

# Hands on with Cloud Computing

12-August-2017

---

Created by :  
Ankit Velani  
Data Scientist,  
Trendwise Software Solution,Bangalore.

Submitted at :  
Department Of MCA  
Siddaganga Institute of Technology,Tumkur.

## Overview

This document contains steps to set up various cloud services on the Amazon Web Service(AWS). Document divides in various section, Create Amazon account, Setup EC-2 on Amazon, Accessing Amazon EC2 machine using FileZilla and SSH, Installing Softwares (PHP,MySQL,R) on EC-2 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/>

AWS Free Tier

The AWS Free Tier enables you to gain free, hands-on experience with the AWS platform, products, and services.

Create a Free Account

Free Tier Details      Get Started      Free Tier Software

**AWS Free Tier Details**

★ FEATURED    ○ 12 MONTHS FREE    ○ ALWAYS FREE    ▾ PRODUCT CATEGORIES    ▾ ALL

COMPUTE    STORAGE & CONTENT DELIVERY    DATABASE

2. Click on *Create a Free account*

Coming soon! We are updating the sign-in experience. [Learn more](#)

Sign In or Create an AWS Account

What is your email (phone for mobile accounts)?

E-mail or mobile number:

I am a new user.

I am a returning user and my password is:

[Sign in using our secure server](#)

[Forgot your password?](#)

AWS Accounts Include  
12 Months of Free Tier Access

Including use of Amazon EC2,  
Amazon S3, and Amazon DynamoDB

Learn more about [AWS Identity and Access Management](#) and [AWS Multi-Factor Authentication](#), features that provide additional security for your AWS Account. View full [AWS Free Usage Tier](#) offer terms.

3. Choose option , Enter the **Email** , choose the option '**I am new user**' if you already have an account then choose option ' I am returning user and my password is '.
4. Click on **Sign in using our Secure Server**.
5. Fill your personal information.

The screenshot shows the 'Contact Information' step of the AWS sign-up process. At the top, there are two radio button options: 'Company Account' and 'Personal Account'. Below these are several required fields: 'Full Name\*', 'Country\*' (set to United States), 'Address\*', 'City\*', 'State / Province or Region\*', 'Postal Code\*', and 'Phone Number\*'. A 'Security Check' section contains a CAPTCHA image showing 'm6e24e' and a text input field for re-typing it. Below the CAPTCHA is the 'AWS Customer Agreement' checkbox, which is checked, followed by a link to the AWS Customer Agreement terms. A 'Create Account and Continue' button is at the bottom.

6. Payment Verification. Amazon will verify your credit/debit card with minimal amount which is Rs. 2 , will be refund in 3-5 business day.

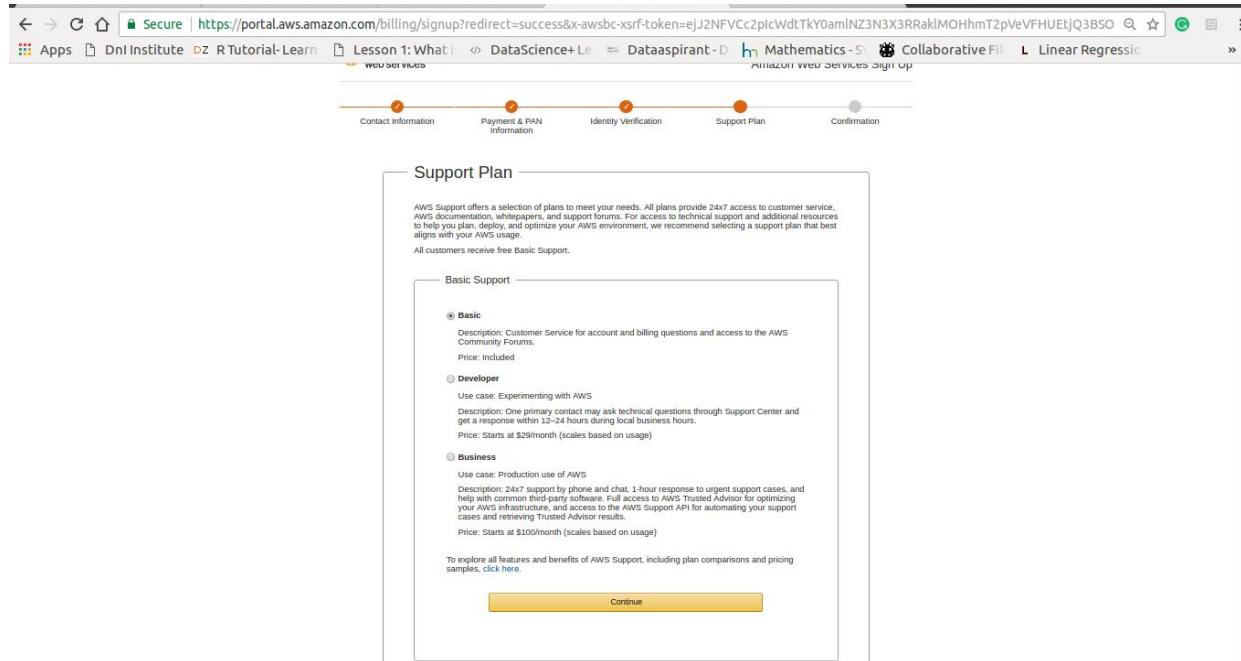
The screenshot shows the 'Payment Information' step of the AWS sign-up process. It features a progress bar with five steps: 'Contact Information', 'Payment & PAN Information', 'Identity Verification', 'Support Plan', and 'Confirmation'. The 'Payment & PAN Information' step is active. It includes fields for 'Cardholder's Name' (with an error message: 'Your card could not be successfully verified. You can retry your card or add a different one.'), 'Credit/Debit Card Number' (with validation messages: 'Cardholder's Name is a required field' and 'Credit/Debit Card Number is a required field'), 'Expiration Date' (set to 07 2017), and 'Address' (set to Bangalore Bangalore Karanataka 560037 IN). There is also a note about a \$2 charge and a 'Please Note' section at the bottom.

## 7. Next Step, Amazon will verify the mobile number.

The screenshot shows the 'Identity Verification' step of the Amazon Web Services Sign Up process. At the top, a progress bar shows five steps: Contact Information, Payment & PAN Information, Identity Verification (which is highlighted with a red dot), Support Plan, and Confirmation. The main area is titled 'Identity Verification' and contains instructions: 'You will be called immediately by an automated system and prompted to enter the PIN number provided.' Below this, there's a 'Security Check' section with a CAPTCHA image showing 'Xco 3 fc' and a text input field for re-typing it. There are dropdowns for 'Country Code' (set to India (+91)) and 'Phone Number' (set to 9428307478), and an empty 'Ext' field. A large yellow 'Call Me Now' button is centered. Below the form, three numbered steps are listed: 1. Provide a telephone number, 2. Call in progress, and 3. Identity verification complete.

This screenshot shows the same 'Identity Verification' page, but the process has been completed. The main area now displays a success message: 'Your identity has been verified successfully.' Below this, a large yellow 'Continue to select your Support Plan' button is visible. The progress bar at the top remains the same, indicating the user is still on the 'Identity Verification' step.

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



9. Chose the Basic plan.

10. Click on Continue.

11. 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>
2. Enter registered email and choose option 'I am returning user and my password is:'.
3. Click on Sign in using our secure server.

Coming soon! We are updating the sign-in experience. [Learn more](#)

**Sign In or Create an AWS Account**

What is your email (phone for mobile accounts)?

E-mail or mobile number:

ankit.velania@aol.com

I am a new user.

I am a returning user and my password is:

**Sign in using our secure server**

[Forgot your password?](#)

Get Started with Deep Learning on AWS

Apache MXNet is the most powerful framework for deep learning in the cloud

[Learn More](#)

Learn more about [AWS Identity and Access Management](#) and [AWS Multi-Factor Authentication](#), features that provide additional security for your AWS Account. View full [AWS Free Usage Tier](#) offer terms.

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

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

Apps DnI Institute R Tutorial-Learn Lesson 1: What is DataScience+ Dataaspirant Mathematics Collaborative File Linear Regression

Services Resource Groups

AWS services

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

Recently visited services

All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine With EC2 or Lightsail ~1 minutes	Build a web app With Elastic Beanstalk ~6 minutes	Host a static website With S3, CloudFront, Route 53 ~5 minutes
Connect an IoT device With AWS IoT ~5 minutes	Start a development project With CodeStar ~5 minutes	Register a domain With Route 53 ~3 minutes

See more

Learn to build

Learn to deploy your solutions through step-by-step guides, labs, and videos.

See all

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](#)

## 5. Click on All Services ,

The screenshot shows the AWS Management Console home page. In the top navigation bar, there are tabs for 'Services' and 'Resource Groups'. On the left, there's a sidebar titled 'AWS services' with sections for 'Recently visited services' and 'All services'. Under 'All services', there are three main categories: 'Compute', 'Storage', and 'Database'. Each category has a list of services. To the right of the sidebar, 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'.

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

## 6. Choose EC2 , from category " Compute "

This screenshot is similar to the previous one, showing the AWS Management Console home page. The 'Compute' category under 'All services' is now highlighted in orange, and its sub-services (EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, Batch) are listed below it. The rest of the interface, including the 'Helpful tips' and 'Explore AWS' sections, remains the same.

## 7. It jumped to EC2 Dashboard. Click on [Launch Instance](#)

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with links like 'EC2 Dashboard', 'Instances', 'AMIs', 'EBS', 'Network & Security', etc. The main area has a heading 'Resources' and a summary of resources: 0 Running Instances, 0 Elastic IPs, 0 Dedicated Hosts, 0 Volumes, 0 Snapshots, 0 Key Pairs, 0 Load Balancers, 1 Security Groups, and 0 Placement Groups. Below this is a callout box with the text: '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.' In the center, there's a 'Create Instance' section with a 'Launch Instance' button. To the right, there are sections for '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), and 'AWS Marketplace' (Find free software trial products). At the bottom, there are links for Feedback, English, and some legal notices.

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

## 9. Select AMI type [Ubuntu Server 14.04 LTS \(HVM\), SSD Volume Type - ami-019abc64](#) and click on Select.

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

The screenshot shows the 'Choose Instance Type' step of the Launch Instance Wizard. At the top, there are tabs for '1. Choose AMI', '2. Choose Instance Type' (which is selected), '3. Configure Instance', '4. Add Storage', '5. Add Tags', '6. Configure Security Group', and '7. Review'. Below this, there's a section titled 'Step 2: Choose an Instance Type' with a paragraph about instance types. There are filters for 'All instance types' and 'Current generation'. A table lists various instance types, with 't2.micro' highlighted as 'Free tier eligible'. The table columns include Family, Type, vCPUs, Memory (GiB), Instance Storage (GB), EBS-Optimized Available, Network Performance, and IPv6 Support. At the bottom, there are buttons for 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Instance Details'.

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.

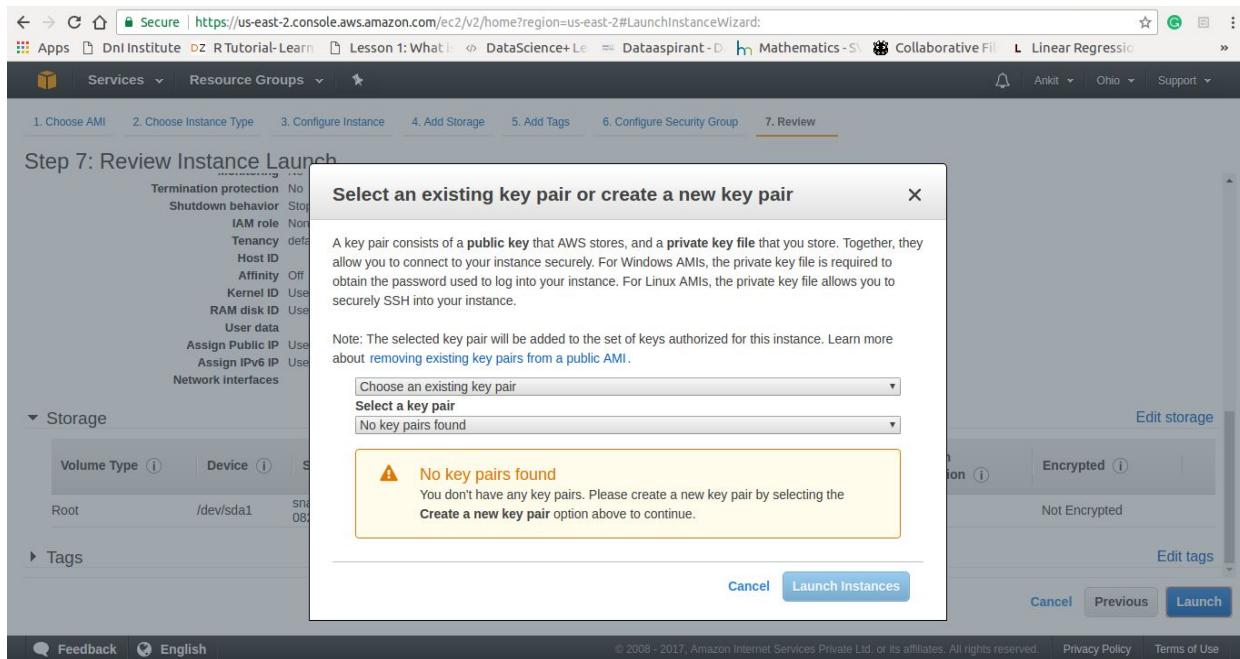
The screenshot shows the AWS Launch Instance Wizard at Step 7: Review Instance Launch. The URL in the browser is <https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard>. The page displays the following configuration:

- AMI Details:** Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-019abc64 (Free tier eligible)
- Instance Type:** t2.micro (Edit instance type)
- Security Groups:** launch-wizard-1 (Edit security groups)

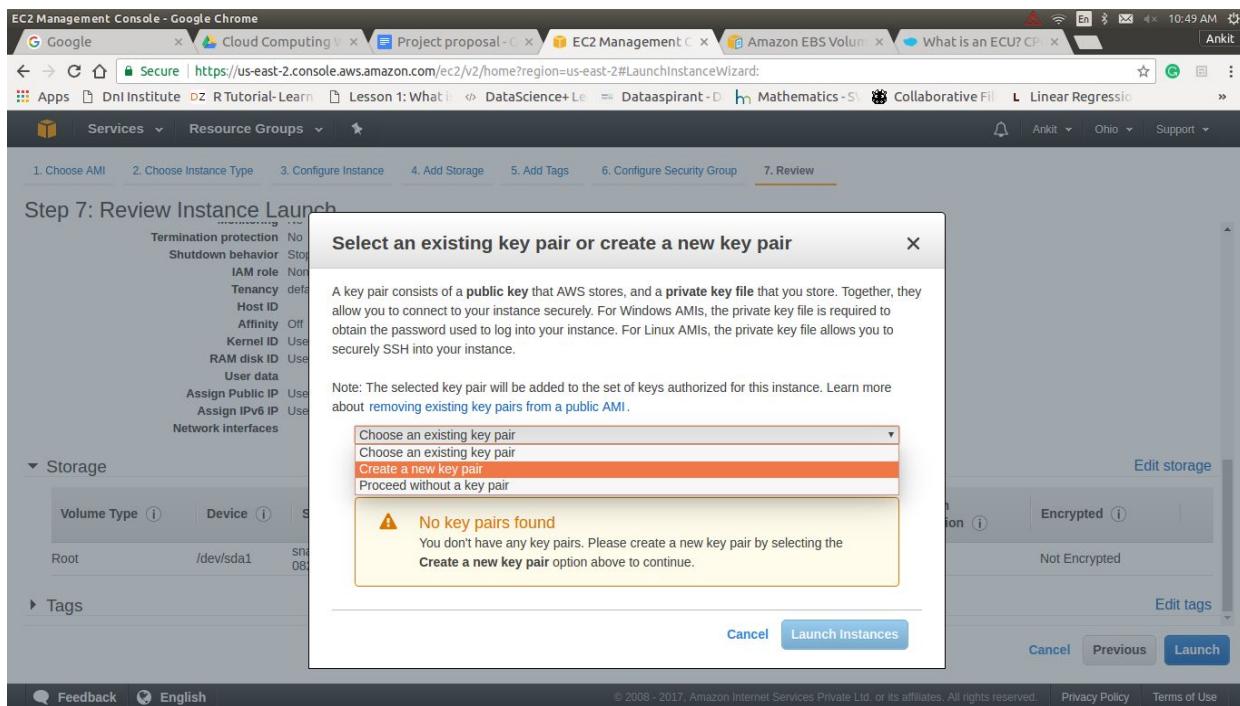
At the bottom right, there are buttons for **Cancel**, **Previous**, and a large blue **Launch** button.

12. Click on Launch.

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



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



## 15. 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

## 16. 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](#)

17. Click on **View Instances**.

18. Back to EC2 Dashboard , under **INSTANCES**, recently created instance will appear.

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 (which is expanded to show 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 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 row is shown in the table, representing an instance named i-0ad2d2f2e17281327, which is a t2.micro type running in us-east-2c with a public DNS of ec2-52-14-56-150.us-east-2.compute.amazonaws.com and a private IP of 172.31.32.230. At the bottom of the page, there are links for Feedback, English, Privacy Policy, and Terms of Use.

