MIRACLE
SOFTWARE SYSTEMS

Deploying your First Java Application to AWS-2 (AWS EC2)

Hands-On Workshop | Digital Summit '18

Miracle Innovation Labs

Miracle Software Systems, Inc.

Deploying your First Java Application to AWS-2 (AWS EC2)

Introduction

The goal of this document is to show you how to create an EC2 instance, install and configure all the necessary software for deploying and running a sample application onto a web server.

This guide was prepared by **Miracle's Innovation Labs!**

Pre-Requisites

All attendees must have their workstation (with Internet) to participate in the workshop (Both PC and MAC are compatible). The following pre-requisites will help you to make the workshop experience easier.

- AWS account
- Download and install PuTTY
- Download and install Tomcat

Technology Involved

- AWS account
- Java
- Apache Tomcat
- PuTTY (for windows)
- Git

Lab Steps

Let us get start the Workshop!

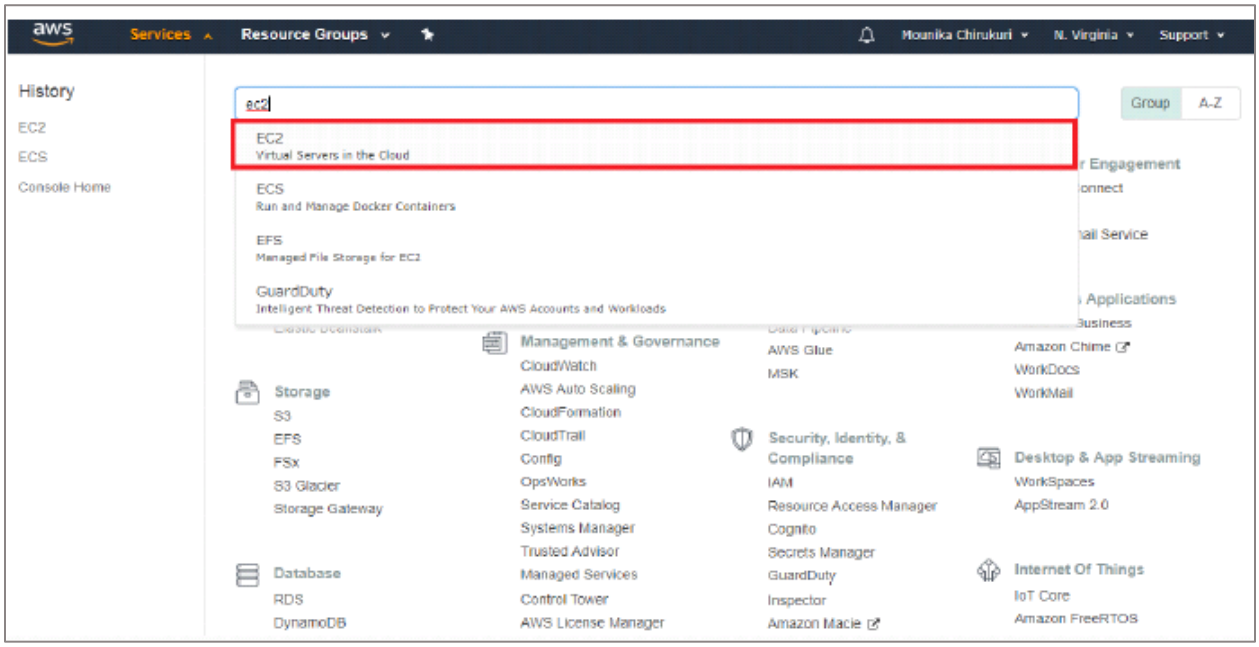
We will show you how to login into AWS Management Console and spin EC2 instances which are virtual machines. These virtual machines are managed by AWS and are called Elastic Compute Cloud. While spinning the instances, we will perform security group configurations which acts as firewall at instance level, generate private key which will be used to login to the instance.

Installing Java which is a basic requirement for running Apache Tomcat Server.

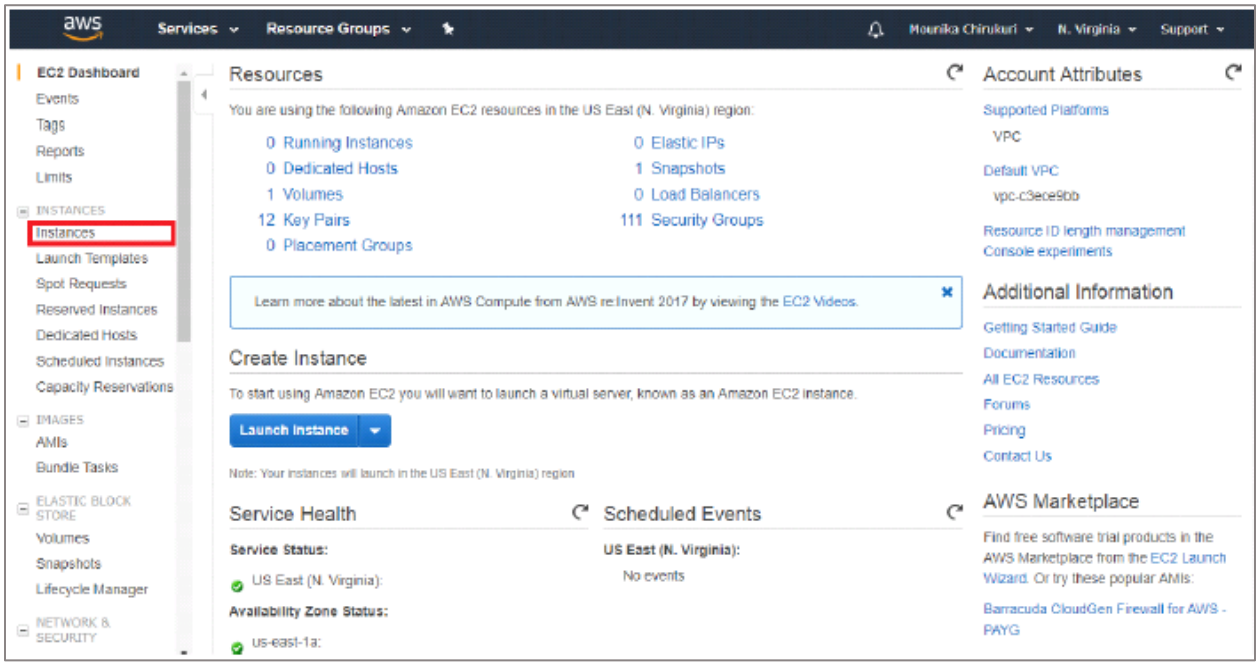
Configure Apache Tomcat on port 80, login user configurations with specific username and password. Once basic configurations are done, we will show you how to deploy the application onto Web Server Apache Tomcat.

