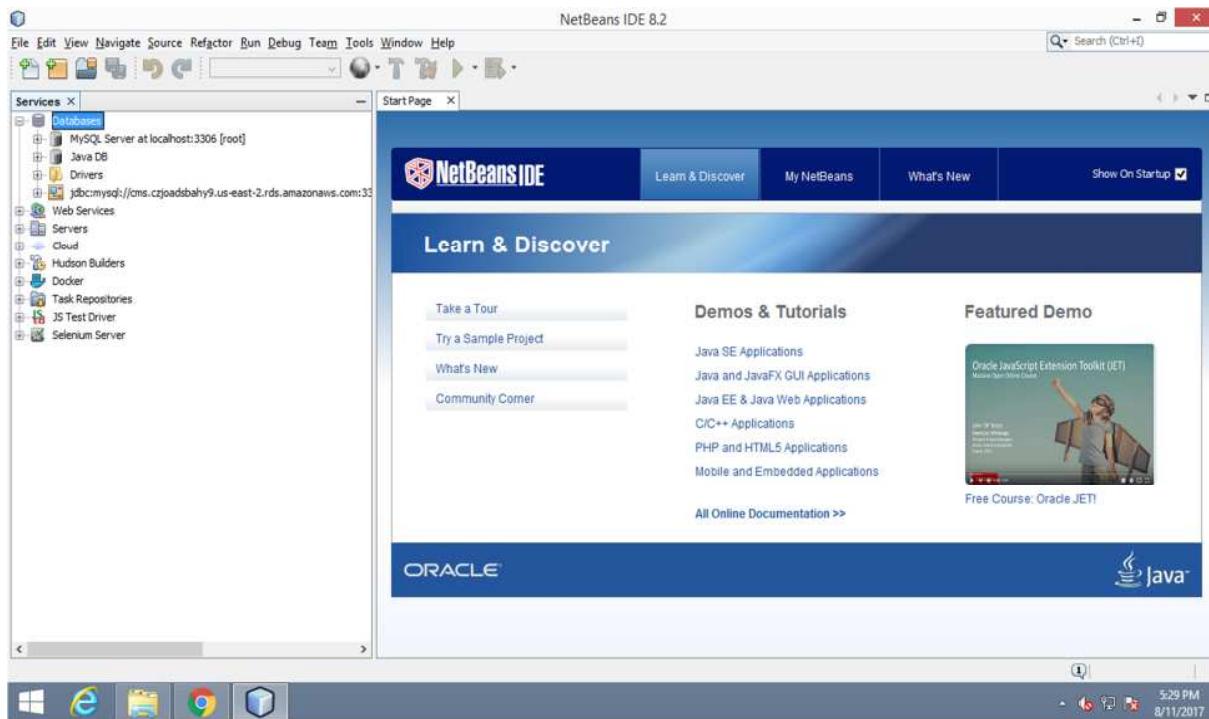


14. Click on Next.

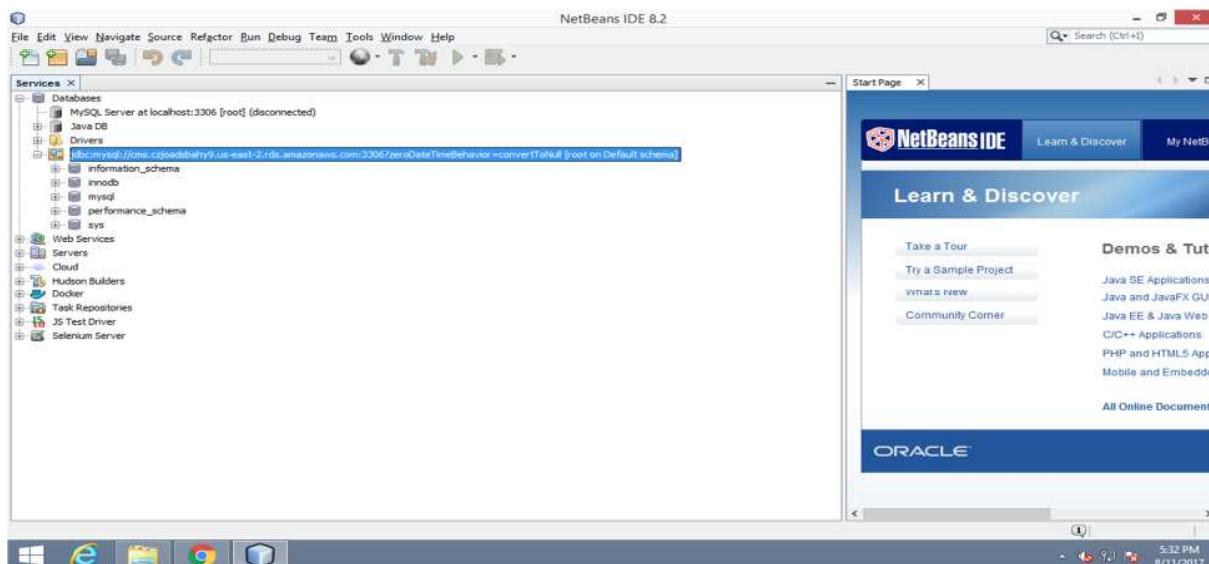


15. Click on Finish.

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.

## Resource Person Profile



### Ankit Velani

Data Scientist

Motivated Data Scientist with the experience of 3 years in building data driven solutions; with a background of Software development, data profiling, and statistical modeling, willing to learn new technologies, Commendable skills in statistical languages, machine learning. Ability to work in different work environments and provide data-driven solutions.

[ankit.velani@aol.com](mailto:ankit.velani@aol.com)

9428307478

Bangalore, India

ankit.velani

[www.linkedin.com/in/ankitvelani](http://www.linkedin.com/in/ankitvelani)

#### WORK EXPERIENCE

##### Data Scientist

Trendwise Software Solutions

04/2015 – Present

Bangalore

Achievements/Tasks

- Worked with clients both domestic and International to solve and provide data driven solutions.
- Worked on project Business Intelligence for a Learning Management Center; Analyzed the current complexities in the client's process and provided the data driven solutions.
- Used Text mining techniques to analyze the Call Center Data; Found out the key problems in products of the client company.
- Performed Sentiment Analysis to find out a quality of a Customer call; used Audio Data of a call center.
- Used Business rules techniques to score inquiry from clients; will be used by sales team for business expansion.

##### Software Engineer

Trendwise Software Solutions

11/2014 – 03/2015

Bangalore

Achievements/Tasks

- Worked with Canadian client to build Multi Comment; A Multi Language Feed Back Web Application.
- Designed a dashboard to provide insight of key customer complaints; Company will use this useful insight to improve customer experience.