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.

This screenshot is identical to the one above, showing the same EC2 Instances page. The instance details are identical: i-0ad2d2f2e17281327, t2.micro, us-east-2c, running, with public DNS ec2-52-14-56-150.us-east-2.compute.amazonaws.com and private IP 172.31.32.230. The "Description" tab is selected, providing a detailed breakdown of the instance configuration, including Instance ID, Instance state, Instance type, Availability zone, 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), and Network interfaces (eth0, Source/dest. check True). The rest of the page structure is also identical to the first 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 is free-tier.

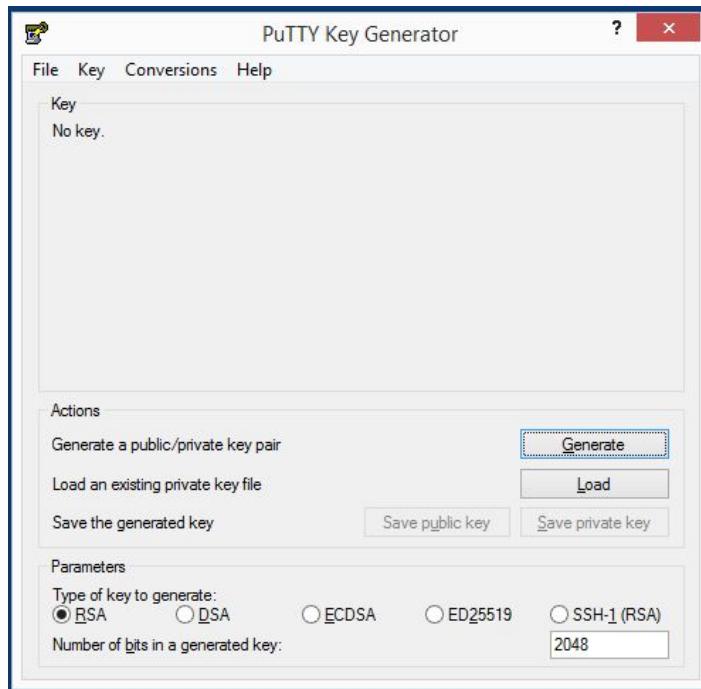
**Step 1: Choose an Amazon Machine Image (AMI)**

Windows	Description	Select	Root device type	Virtualization type
Free tier eligible	Microsoft Windows Server 2016 Base Nano - ami-0994b56c	64-bit	ebs	hvm
Free tier eligible	Microsoft Windows Server 2012 R2 Base - ami-ceaa8bb8	64-bit	ebs	hvm
Free tier eligible	Microsoft Windows Server 2012 Base - ami-ffae8f9a	64-bit	ebs	hvm
Free tier eligible	Microsoft Windows Server 2008 R2 Base - ami-37ab8a52	64-bit	ebs	hvm
	Microsoft Windows Server 2003 R2 Base - ami-82af8ee7 (64-bit) / ami-47a68722 (32-bit)	Select		

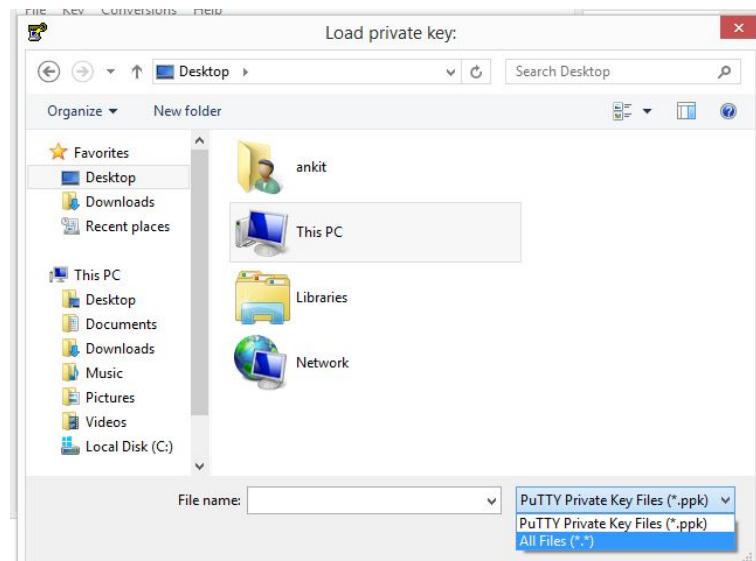
## Converting Encrypted file to ppk ( Private Key) to access EC2

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

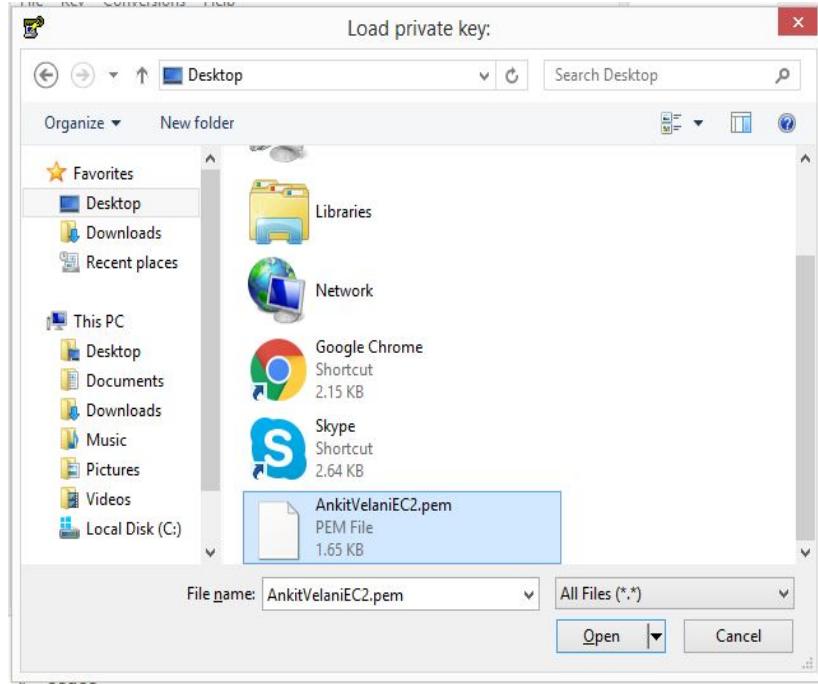
1. Open PuTTygen.



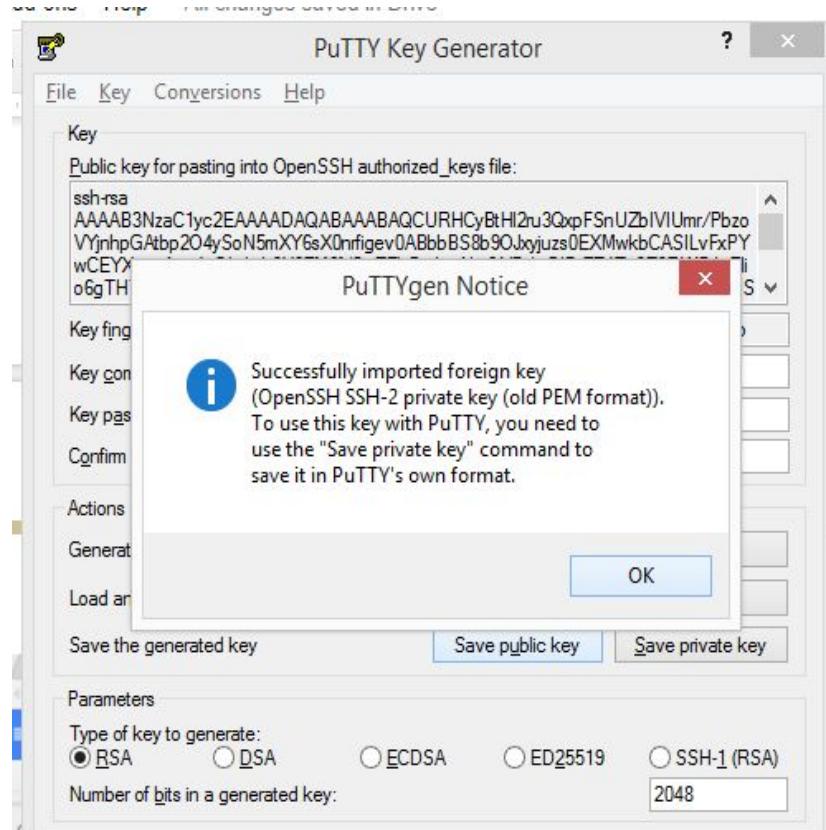
2. Click on **Load** , It gives FileBox , choose file type as **All Files(\*)**



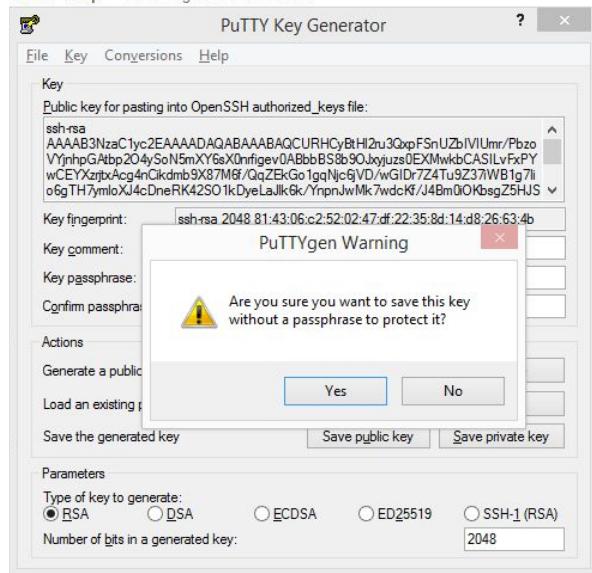
3. Select your .pem file and click on Open.



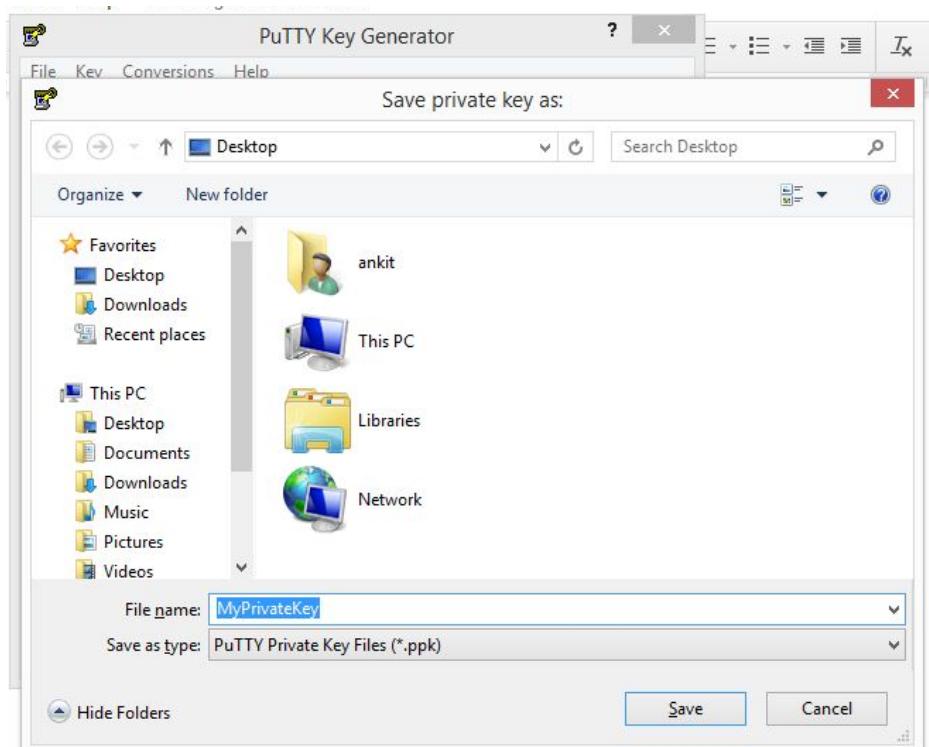
4. Click on Ok , if Putty show any notice.



5. Click on **Save private key**



6. Click on Yes.



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

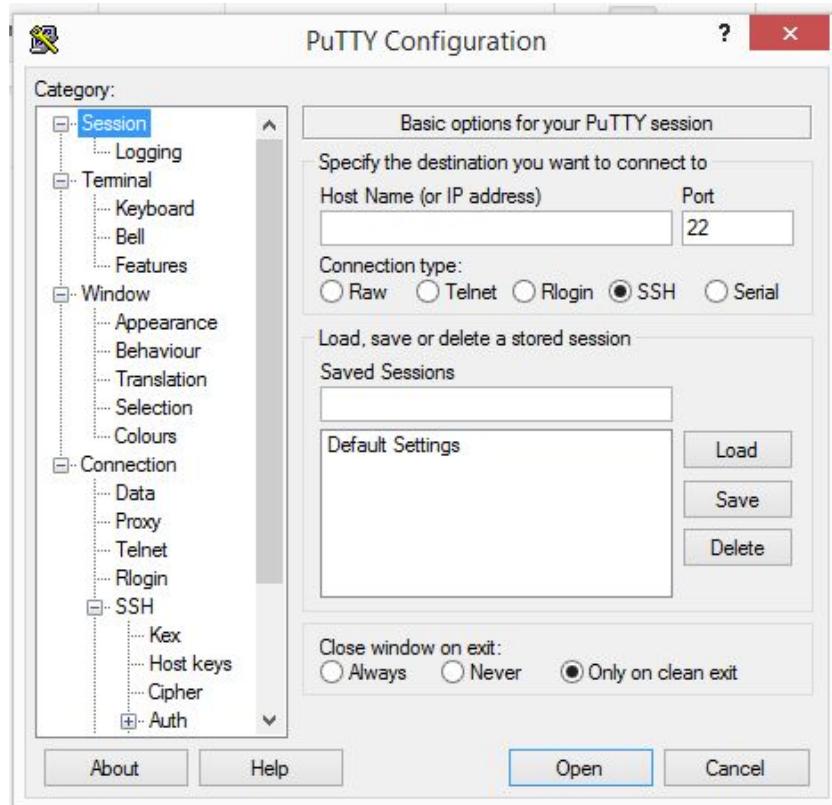
Congratulations :) Successfully created Private key(ppk) file.

## Accessing Amazon EC2 Machine using SSH Client

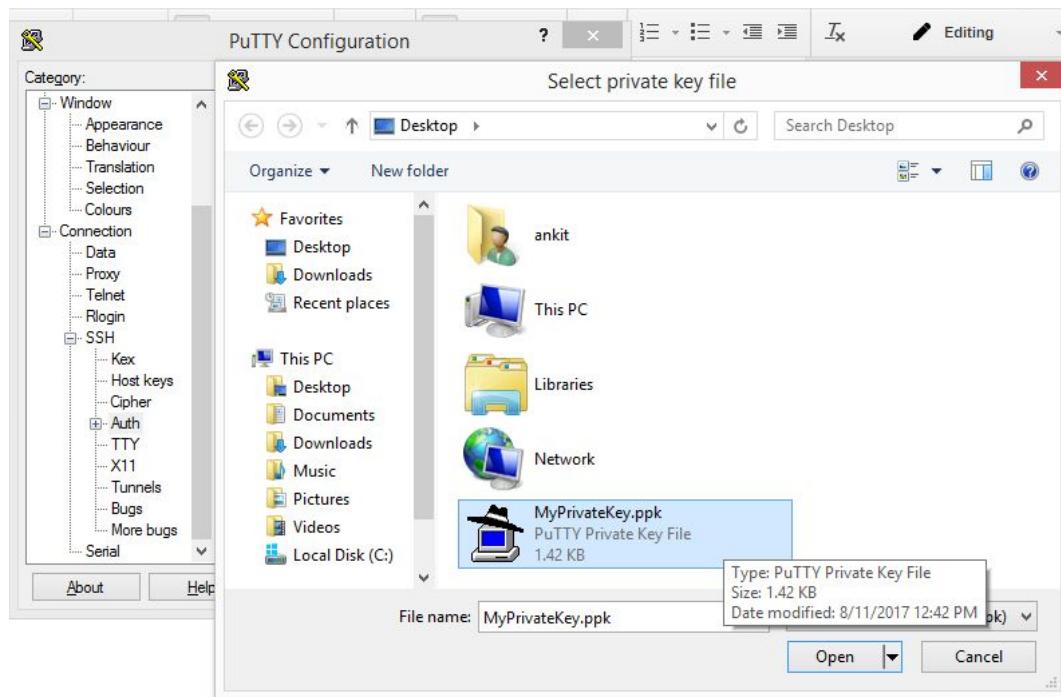
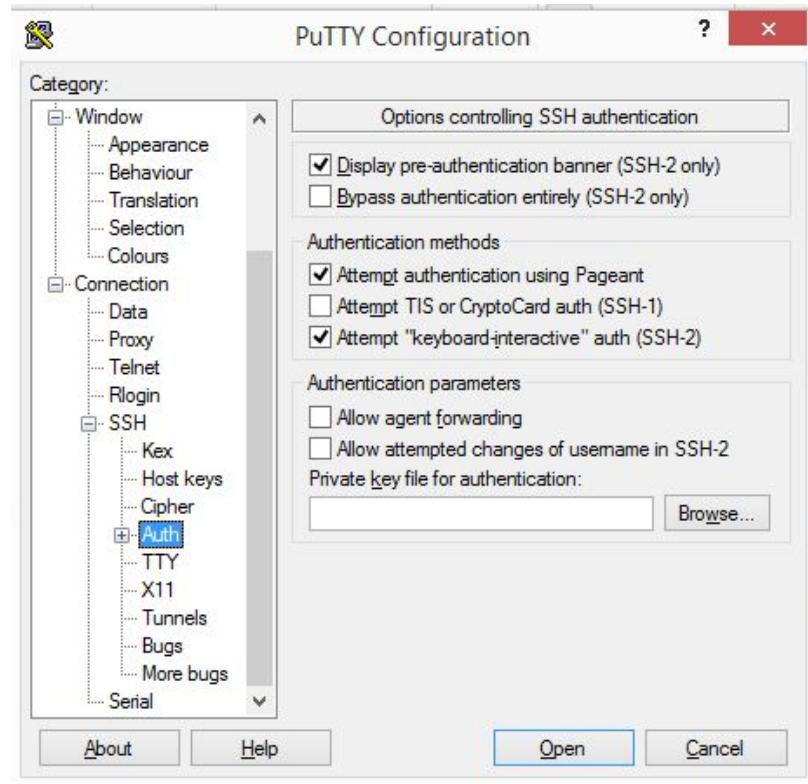
Note : Make sure that Putty ssh client should be installed in your machine.