Step #1 | AWS EC2 Instance Creation

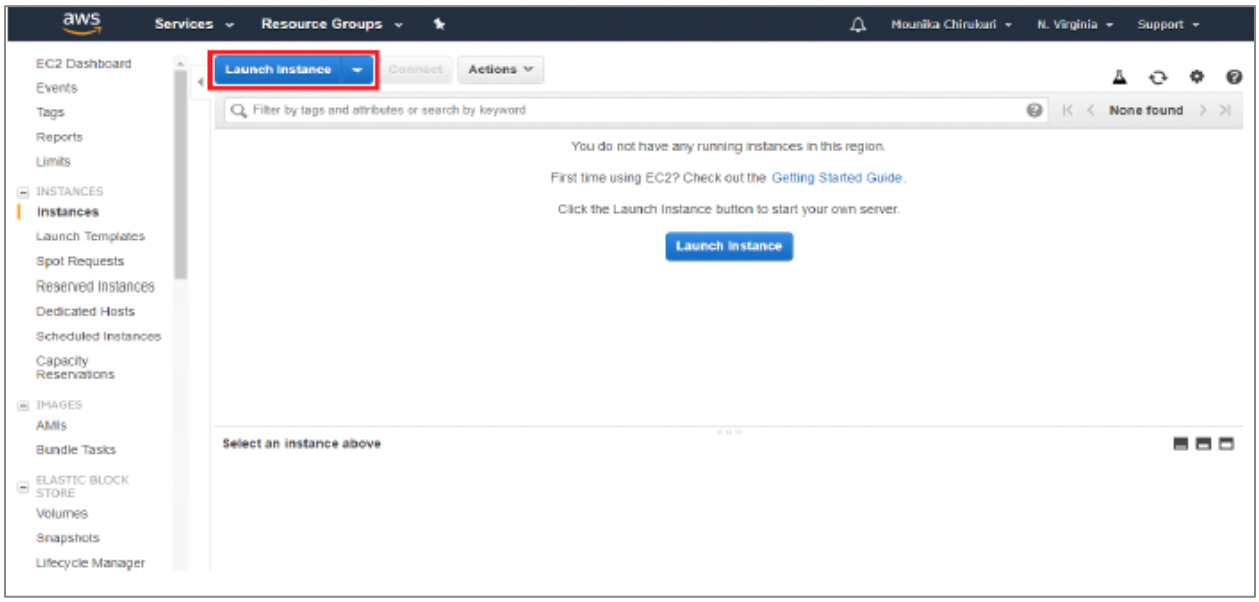
Goto AWS console and enter EC2 in the search bar as shown below and select the EC2 from the list of AWS services.



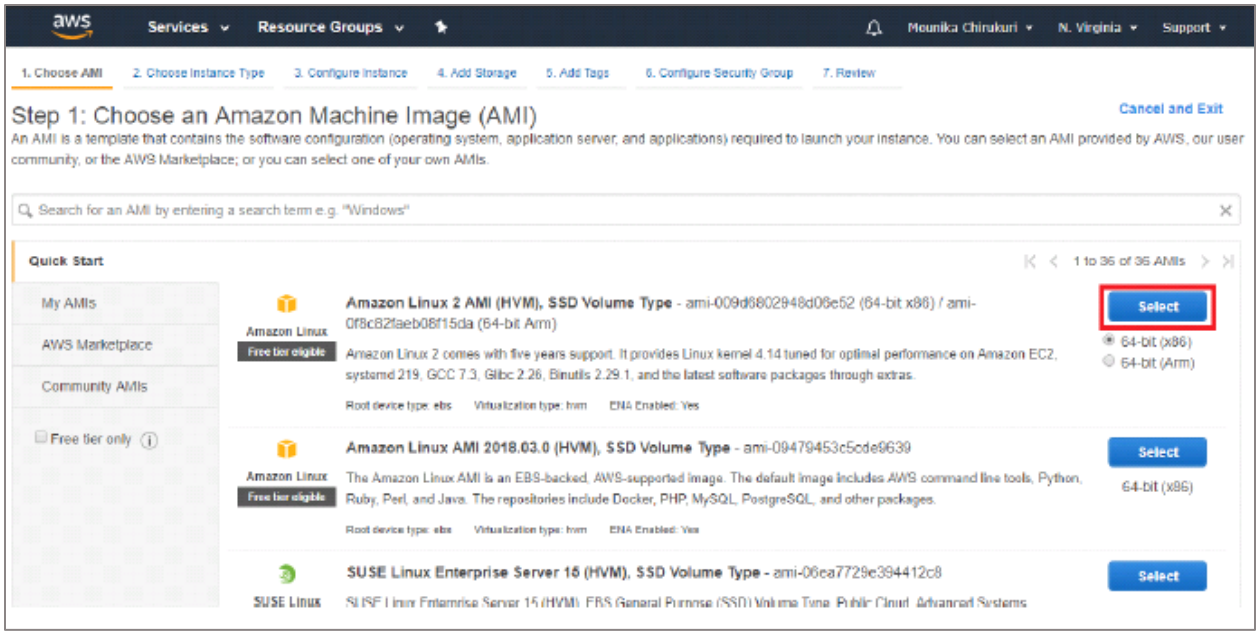
Now click on **Instances** which is at the left side menu.