- Designed the REST API for the Android application.
- Designed Responsive Web Interface for an internal project.

#### EDUCATION

##### Master Of Computer Application

Siddaganga Institute of Technology

01/2011 – 07/2014

Tumkur, Karnataka

##### O Level

National Institute of Electronics and Information Technology (NIELIT)

06/2010 – 07/2014

New Delhi

#### SKILLS

R    Python    Machine Learning    Data Mining  
 Tableau    NLP    SQL    Bluemix    AWS Cloud  
 Google Cloud    Azure ML Studio    PHP

#### PUBLICATION & TECHNICAL TALK

Conducted workshop in 3 Engineering colleges with a participation of 600 students. (06/2016 – Present)

Workshops were aimed to provide hand on experience in Data Exploratory Analysis, Regression, Classification, Clustering and Text Mining using R.

National level paper presented on "Pragmatic Study of LEACH and T-LEACH Protocols of Wireless Sensor Network".

In this paper, the study of LEACH (Low-Energy Adaptive Clustering Hierarchy), a clustering-based protocol shows that utilization of randomized rotation of cluster-heads to evenly distribute the energy load among the sensors in the network.

National level paper presented on "Social media analytics: A survey of analysis conducted on Social Media".

In this paper, we present a pragmatic study of the different kinds of behavioral studies conducted on Social Networking Sites(such as Facebook, Twitter, LinkedIn etc.).

##### Project Mentor

ICT incubation Center (2015-16) student project in collaboration with Siddaganga Institute of Technology Tumkur titled "Tumkur Mobile value added service".

#### CERTIFICATES

Cloud Computing (09/2015 – 09/2015)

Data Analytics using R (03/2016 – 03/2016)

Data Manipulation at Scale: Systems and Algorithms (10/2015 – 02/2016)   
*Coursera / University of Washington*