### 1. Open Putty

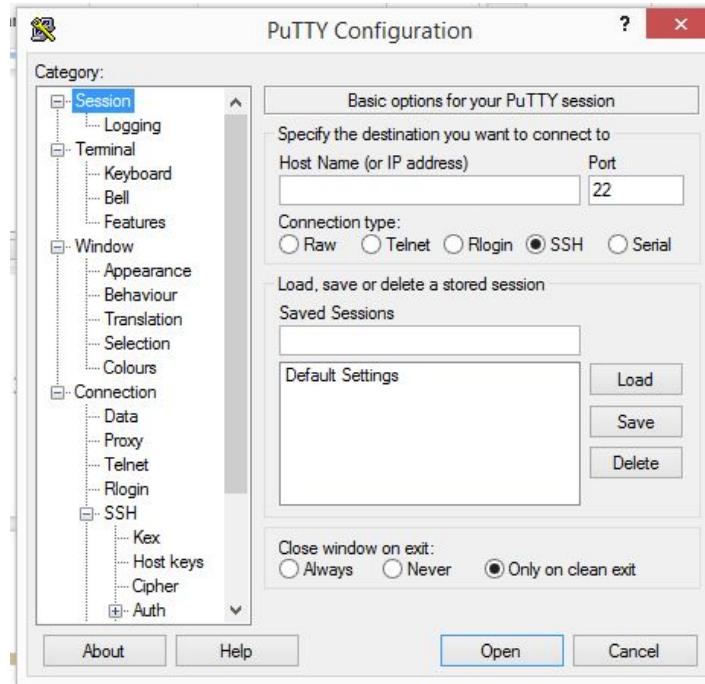


2. Go to **Connection > SSH > Auth** , Click on **Browse** , Choose private key (.ppk) file.



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>

A screenshot of the AWS Management Console homepage in Google Chrome. The URL is https://us-east-2.console.aws.amazon.com/console/home?region=us-east-2. The top navigation bar shows tabs for inbox, secure, AWS Management, and WinSCP. Below the navigation is a search bar and a navigation menu with 'Services' and 'Resource Groups'. The main content area is titled 'AWS services' and lists various services under categories: 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), and Mobile Services (Mobile Hub, Cognito, Device Farm). To the right, there's a 'Helpful tips' section with 'Manage your costs' and 'Create an organization', and an 'Explore AWS' section with 'New Product Announcements' and 'Migrate from Oracle to Amazon Aurora'.

7. Click EC2 under Compute Category.

## 8. Screen will be on EC2 Dashboard.

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**

**Scheduled Events**

**AWS Marketplace**

## 9. Click on Running Instances.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public IP
i-0fcfc4fc3afe2f71e	i-0fcfc4fc3afe2f71e	t2.micro	us-east-2c	running	2/2 checks ... None	None	52.14.202.117

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	IPv4 Public IP: 52.14.202.117	
Instance state: running	IPv6 IPs: -		
Instance type: t2.micro	Private DNS: ip-172-31-43-82.us-east-2.compute.internal		
Elastic IPs:	Private IPs: 172.31.43.82		
Availability zone: us-east-2c			

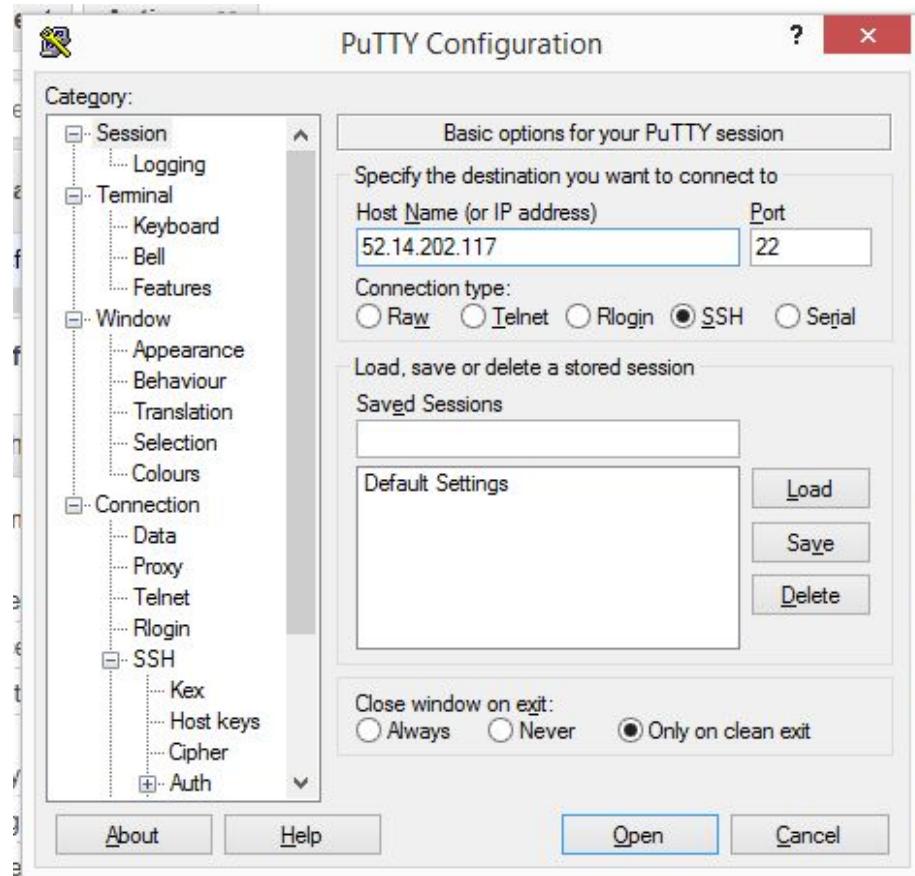
10. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc details.

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 (which is selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security, and Security Groups. 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, t2.micro, us-east-2c, running, with 2/2 checks passing and 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. Under the Description tab, detailed information is provided for the instance, including Instance ID, Instance state, Instance type, Elastic IPs, Availability zone, Security groups, Scheduled events, AMI ID, and VPC ID. 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 and Terms of Use.

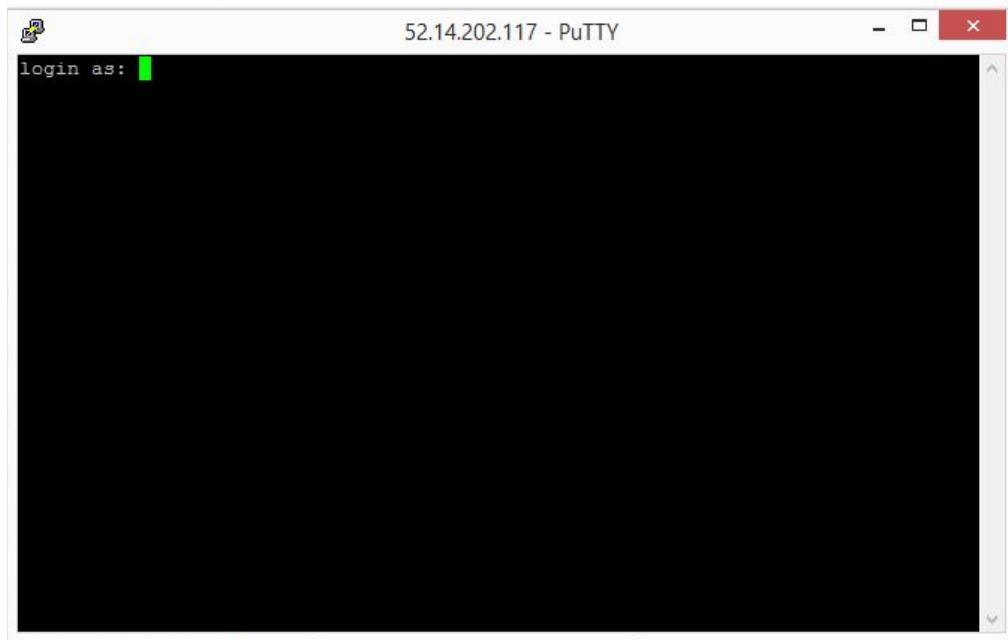
11. Copy the Public IP.

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.

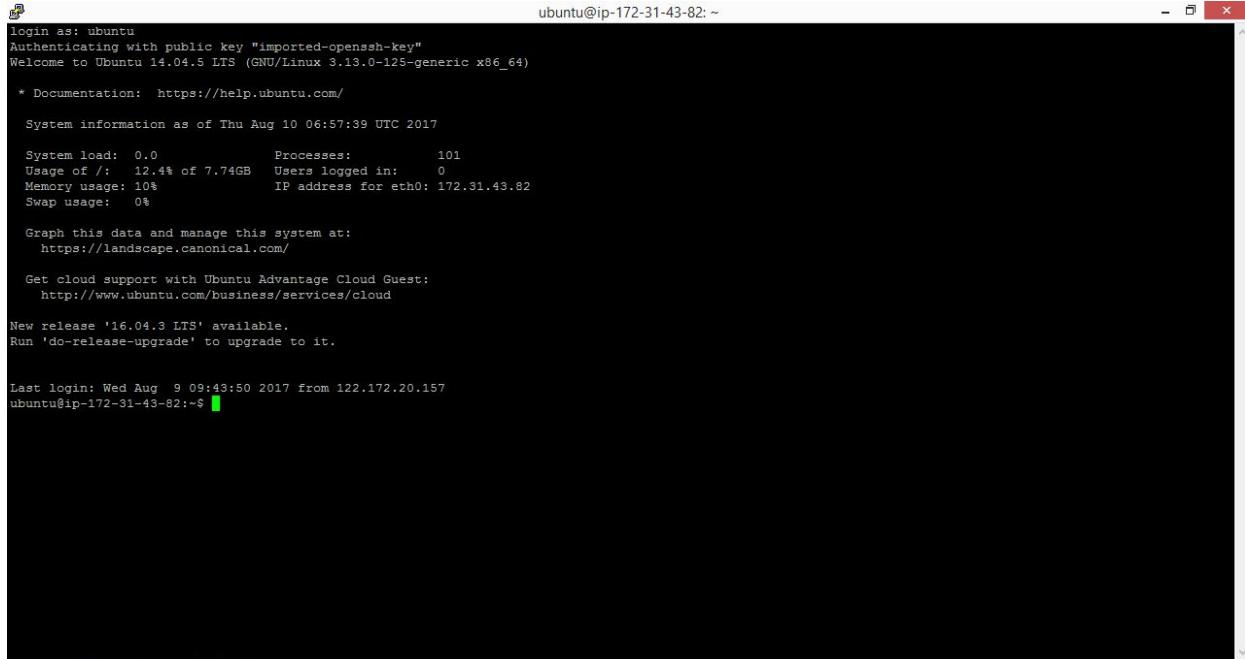


13. Click on Open.



14. Ask for username , login as : enter **ubuntu** as user name

15. After successfully login , will have terminal of EC2 virtual server.



The screenshot shows a terminal window titled "ubuntu@ip-172-31-43-82: ~". The window displays the following text:

```
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:~$
```

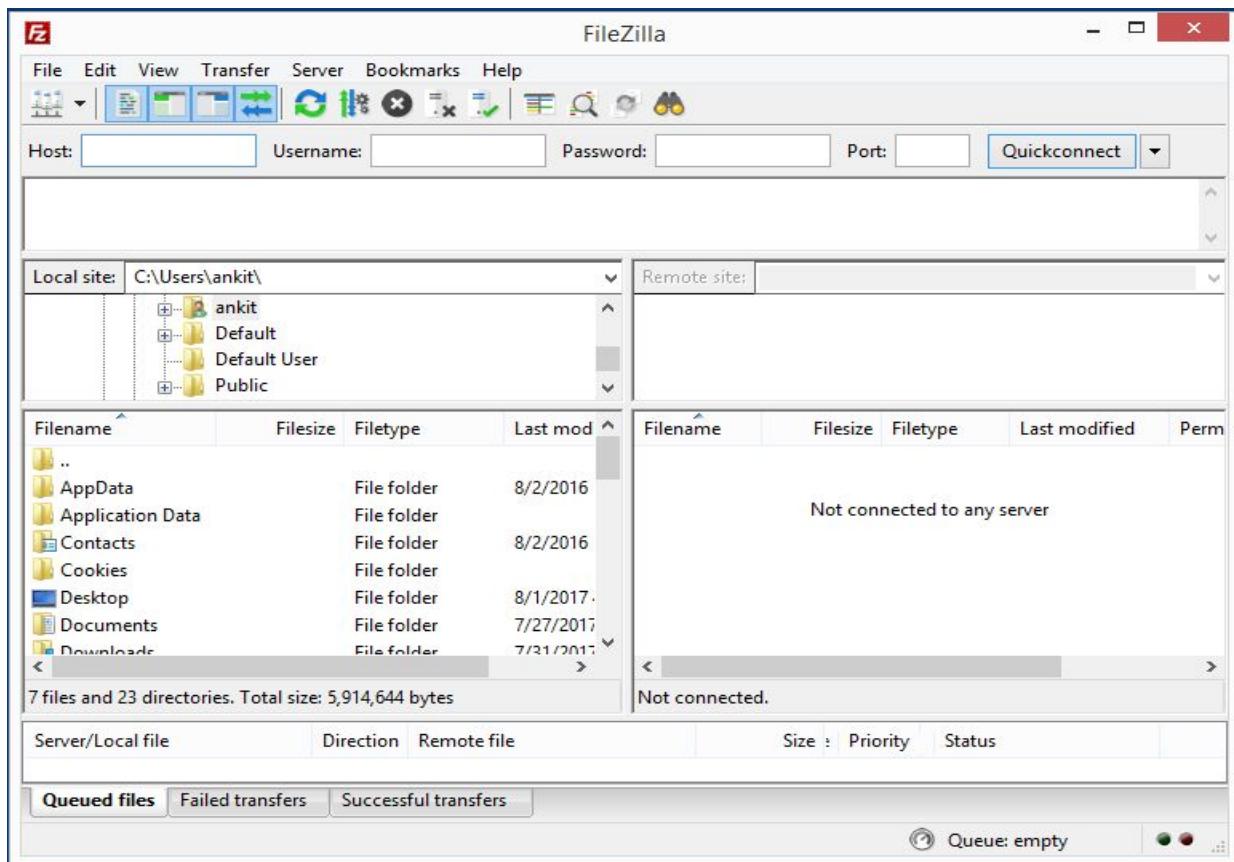
Congratulations :) you are connected to virtual machine using Putty SSH Client.