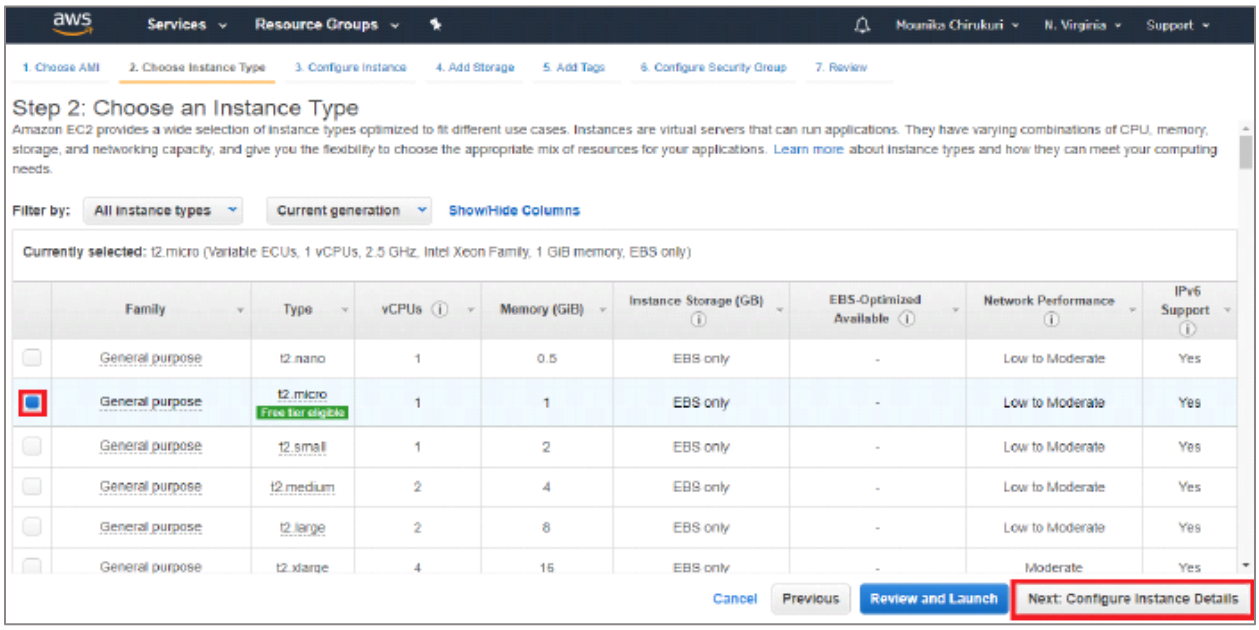
Click on **Launch Instance.**



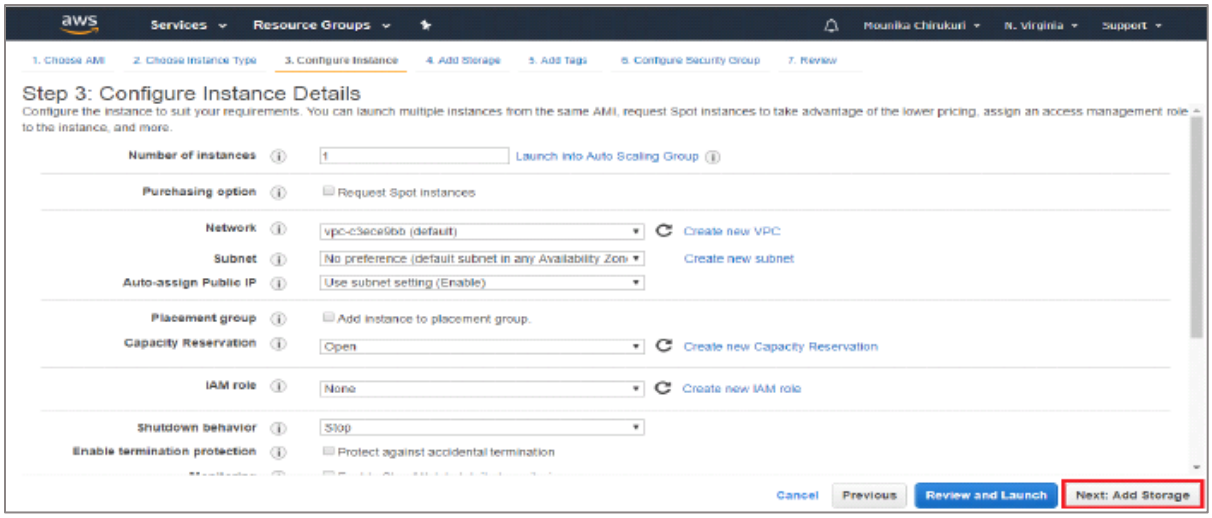
Click on **Select** for the **Amazon Linux 2 AMI (HVM)** option.