## Accessing Amazon EC2 Machine using FileManager

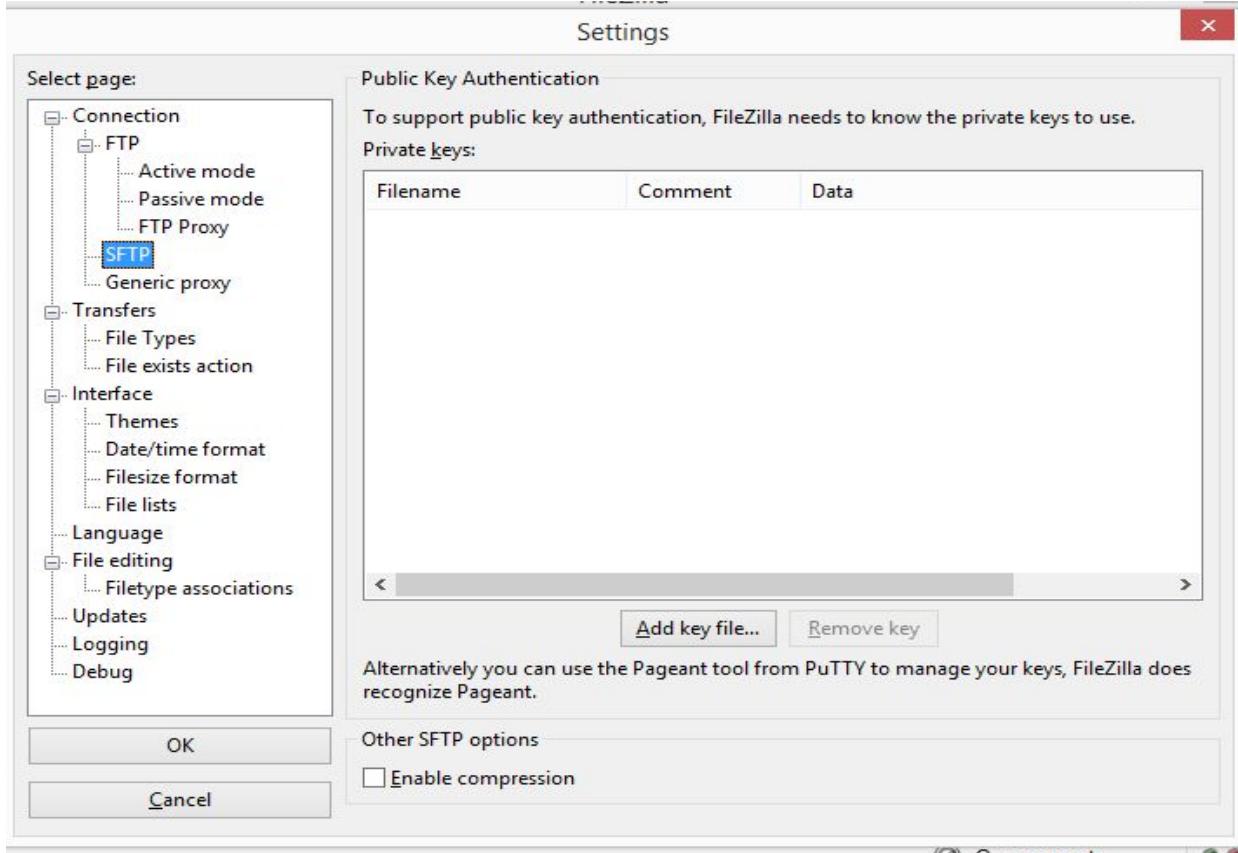
Note : Make sure that any one file manager ( FileZilla , WinScp ..etc ) should be installed in your machine.



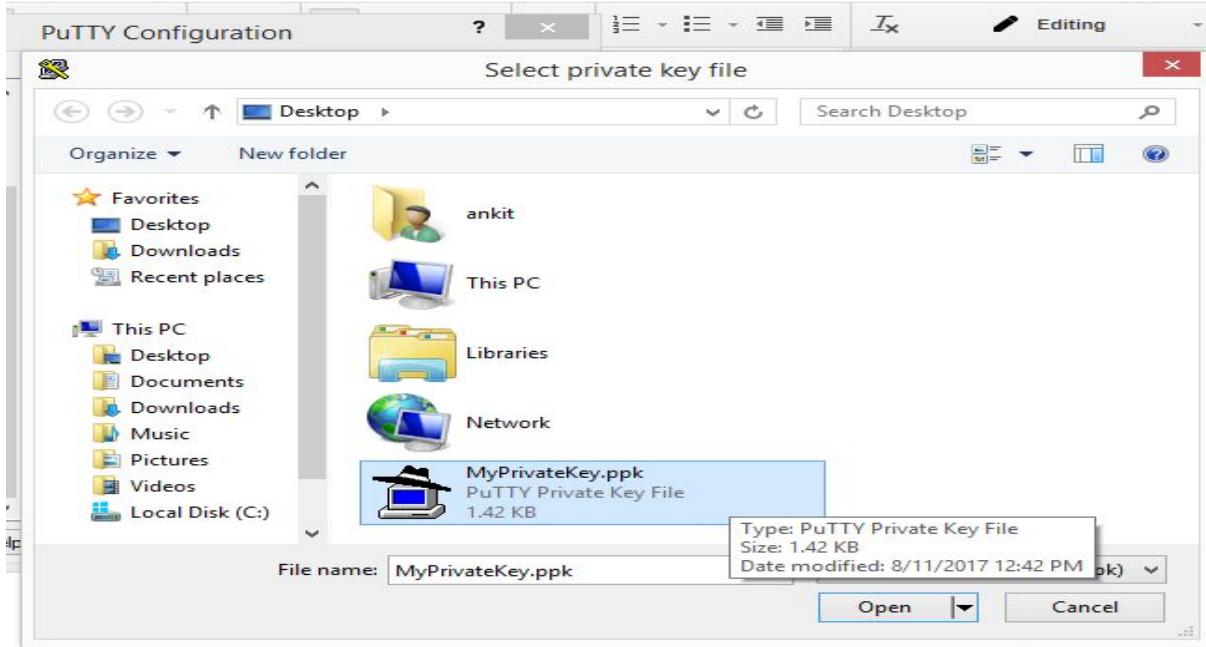
1. Open FileZilla.



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



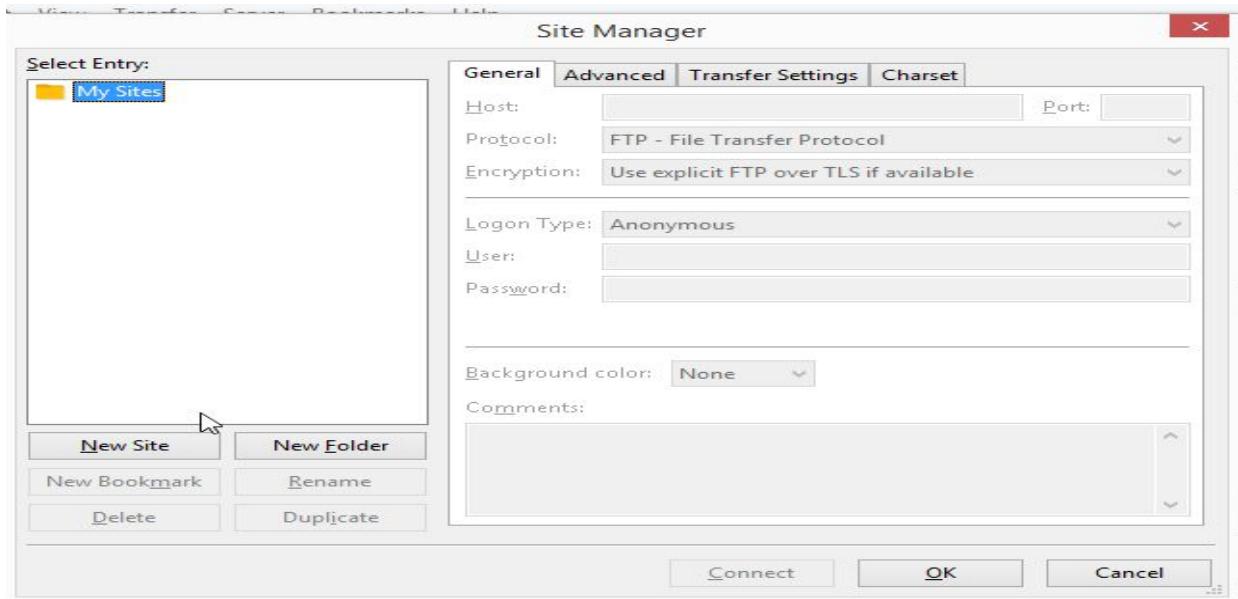
3. Click on **Add key file...**



4. Select your **.pem** or **.ppk** file and click on **Open**.

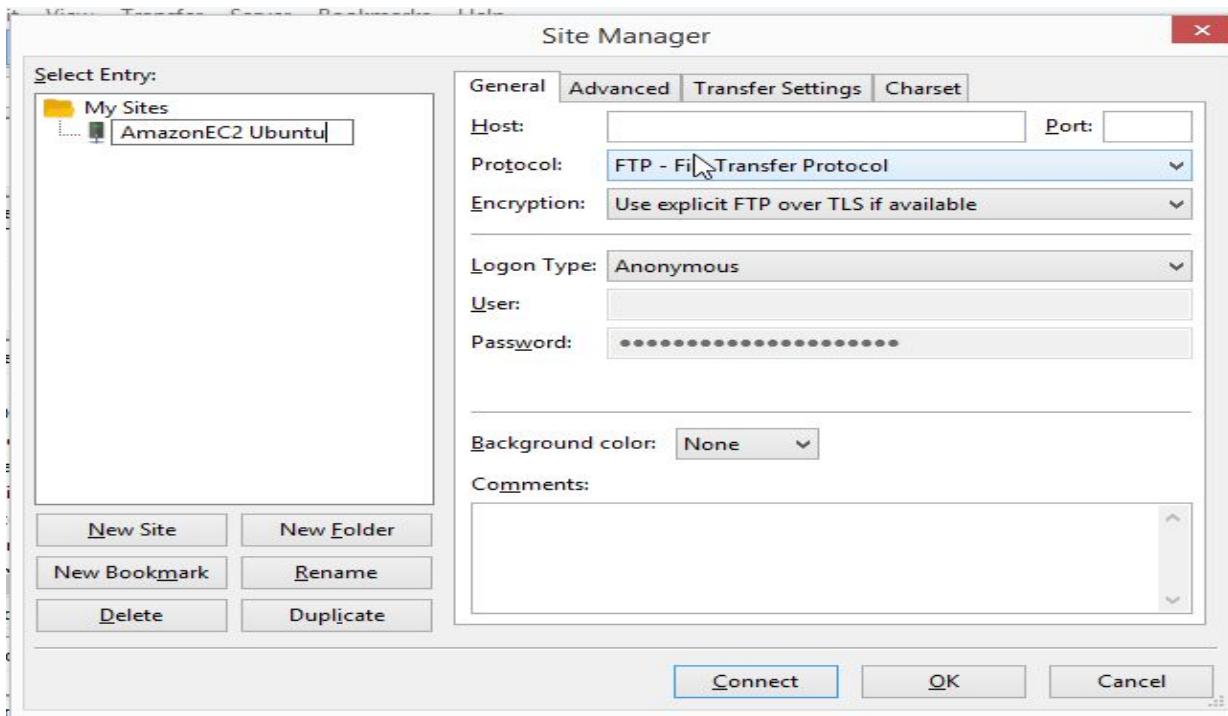
5. Click on **OK**.

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.

**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**

**Scheduled Events**

**AWS Marketplace**

12. Click on Running Instances.

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

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 (which is selected), Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security, and Security Groups. 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, and Public DNS. A single instance is listed: i-0fcfc4fc3afe2f71e, t2.micro, us-east-2c, running, 2/2 checks, None, and 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

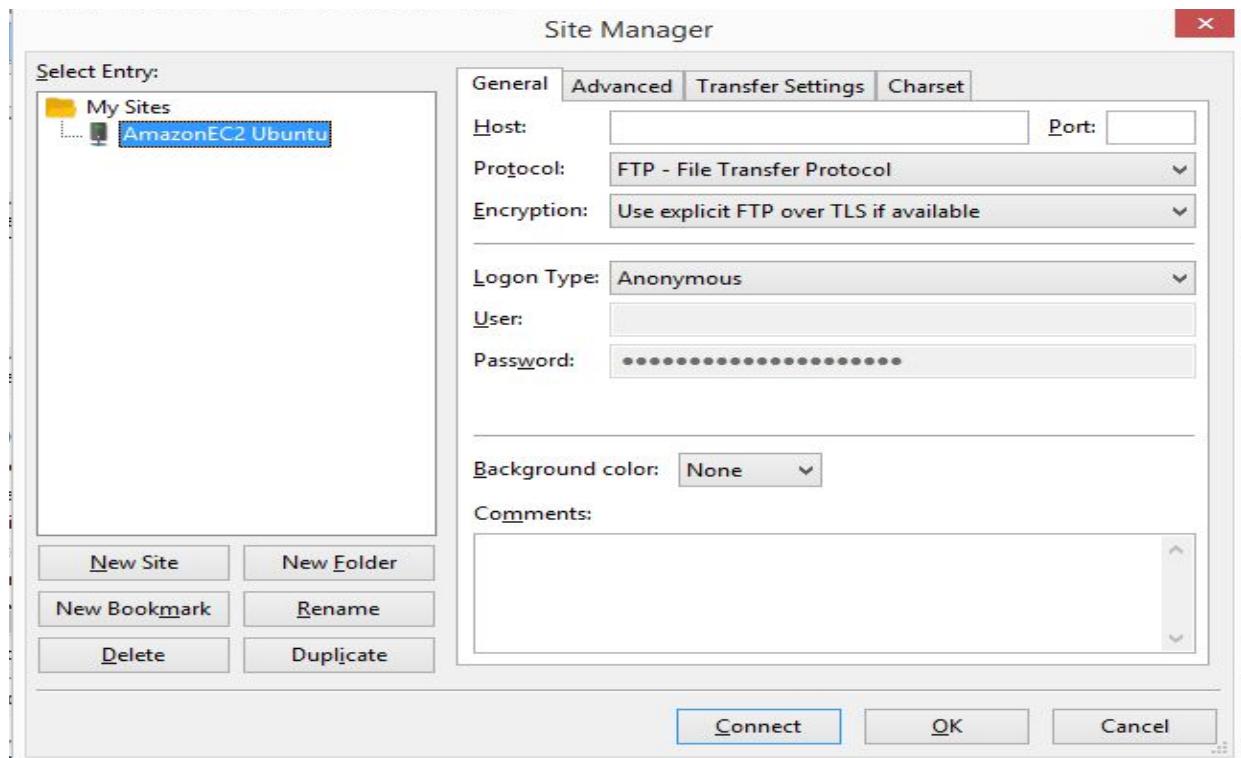
**Description**

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-13391926)	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.