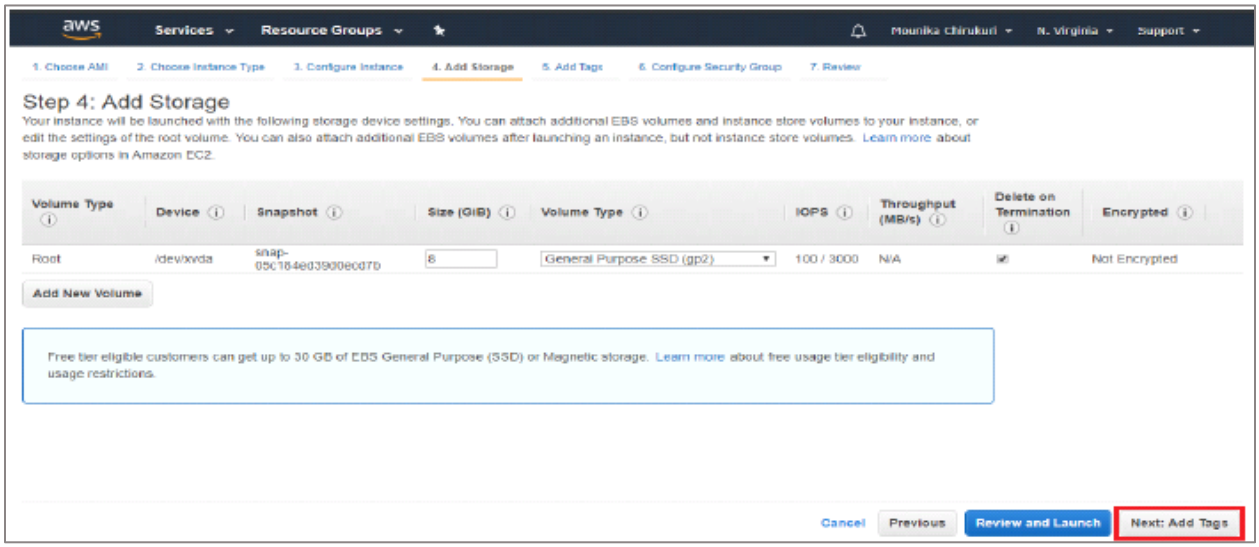
Select Instance type as **t2.micro** which is eligible for free tier and click on **Next: Configure Instance Details.**



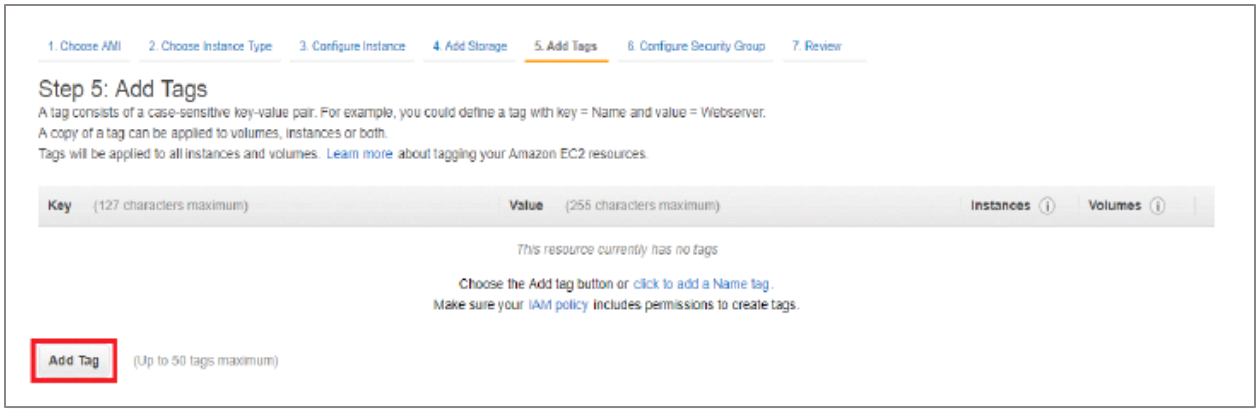
In this page, you can configure Number of instances, VPC, Subnet, Public IP etc., for now leave the default values and click on **Next: Add Storage.**



In this page you can configure volume and for now leave the default values and click on **Next: Add Tags.**



Click on **Add Tag** to give custom name for the instance.

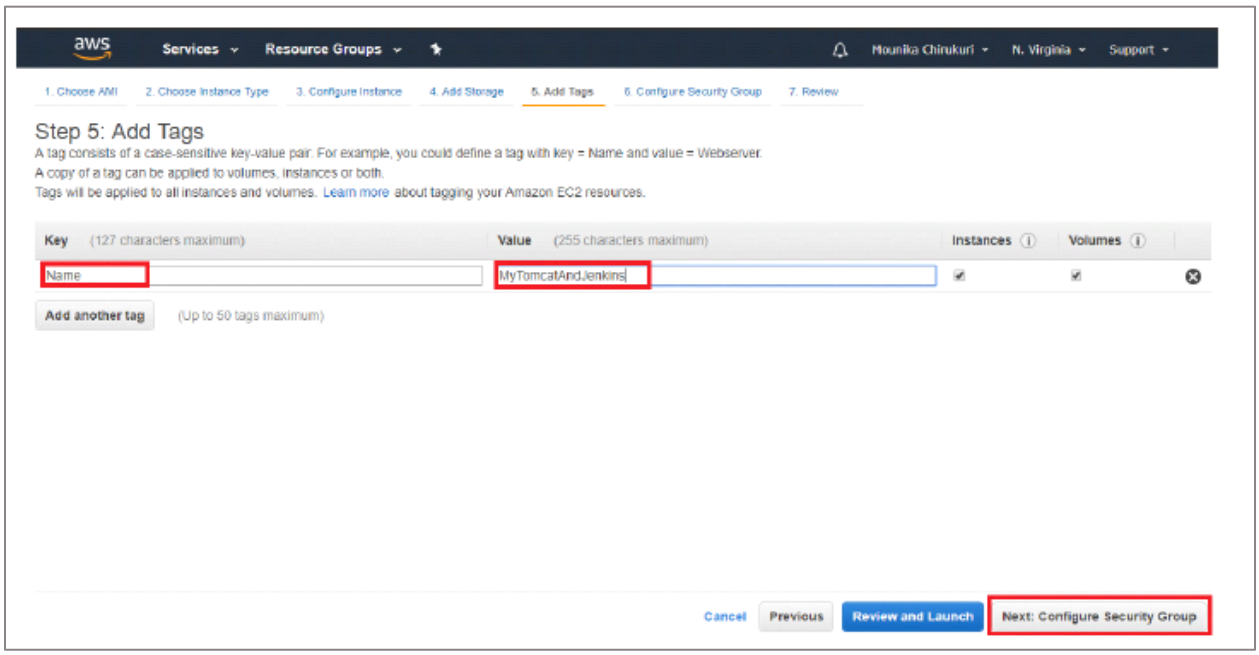


Enter the below values.

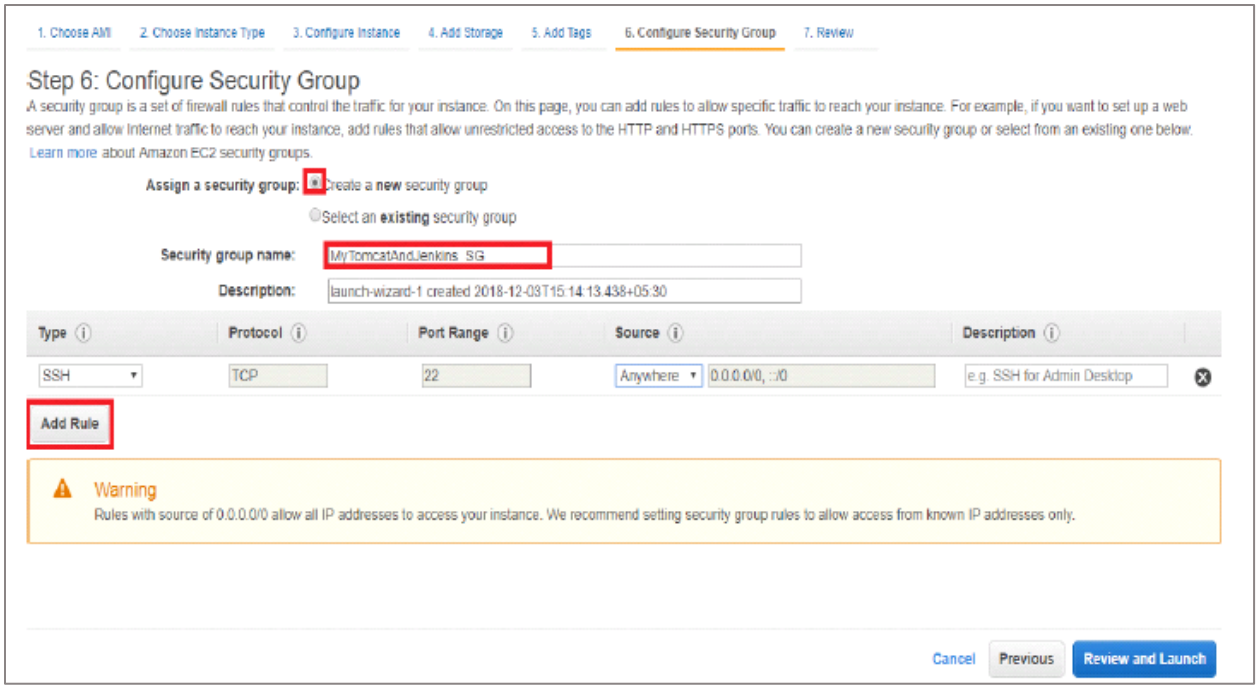
Key : Name

Value :<your-default-name>

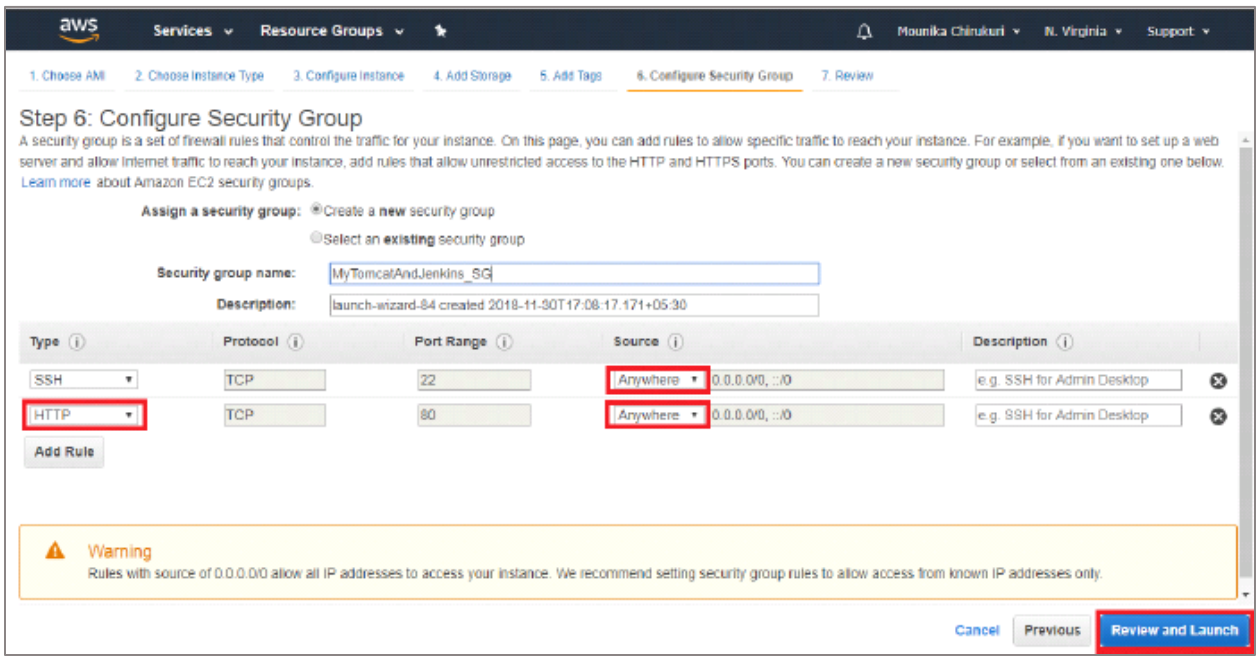
Click on **Next : Configure Security Group**