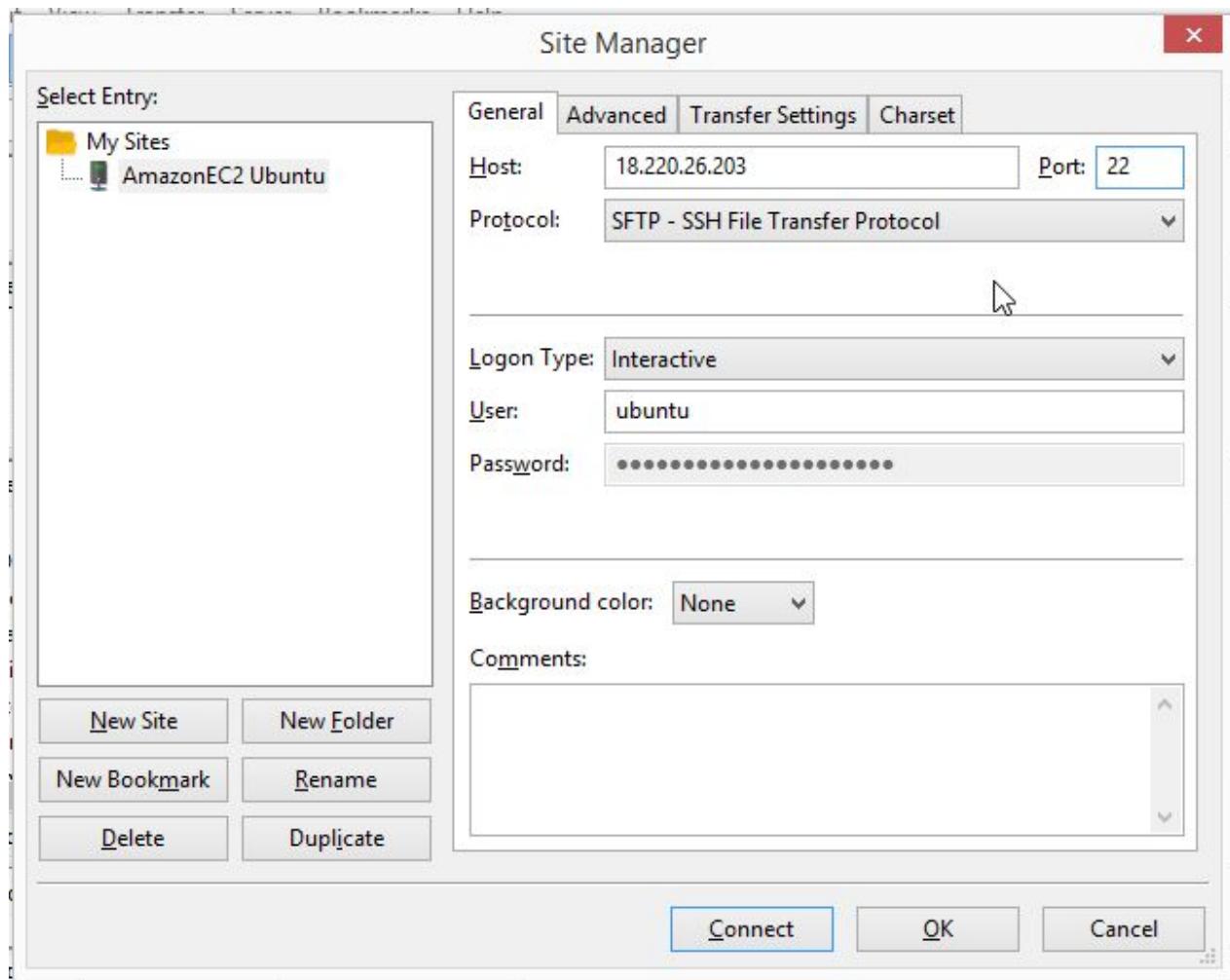
Host : Public IP

Port : 22

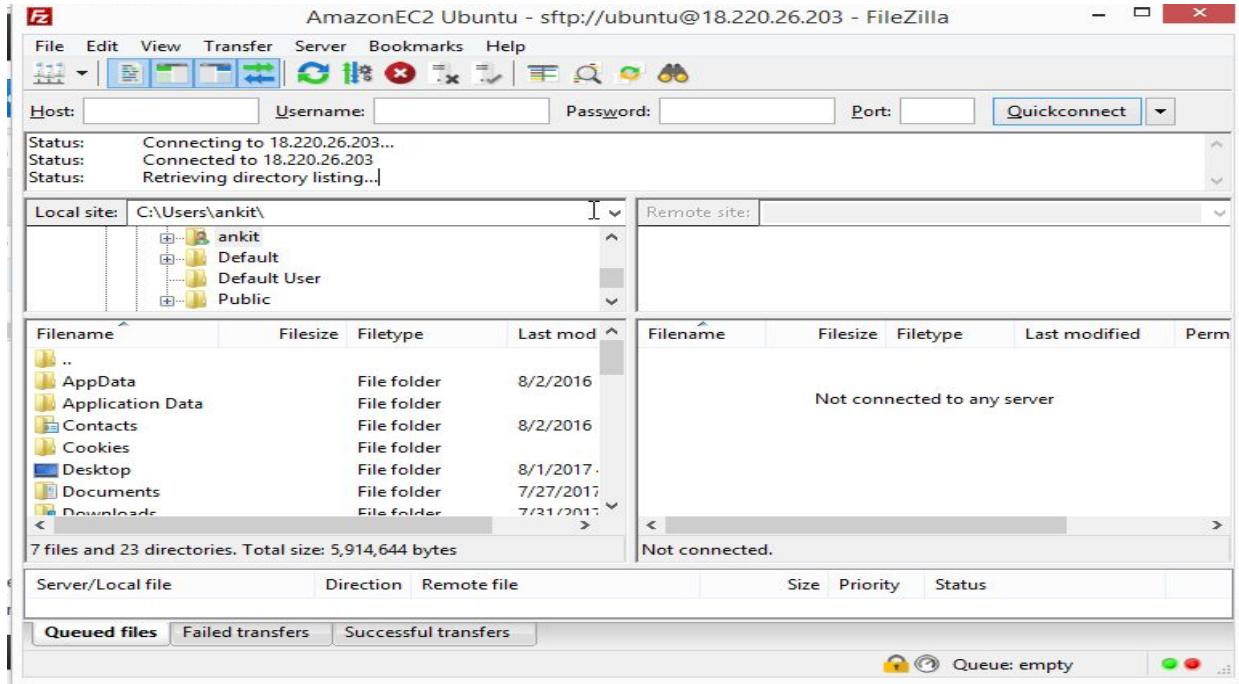
Protocol : SFTP

Logon Type : Interective

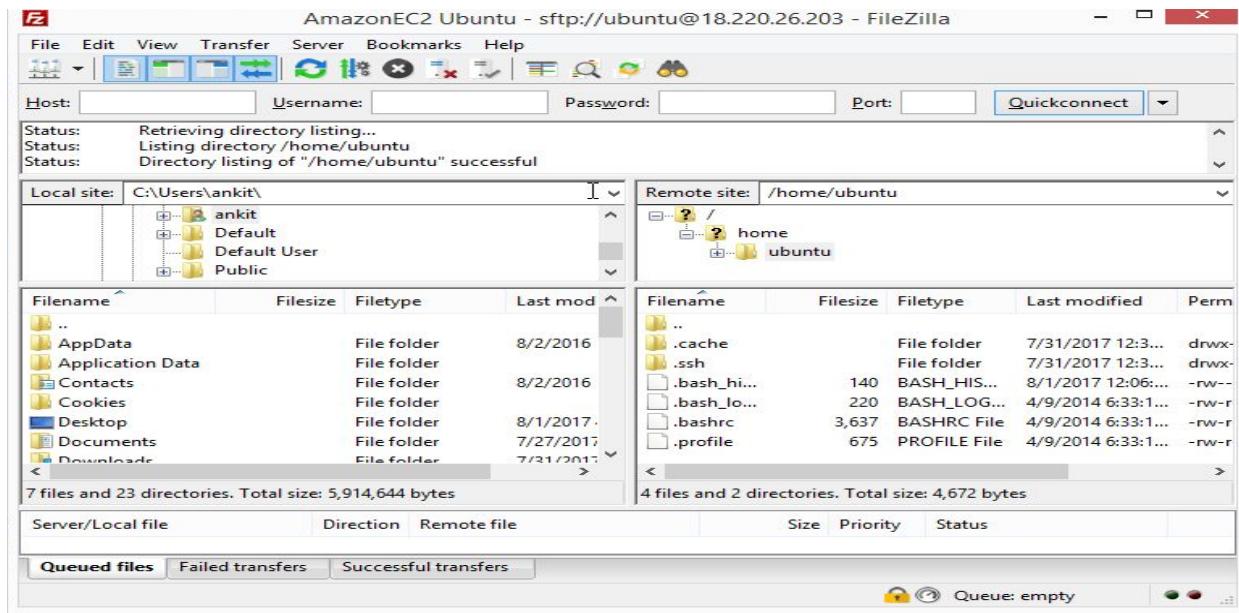
User : ubuntu



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.

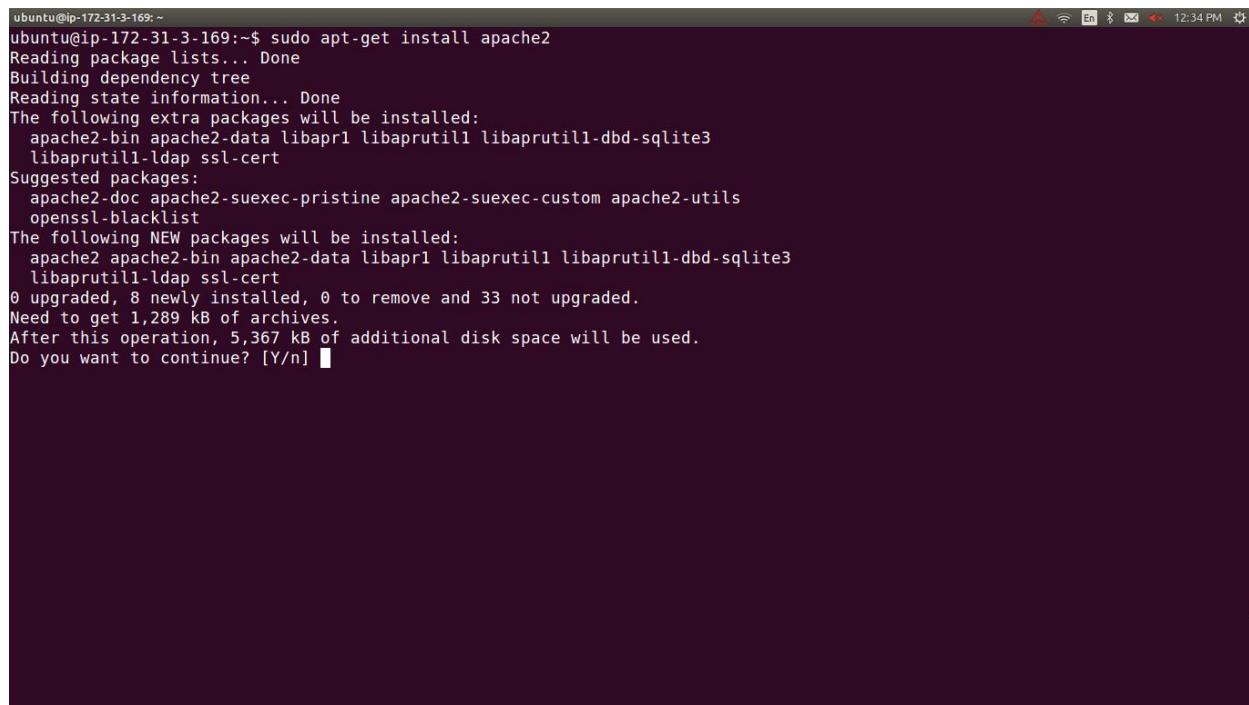
Last login: Wed Aug  9 09:43:50 2017 from 122.172.20.157
ubuntu@ip-172-31-43-82:~$ 
```

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 
```



```
ubuntu@ip-172-31-3-169: ~
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

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 system. The command entered is "sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt". The terminal shows standard output from the package manager, including dependency resolution and upgrade information.

A screenshot of a terminal window showing the detailed output of the "sudo apt-get install" command. It displays the process of reading package lists, building a dependency tree, and determining which packages will be installed (extra and NEW). It also shows the number of upgrades, newly installed packages, and the size of the archive to be downloaded. The user is prompted with "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
```

A screenshot of a terminal window on an Ubuntu system. The command 'sudo apt-get install mysql-server php5-mysql' is being typed into the terminal. The terminal window has a dark background and light-colored text. At the top right, there are icons for signal strength, battery level, and the current time (11:57 AM).

A screenshot of a terminal window showing the output of the 'sudo apt-get install mysql-server php5-mysql' command. The output includes:

- 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 libmldb-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

ubuntu@ip-172-31-3-169: ~  
Package configuration

```
Configuring mysql-server-5.5
While not mandatory, it is highly recommended that you set a password for the MySQL administrative "root" user.
If this field is left blank, the password will not be changed.
New password for the MySQL "root" user:
<0k>
```

ubuntu@ip-172-31-3-169: ~  
Package configuration

```
Configuring mysql-server-5.5
Repeat password for the MySQL "root" user:
<0k>
```

4. Next, Restart the apache2 service to recognise 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 'AWS services' under categories like Compute, Storage, and Database. On the right, there's a 'Helpful tips' section with links to 'Manage your costs', 'Create an organization', and 'Explore AWS'.

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

The screenshot shows the EC2 Management Console dashboard. It displays resource counts (1 Running Instances, 2 Elastic IPs, etc.) and a sidebar with navigation links for EC2 Dashboard, Instances, Images, and more. On the right, there are sections for Account Attributes (Supported Platforms, Default VPC) and Additional Information (Getting Started Guide, Documentation).

4. Click on Running Instances.

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: Instance ID i-0fcfc4fc3afe2f71e, Instance Type t2.micro, Availability Zone us-east-2c, and Instance State running. Below the table, a detailed view of the instance is shown with tabs for Description, Status Checks, Monitoring, and Tags. The 'Description' tab is selected, displaying the following information:

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

5. 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. At the top, it displays the instance ID and Public DNS. Below this, there are tabs for Description, Status Checks, Monitoring, and Tags. The 'Description' tab is selected, showing the following details:

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</a> , <a href="#">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>

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

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

7. Click on Inbound

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0

8. Click on Edit

**Edit inbound rules**

Type	Protocol	Port Range	Source
SSH	TCP	22	Custom 0.0.0.0/0

**Add Rule**

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.

**Cancel Save**

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0

## 9. Click on Add Rule

**Protocol**

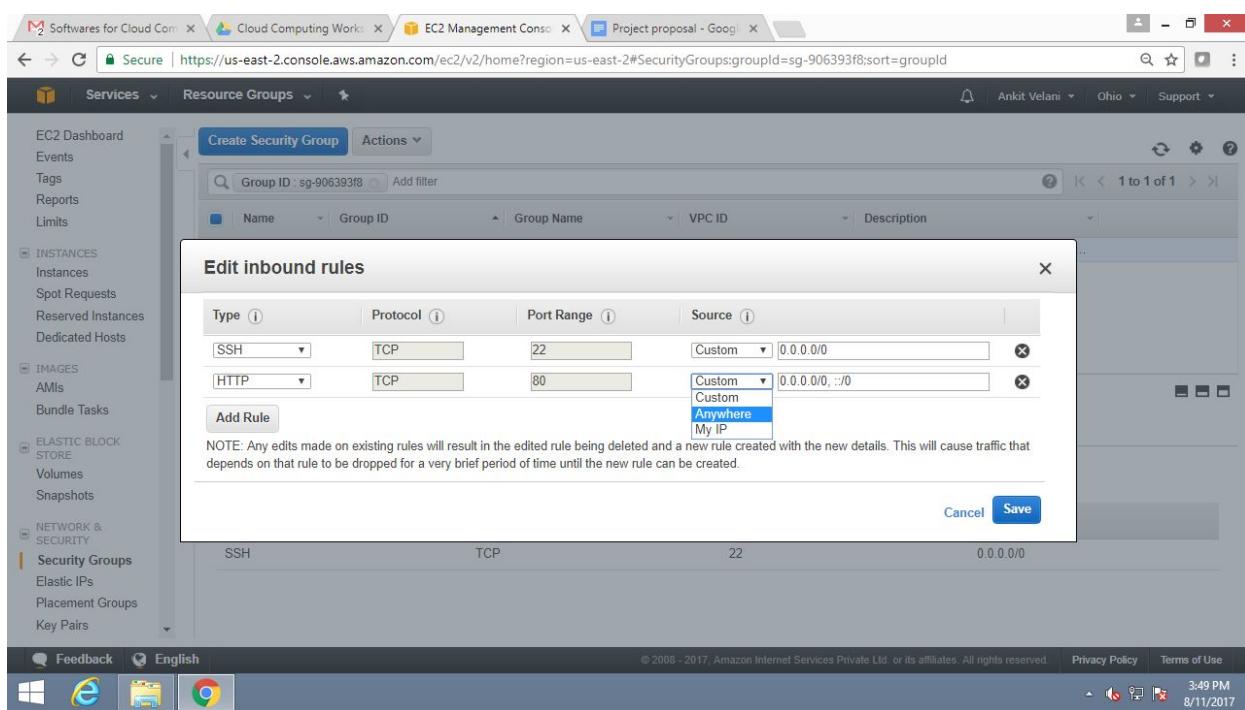
- All TCP
- All UDP
- All ICMP - IPv4
- All ICMP - IPv6
- All traffic
- SSH
- SMTP
- DNS (UDP)
- DNS (TCP)
- HTTP**
- POP3
- IMAP
- LDAP
- HTTPS
- SMTPS
- Custom TCP

**Add Rule**

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.

**Cancel Save**

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0



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 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, 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 selected, showing "i-0fcfc4fc3afe2f71e" as the Instance ID, "t2.micro" as the Instance Type, "us-east-2c" as the Availability Zone, and "running" as the Instance State. It also shows "2/2 checks ... None" under Status Checks and "ec2-52-14-202-117.us-east-2.compute.amazonaws.com" under Public DNS. At the bottom of the table, there are tabs for Description, Status Checks, Monitoring, and Tags. The "Description" tab is active, displaying detailed information about the instance, such as its public and private IP addresses, security groups, and VPC details.