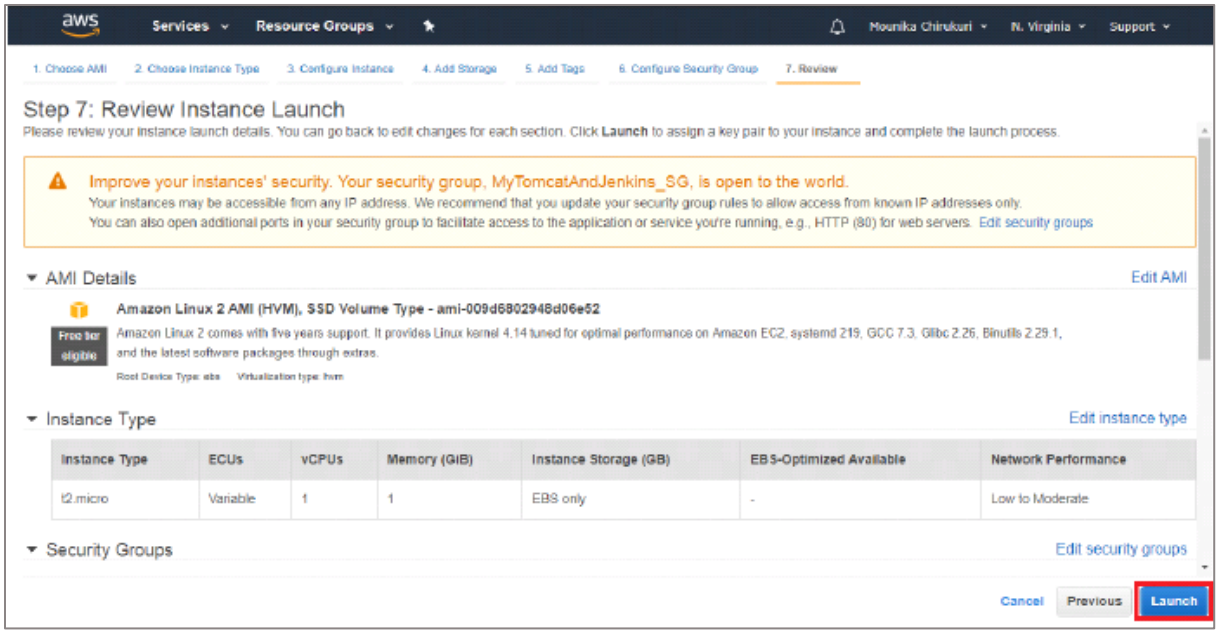
Select **Create a new security Group** and enter name for security group and click on **Add Rule** to open ports.



Select **HTTP** and **Source** as Anywhere, and click on **Review and Launch.**



If you want to modify anything or review, you can check here and click on **Launch.**

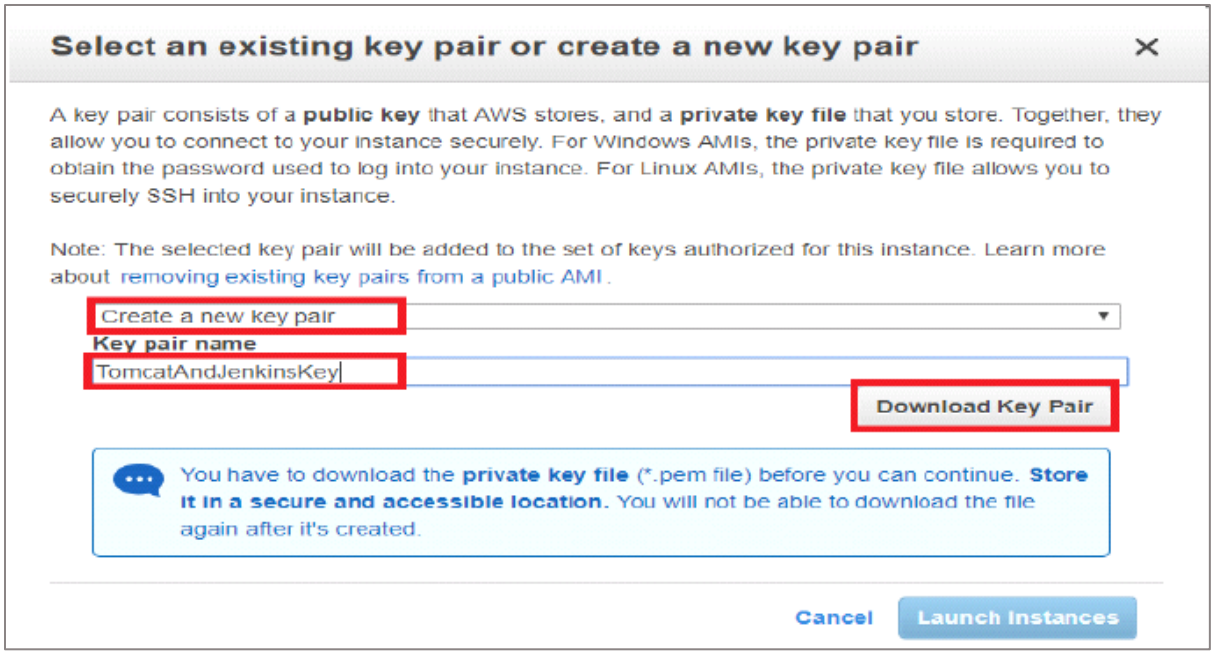


You need to create a key pair for your instance to SSH into instance. For that select **Create a new key pair.**