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

**Public DNS (IPv4)** ec2-52-14-202-117.us-east-2.compute.amazonaws.com

**IPv4 Public IP** 52.14.202.117

**IPv6 IPs** -

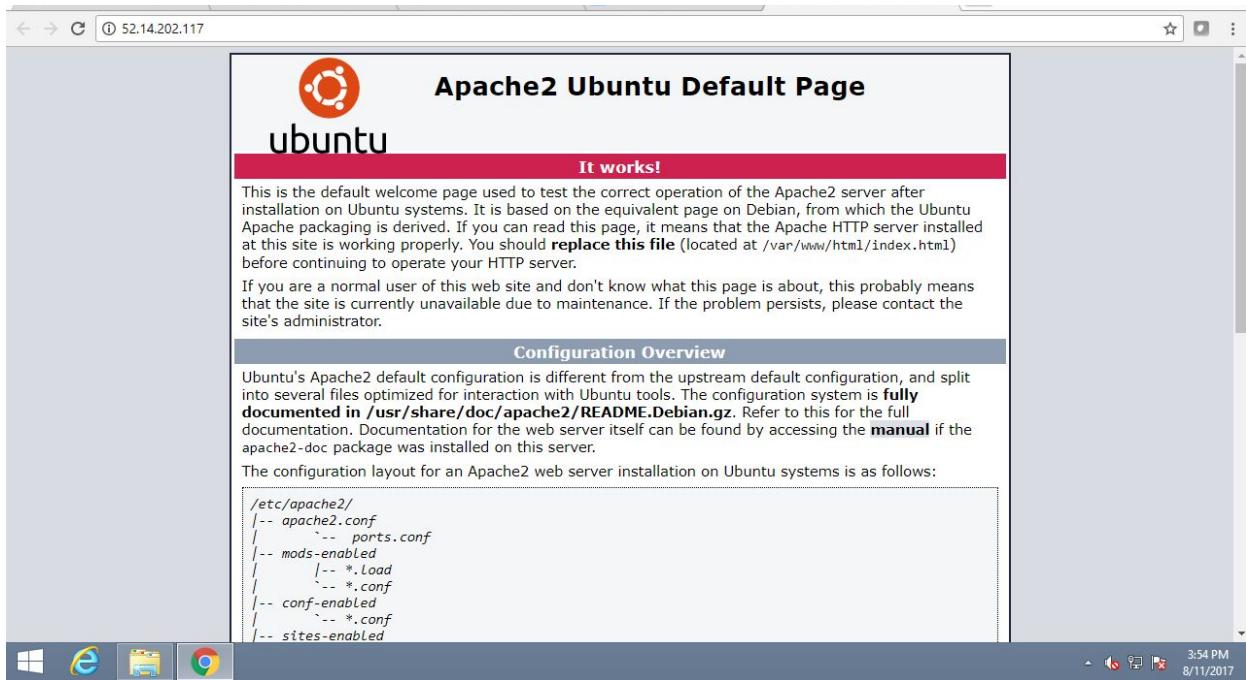
**Private DNS** ip-172-31-43-82.us-east-2.compute.internal

**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 ?

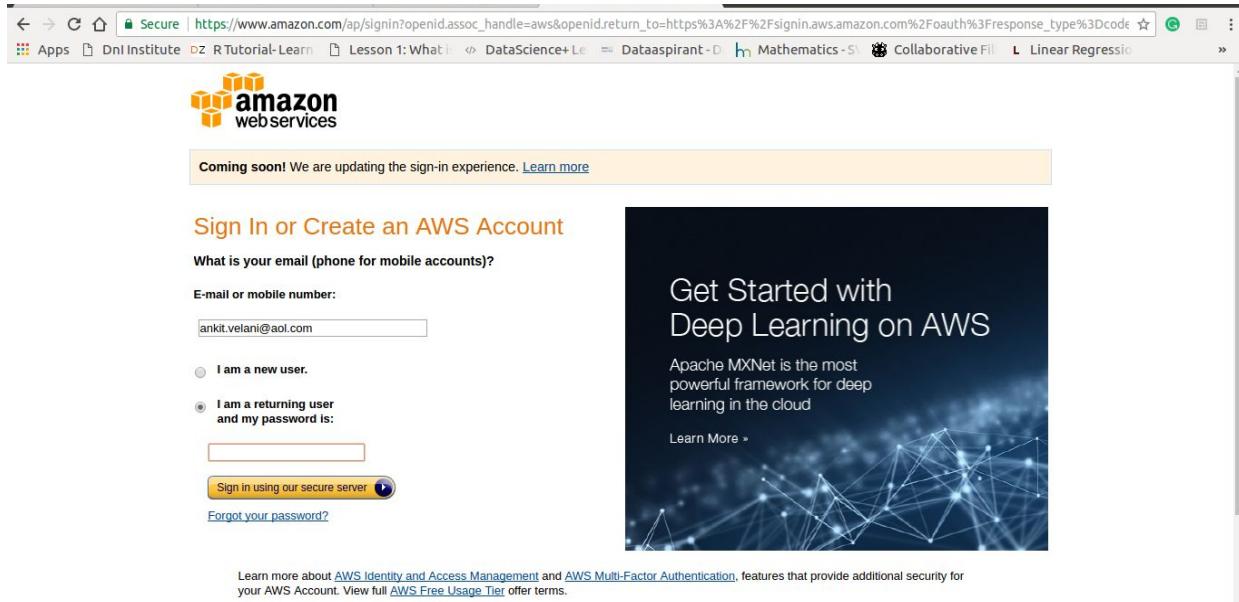
Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

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.

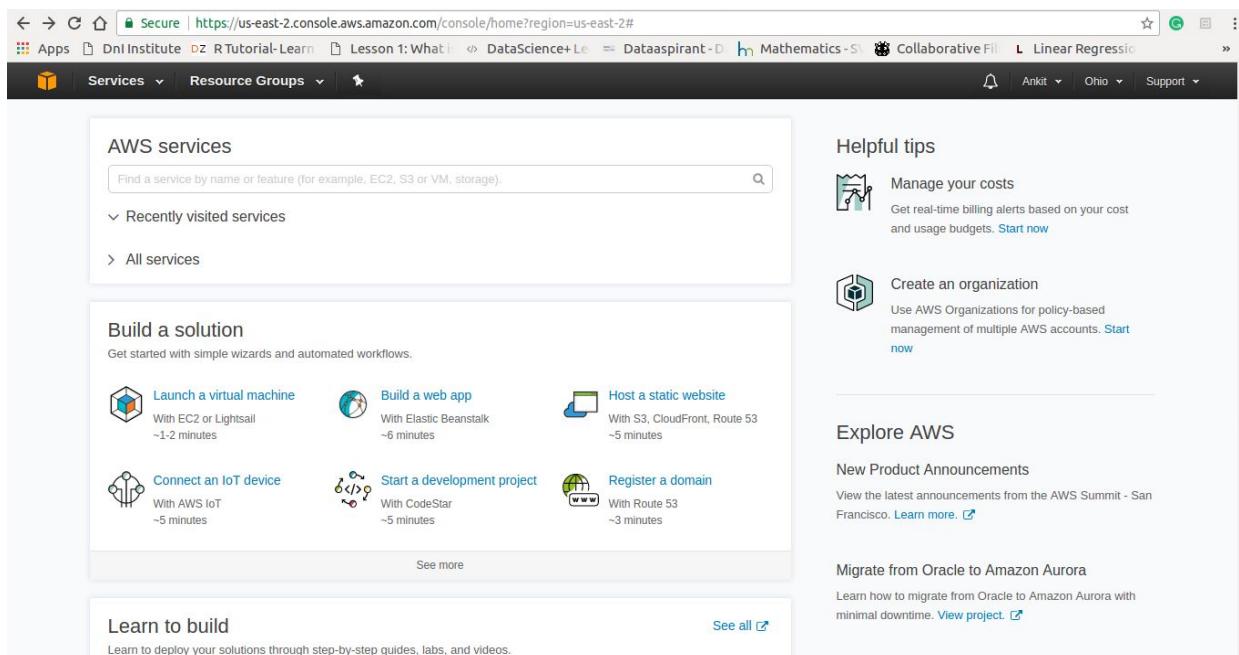
- When you buy a server, you get CPU, memory, storage, and IOPS, all bundled together. With Amazon RDS, these are split apart so that you can scale them independently. So, for example, if you need more CPU, less IOPS, or more storage, you can easily allocate them.
- Amazon RDS manages backups, software 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 home page. In the top navigation bar, 'Services' is selected. The main content area is titled 'AWS services' and contains a search bar. Below the search bar, there are two sections: 'Recently visited services' and 'All services'. The 'All services' section is expanded and displays several categories of 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, Managed Services.
- 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 of the page, there is a 'Helpful tips' sidebar with links to 'Manage your costs' and 'Create an organization'. Below that is a 'Explore AWS' section with links to 'New Product Announcements' and 'Migrate from Oracle to Amazon Aurora'.

## 6. Choose RDS, from category " Database"

The screenshot shows the same AWS Management Console home page as above, but the 'Database' category under 'All services' is now expanded. This reveals the following sub-services:

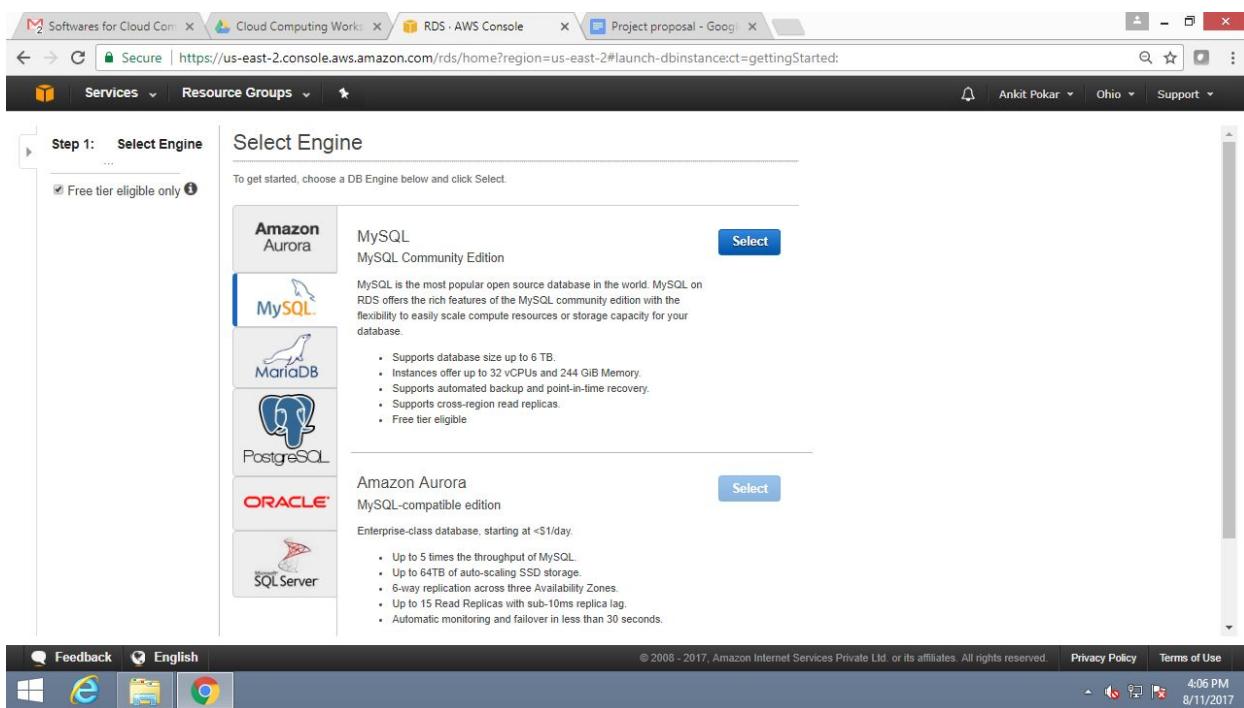
- Database:** RDS, DynamoDB, ElastiCache, Redshift.
- Storage:** S3, EFS, Glacier, Storage Gateway.
- Networking & Content Delivery:** VPC, CloudFront, Direct Connect, Route 53.
- Migration:** Application Discovery Service, DMS, Server Migration, Snowball.
- Management Tools:** CloudWatch, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Trusted Advisor, Managed Services.
- Security, Identity & Compliance:** IAM, Inspector, Certificate Manager, Directory Service, WAF & Shield, Artifact.
- Analytics:** Athena, EMR, CloudSearch, Elasticsearch Service, Kinesis, Data Pipeline, QuickSight.
- Game Development:** Amazon GameLift.
- Mobile Services:** Mobile Hub, Cognito, Device Farm, Mobile Analytics, Pinpoint.
- Application Services:** Step Functions, SWF, API Gateway, Elastic Transcoder.
- Messaging:** Simple Queue Service, Simple Notification Service, SES.
- Business Productivity:** WorkDocs, WorkMail, Amazon Chime.

On the right side of the page, there are several promotional cards: 'Get started with the most scalable framework for deep learning in the cloud.', 'Build Applications with AWS Lambda', 'Amazon DynamoDB', 'AWS Marketplace', and 'Have feedback?'. Each card includes a 'Learn more.' link.

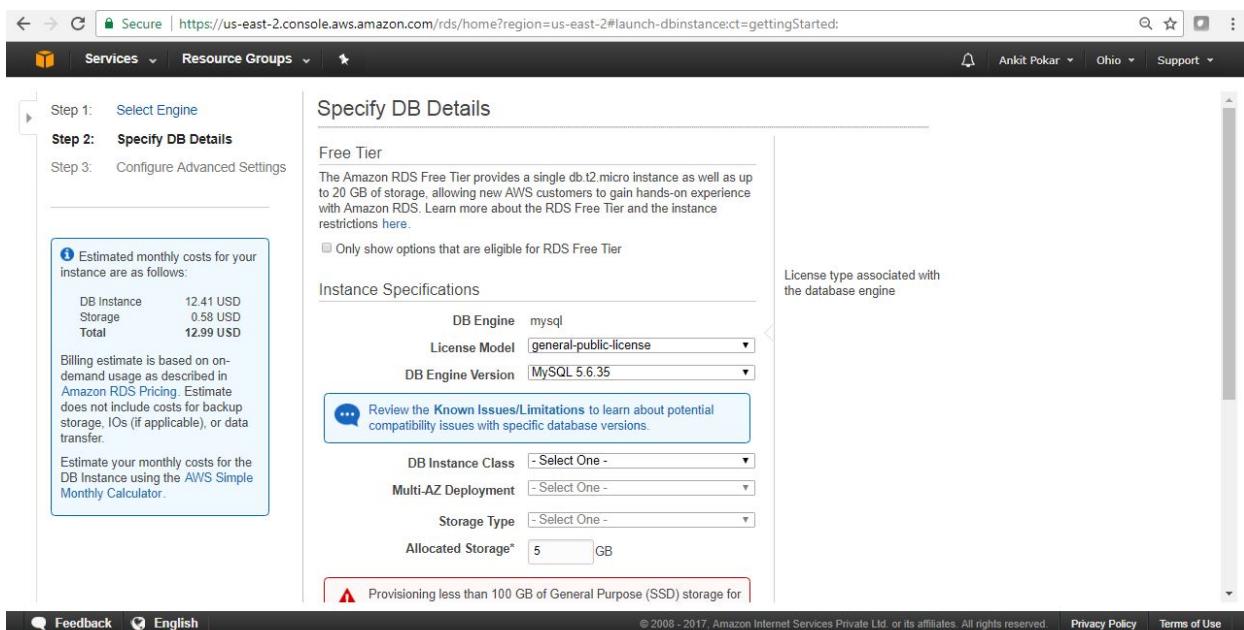
## 7. Click on RDS.

## 8. Click on Get Started Now

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



## 10. Click on Select.



## 11. Select the Only "show options that are eligible for RDS Free Tier"

**Step 1: Select Engine**

**Step 2: Specify DB Details**

**Step 3: Configure Advanced Settings**

**Free Tier**  
The Amazon RDS Free Tier provides a single db.t2.micro instance as well as up to 20 GB of storage, allowing new AWS customers to gain hands-on experience with Amazon RDS. Learn more about the RDS Free Tier and the instance restrictions here.

Only show options that are eligible for RDS Free Tier

**Instance Specifications**

DB Engine	mysql
License Model	general-public-license
DB Engine Version	MySQL 5.6.35

Review the Known Issues/Limitations to learn about potential compatibility issues with specific database versions.

DB Instance Class	db.t2.micro — 1 vCPU, 1 GiB RAM
Multi-AZ Deployment	No
Storage Type	General Purpose (SSD)
Allocated Storage*	5 GB

Estimated monthly costs for your instance are as follows:

DB Instance	12.41 USD
Storage	0.58 USD
Total	12.99 USD

Billing estimate is based on on-demand usage as described in Amazon RDS Pricing. Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

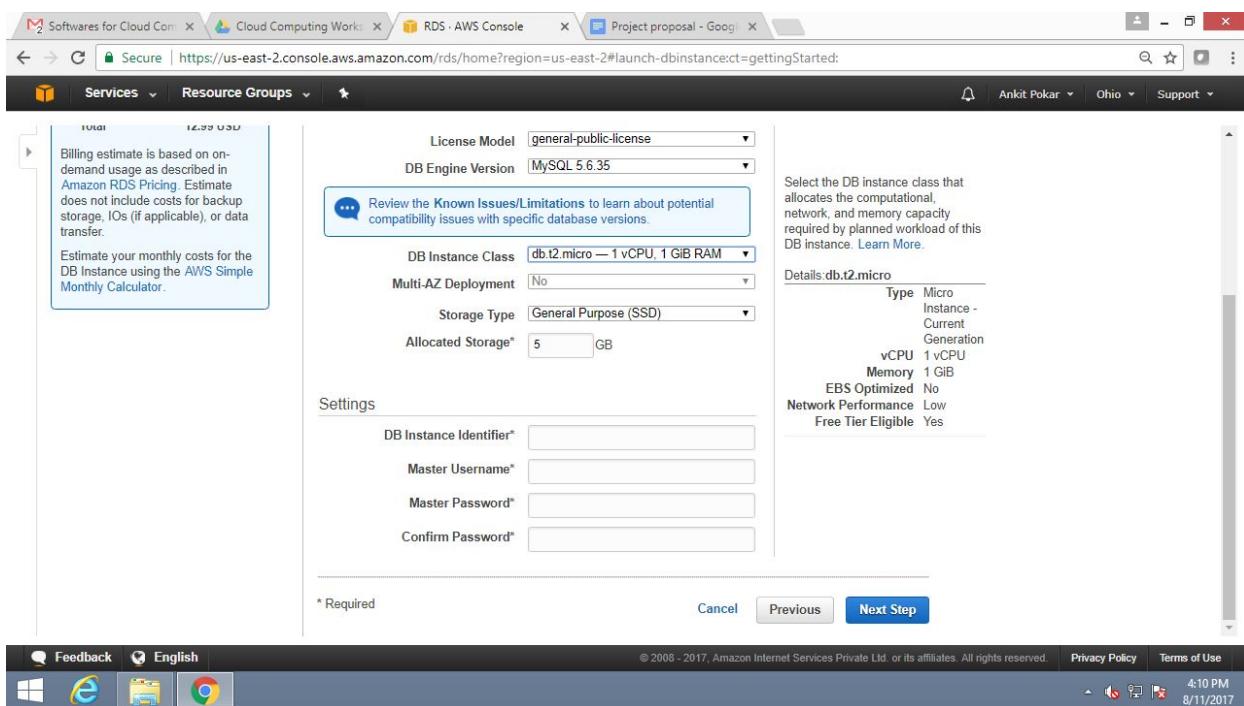
Estimate your monthly costs for the DB Instance using the AWS Simple Monthly Calculator.

Feedback English © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use 4:09 PM 8/11/2017

## 12. Free tier Instance configuration

Details:db.t2.micro	
Type	Micro Instance - Current Generation
vCPU	1 vCPU
Memory	1 GiB
EBS Optimized	No
Network Performance	Low
Free Tier Eligible	Yes

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



## Settings

<b>DB Instance Identifier*</b>	<input type="text" value="cms"/>
<b>Master Username*</b>	<input type="text" value="root"/>
<b>Master Password*</b>	<input type="password" value="....."/>
<b>Confirm Password*</b>	<input type="password" value="....."/>

### 14. Specify the Database settings

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

Master Username :====> **root**

Master Password :====> **mypassword**

Confirm Password :====> **mypassword**

Billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, I/Os (if applicable), or data transfer.

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

Review the Known Issues/Limitations to learn about potential compatibility issues with specific database versions.

DB Instance Identifier\*: cms

Master Username\*: root

Master Password\*: .....  
Confirm Password\*: .....

\* Required      Cancel      Previous      **Next Step**

15. Click on Next.

16. It allow us to do any advanced configuration.

Step 1: Select Engine

Step 2: Specify DB Details

**Step 3: Configure Advanced Settings**

### Configure Advanced Settings

#### Network & Security

VPC\*: Default VPC (vpc-ba2b68d3)

Subnet Group: default

Publicly Accessible: Yes

Availability Zone: No Preference

VPC Security Group(s): Create new Security Group  
default (VPC)  
launch-wizard-1 (VPC)  
launch-wizard-2 (VPC)

#### Database Options

Database Name:

Note: if no database name is specified then no initial MySQL database will be created on the DB instance.

Database Port: 3306

DB Parameter Group: default.mysql5.6

Option Group: default.mysql-5.6

DB Parameter Group: default.mysql5.6  
 Option Group: default.mysql-5-6  
 Copy Tags To Snapshots:   
 Enable IAM DB Authentication: No Preference  
 Enable Encryption: No

**Backup**  
 Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail here.  
 Backup Retention Period: 7 days  
 Backup Window: No Preference

**Monitoring**  
 Enable Enhanced Monitoring: No

**Maintenance**  
 Auto Minor Version Upgrade: Yes  
 Maintenance Window: No Preference

\* Required      Cancel      Previous      **Launch DB Instance**

## 17. Click on **Launch DB Instance**.

Step 1: Select Engine  
 Step 2: Specify DB Details  
 Step 3: Configure Advanced Settings

**Your DB Instance is being created.**  
 Note: Your instance may take a few minutes to launch.

Connecting to your DB Instance  
 Once Amazon RDS finishes provisioning your DB instance, you can use a SQL client application or utility to connect to the instance.  
 Learn about connecting to your DB instance

**View Your DB Instances**

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

The screenshot shows the Amazon RDS Dashboard. On the left, there's a sidebar with options like Instances, Clusters, Reserved Instances, Snapshots, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions, and Notifications. The main area has tabs for Launch DB Instance, Show Monitoring, and Instance Actions. A search bar at the top says "Search DB Instances...". Below it, a table shows one instance: Engine: MySQL, DB Instance: cms, Status: creating, Current Activity: None, Class: db.t2.micro, VPC: vpc-ba2b68d3, Multi-AZ: No. The monitoring section shows CPU, Memory, and Storage metrics with no data available. At the bottom, there are buttons for Instance Actions, Tags, and Logs.

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 various RDS services: Instances, Clusters, Reserved Instances, Snapshots, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions, and Notifications. The main content area features a section titled "Amazon Aurora" with a brief description and a "Launch Aurora (MySQL)" button. Below this is a "Resources" section showing usage statistics: DB Instances (1/40), Parameter Groups (1), Allocated Storage (5.00 GB/100.00 TB), Default (1), Click here to increase DB instances limit, Custom (0/100), Reserved DB Purchases (0/40), Option Groups (1), Snapshots (7), Default (1), Manual (0/100), Automated (1), Custom (0/20), Recent Events (4), Subnet Groups (1/50), Supported Platforms VPC, Event Subscriptions (0/20), and Default Network vpc-ba2b68d3. There's also a "Create Instance" section with a "Launch a DB Instance" button and a note about launching in the US East (Ohio) region. To the right, there's an "Additional Information" section with links to Starting with RDS, Overview and Features, Documentation, Articles and Tutorials, Data import guide for MySQL, Data import guide for Oracle, Data import guide for SQL Server, Pricing, and Forums. A "What's New" section highlights "Amazon Aurora – Lambda Integration". The footer includes standard AWS links like Feedback, English, Privacy Policy, and Terms of Use.

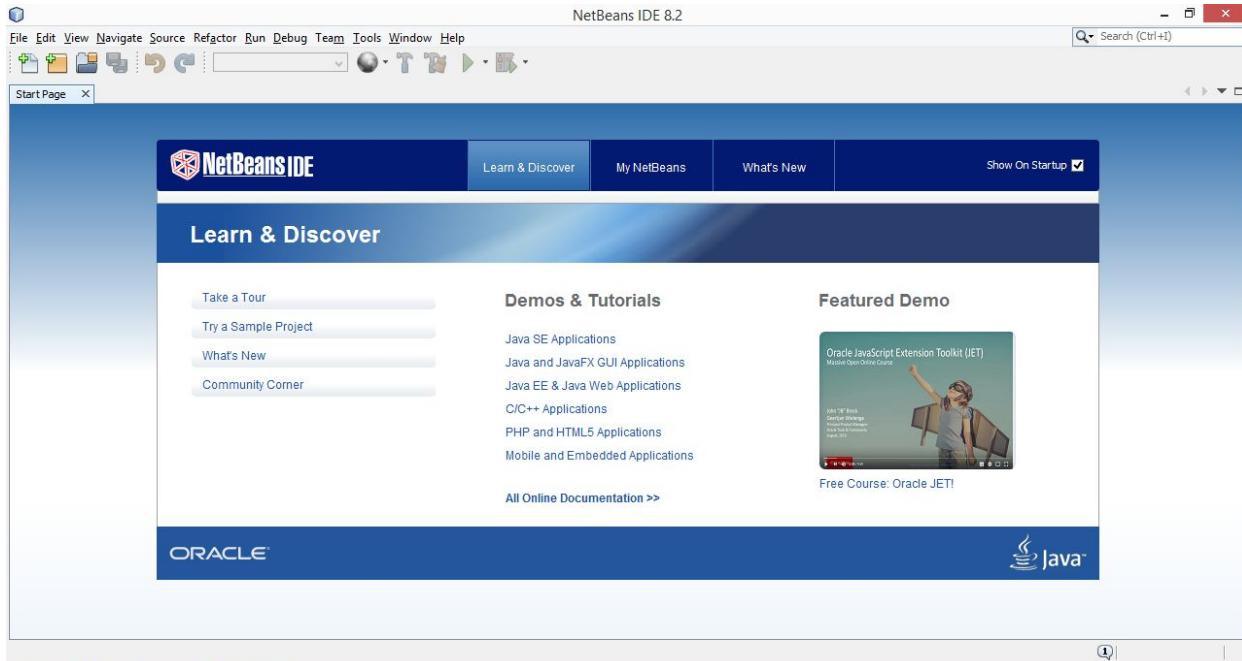
### 2. Click on DB Instances.

The screenshot shows the AWS RDS Instances dashboard. The sidebar is identical to the previous dashboard. The main area displays a table for "All Instances" with one entry: Engine: MySQL, DB Instance: cms, Status: available, Current Activity: 1.17%, Maintenance: None, Class: db.t2.micro, VPC: vpc-ba2b68d3, Multi-AZ: No. Below the table, an "Endpoint" link is shown: cms.czjoadsbahy9.us-east-2.rds.amazonaws.com:3306 (authorized). The dashboard also includes sections for "Alarms and Recent Events" (listing events like backup completion and instance creation) and "Monitoring" (showing CPU, Memory, and Storage metrics with graphs for Read IOPS, Write IOPS, and Swap Usage). The footer includes standard AWS links like Feedback, English, Privacy Policy, and Terms of Use.

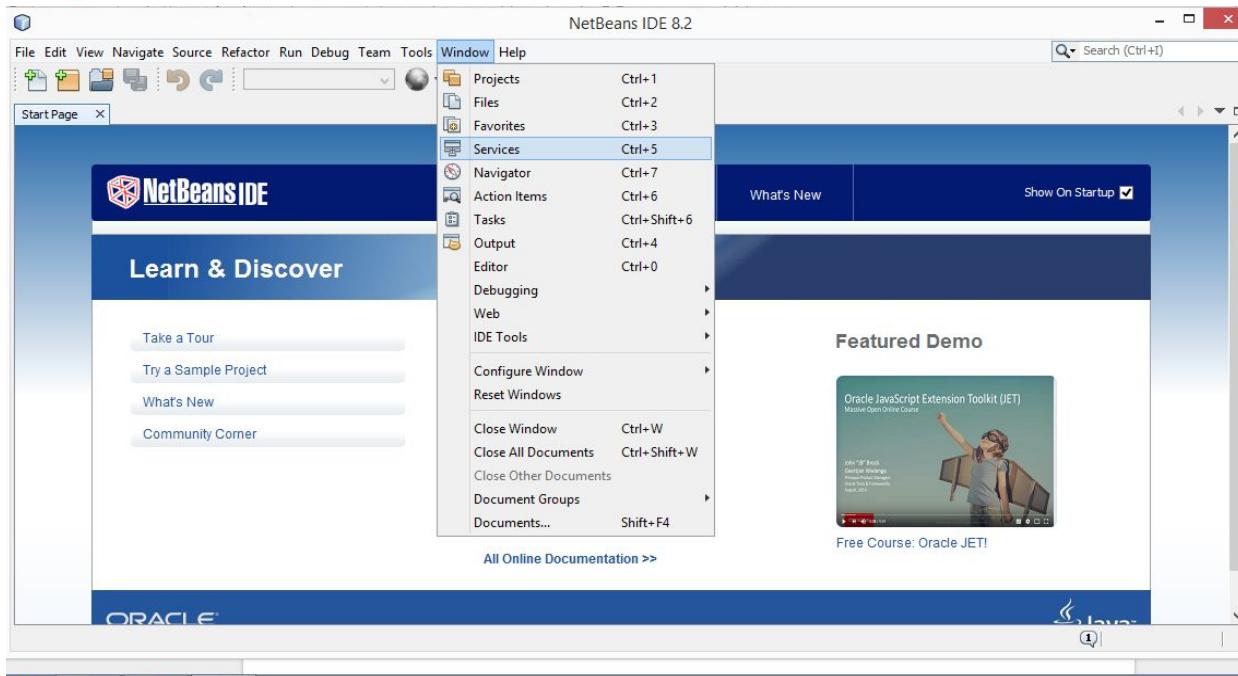
3. Endpoint of the DB Instances help us to access MySQL.

The screenshot shows the AWS RDS console with a search bar at the top. Below it is a table header with columns: Engine, DB Instance, Status, CPU, and Current Allocated Storage. A single row is selected, showing 'MySQL' as the engine, 'cms' as the instance name, 'available' as the status, and '1.17%' as the current allocated storage. At the bottom, the endpoint information is displayed: 'Endpoint: cms.czjoadsbahy9.us-east-2.rds.amazonaws.com:3306 (authorized)'. There is also an info icon next to the endpoint.

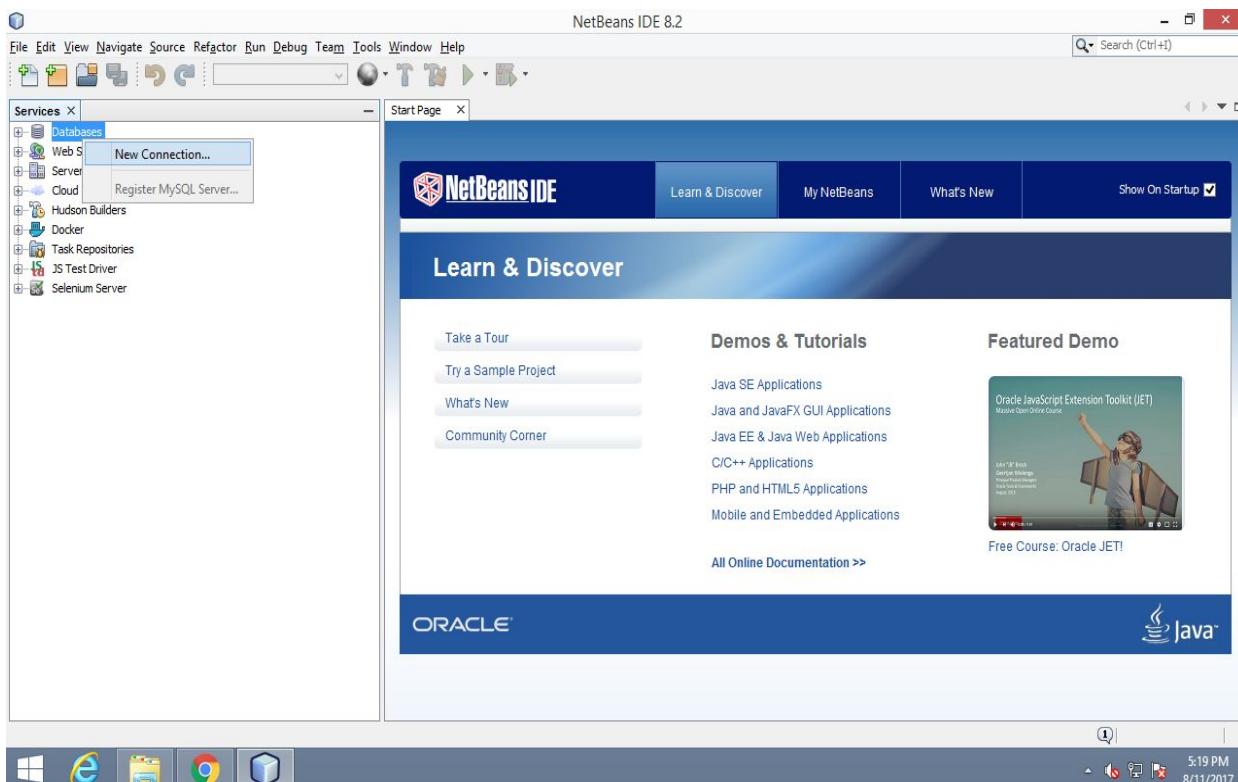
4. We can use this Endpoint to connect with MySQL Database in any programming language or tools like MySQL WorkBench, SqlWorkbenchJ, in Netbeans or Eclipse ..etc
5. We will try to connect with Netbeans. Open **Netbeans**.