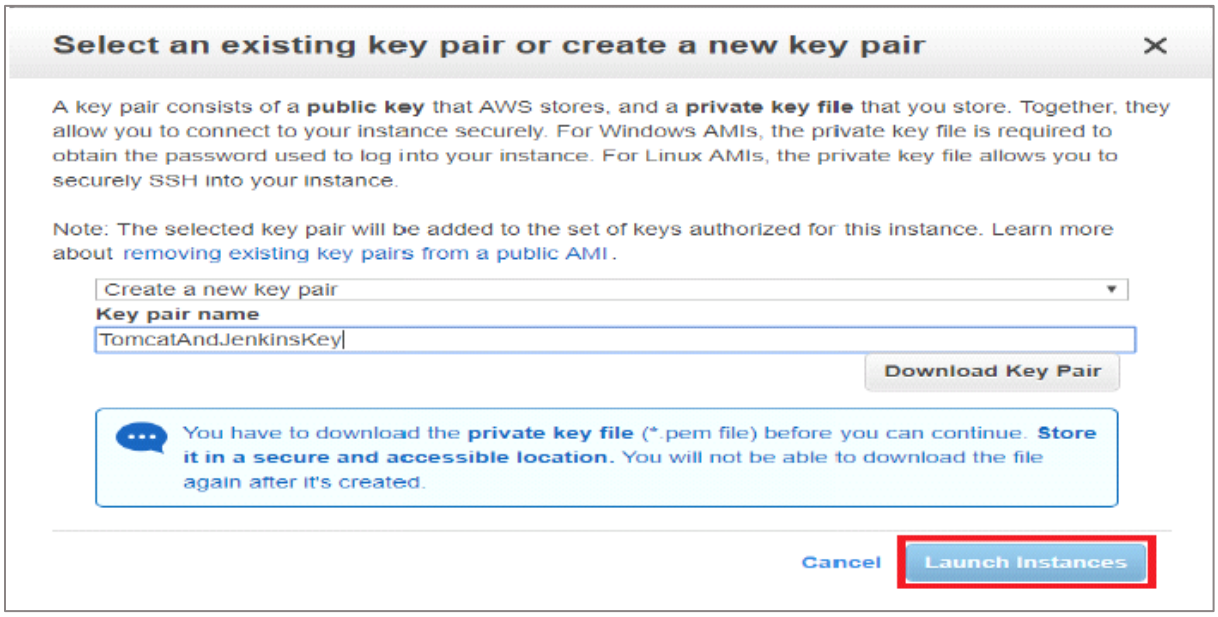
Key pair name : <your-key-pair-name>

Click on **Download Key Pair.**

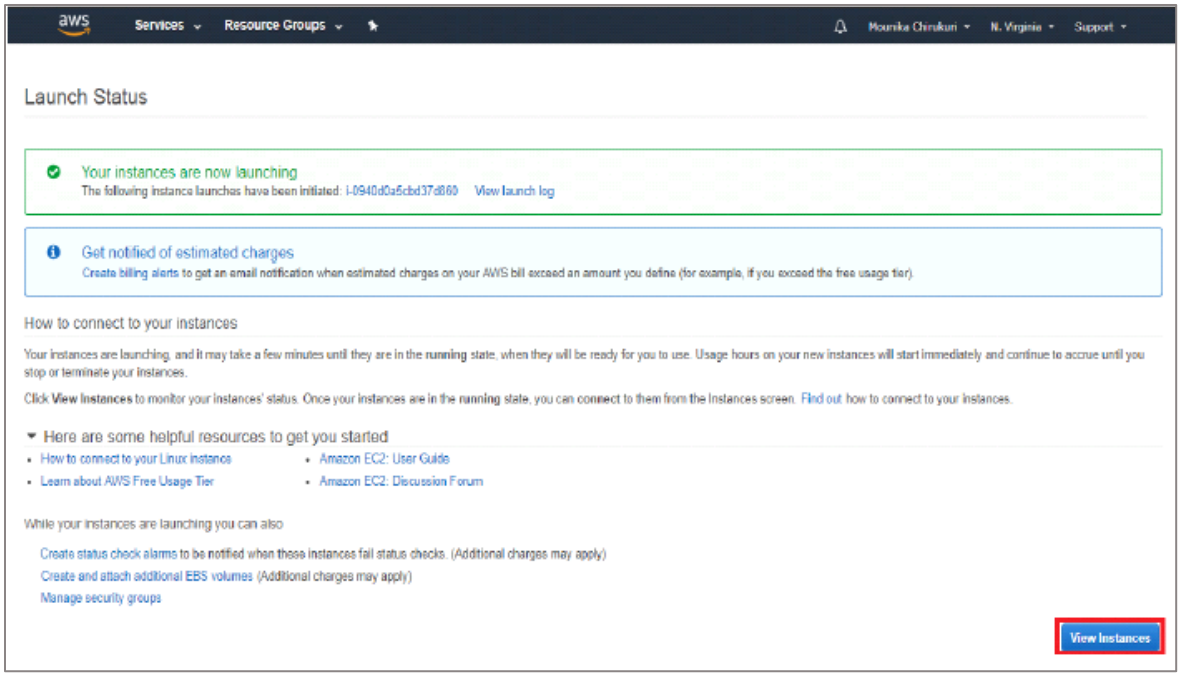
Note - If you forget to download this .**pem** file, you will not be able to SSH to this instance.