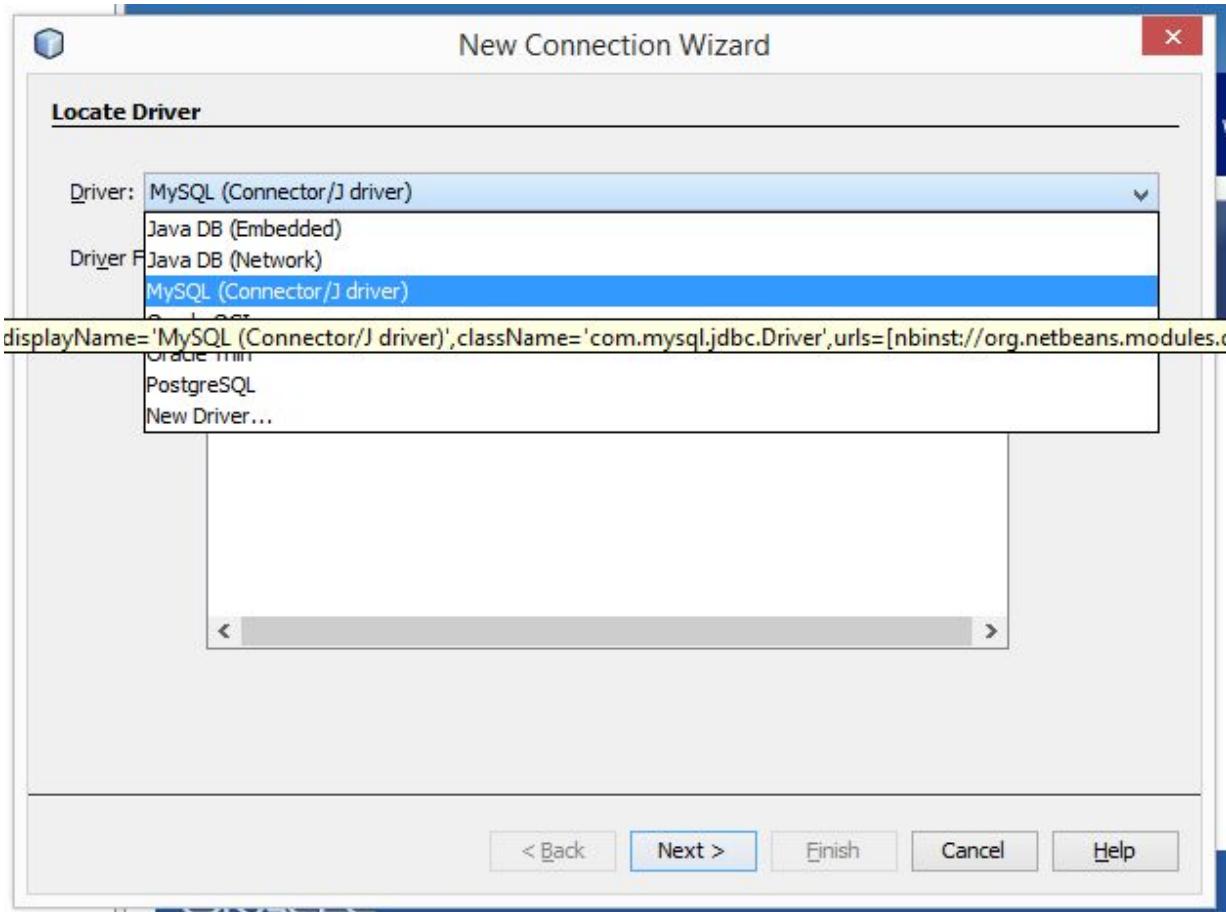
## 6. Go to Windows > Services



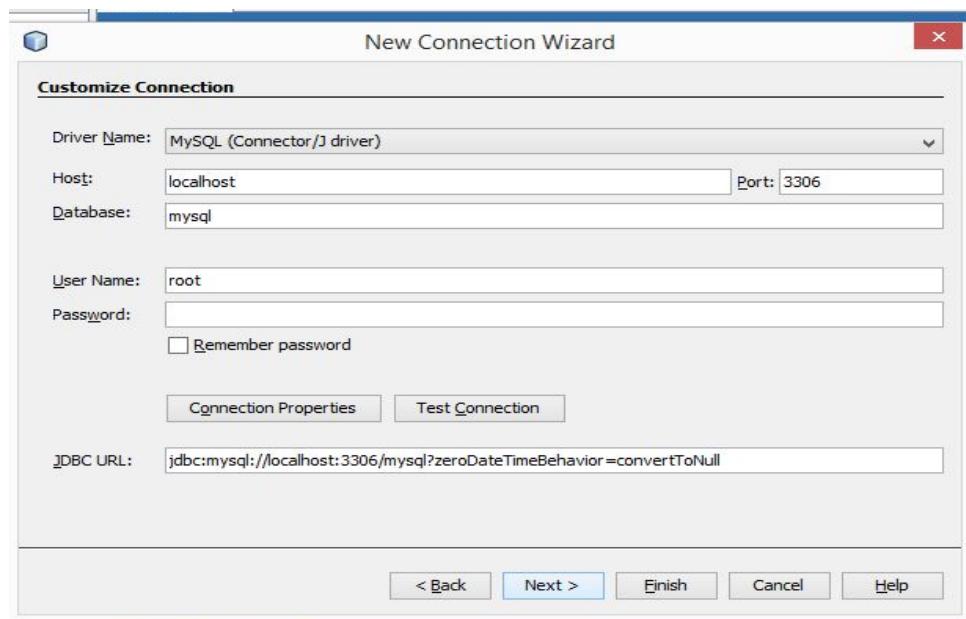
## 7. Right click on Databases > New Connection.



8. Select MySQL Driver from dropdown.



9. Click on **Next**.



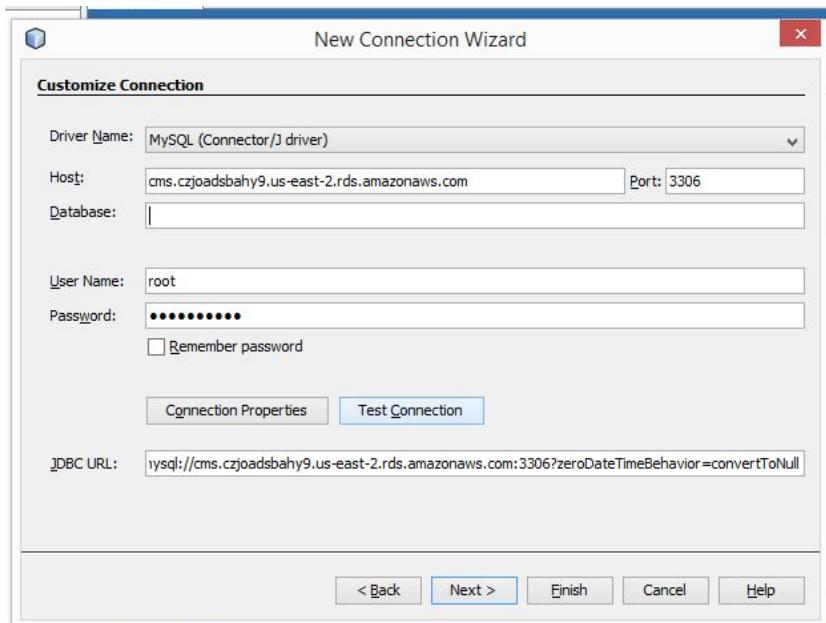
10. Provide connection details.

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

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

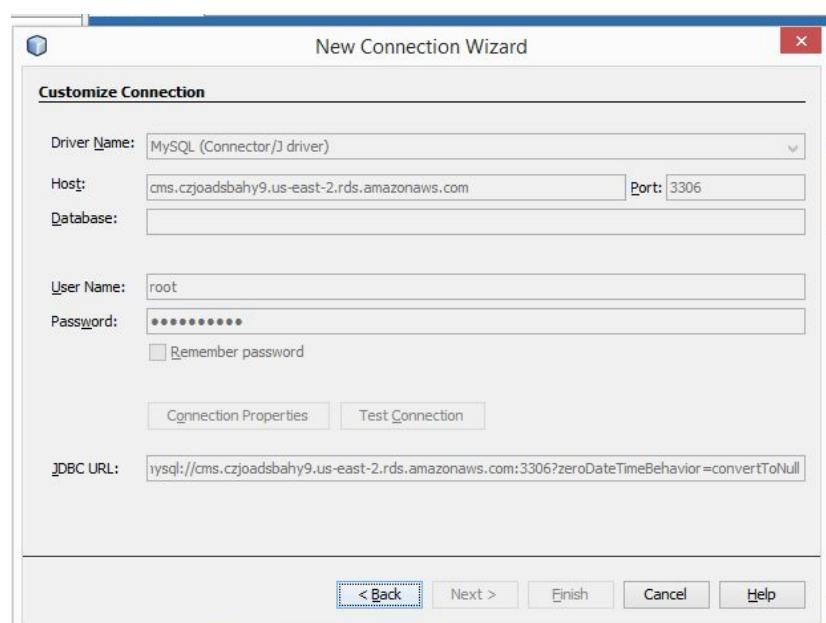
Username : Database **Username ( root )**

Password : Database **Password ( my password )**

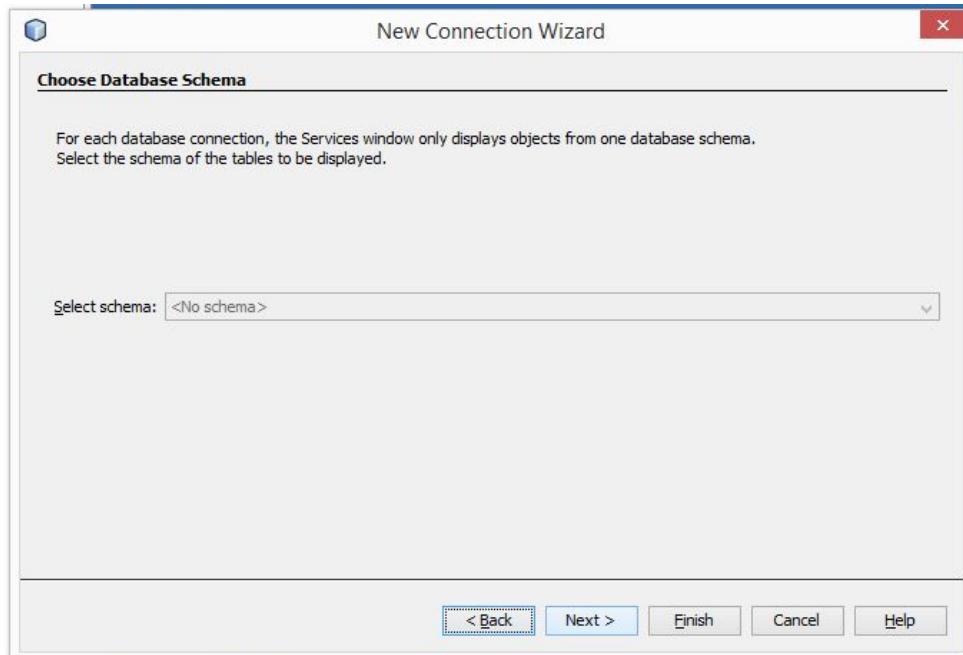


11. Click on Test Connection.

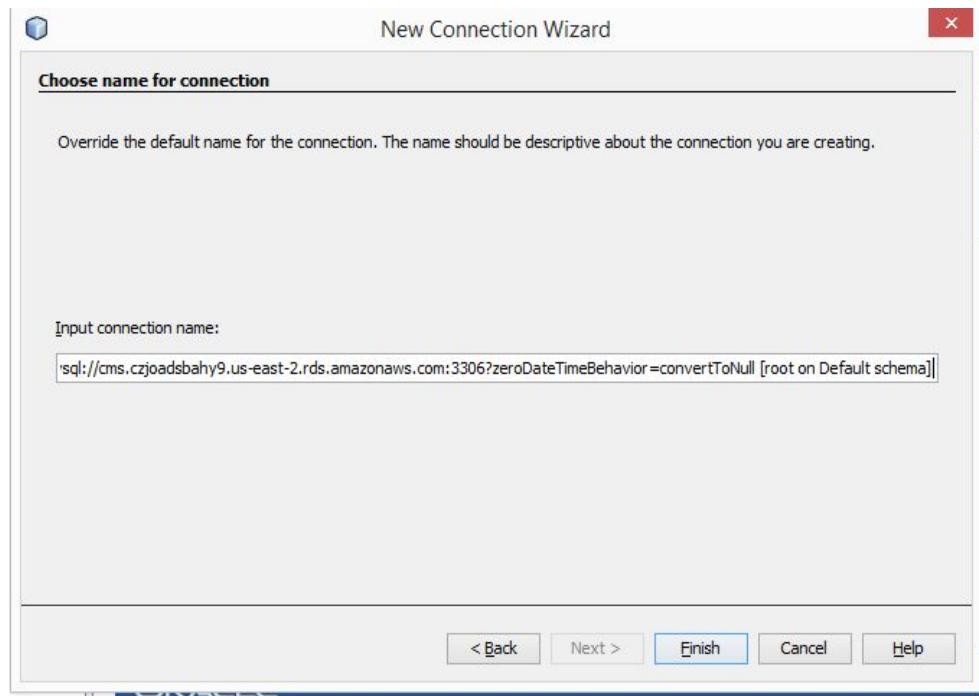
12. If Connection is successfully then click on Next.



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

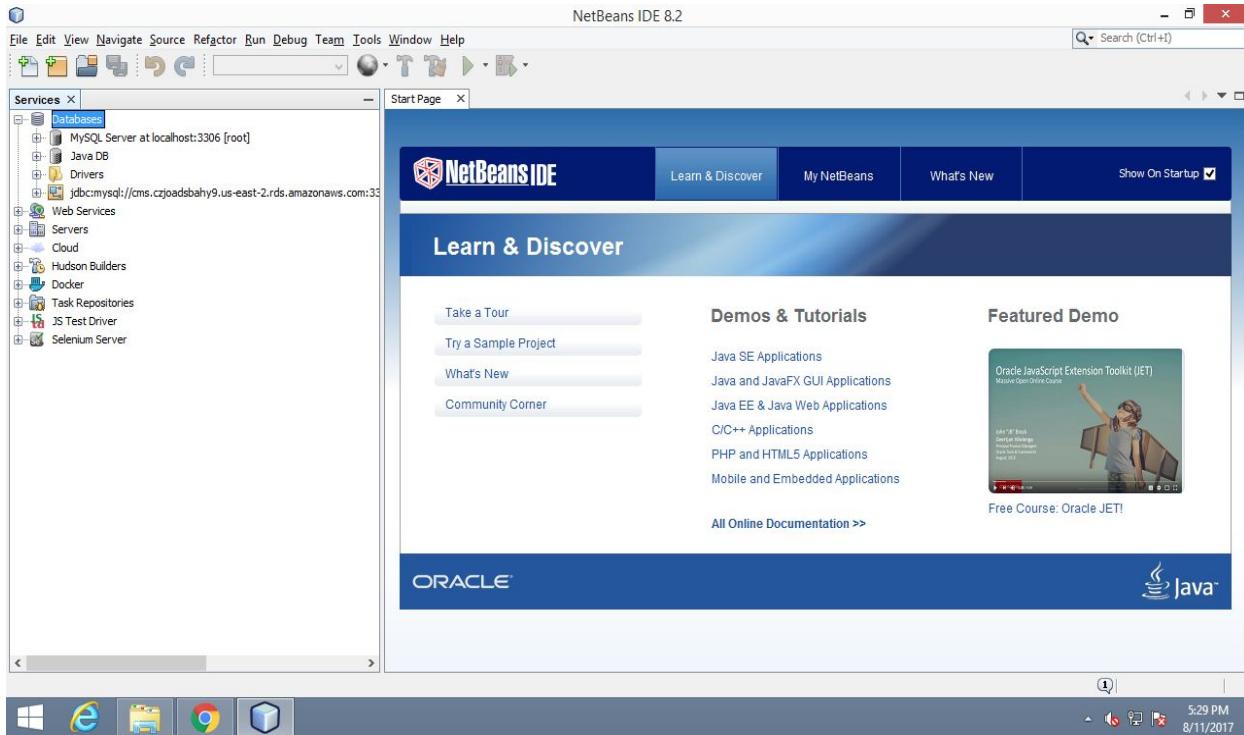


14. Click on Next.

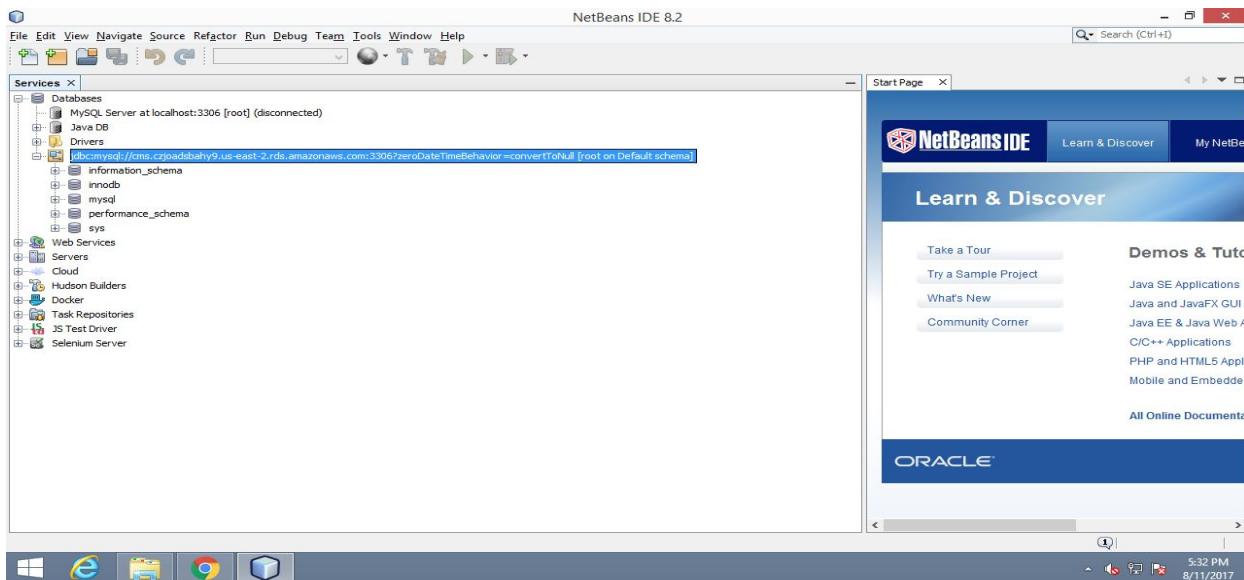


15. Click on Finish.

## 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 hands-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)