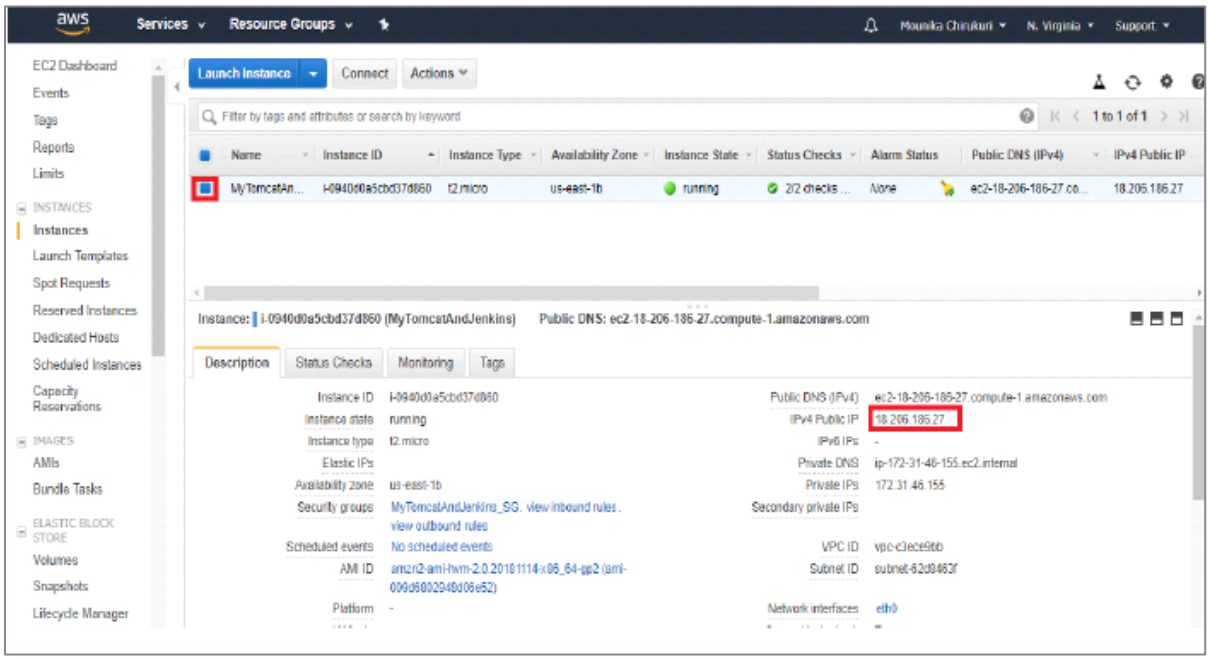
Click on **Launch Instances.**



Click on **View Instances** to navigate to EC2 home page.

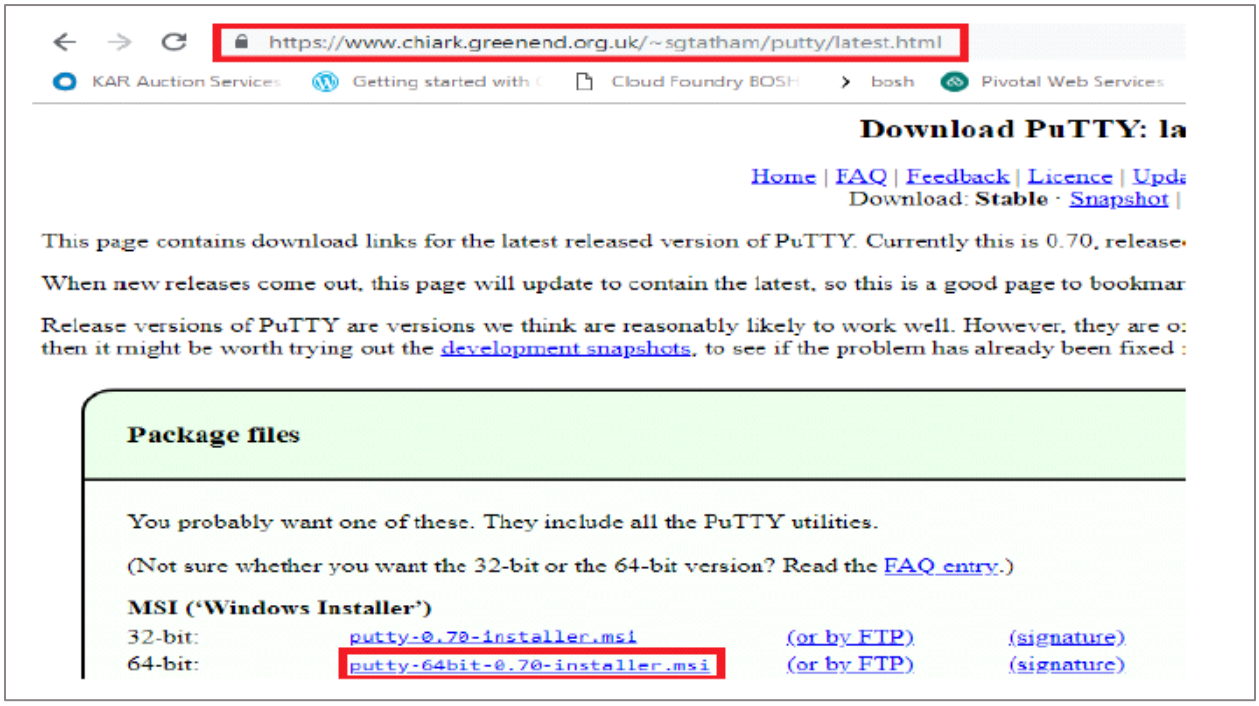


After navigating to EC2 home page, select your instance and check for the **IPV4 Public IP** which is at under **Description** tab.

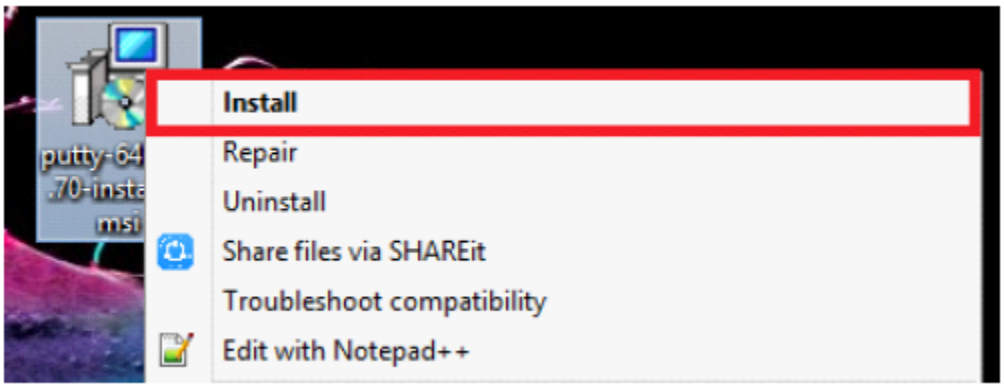


Step #2 | Installing Putty

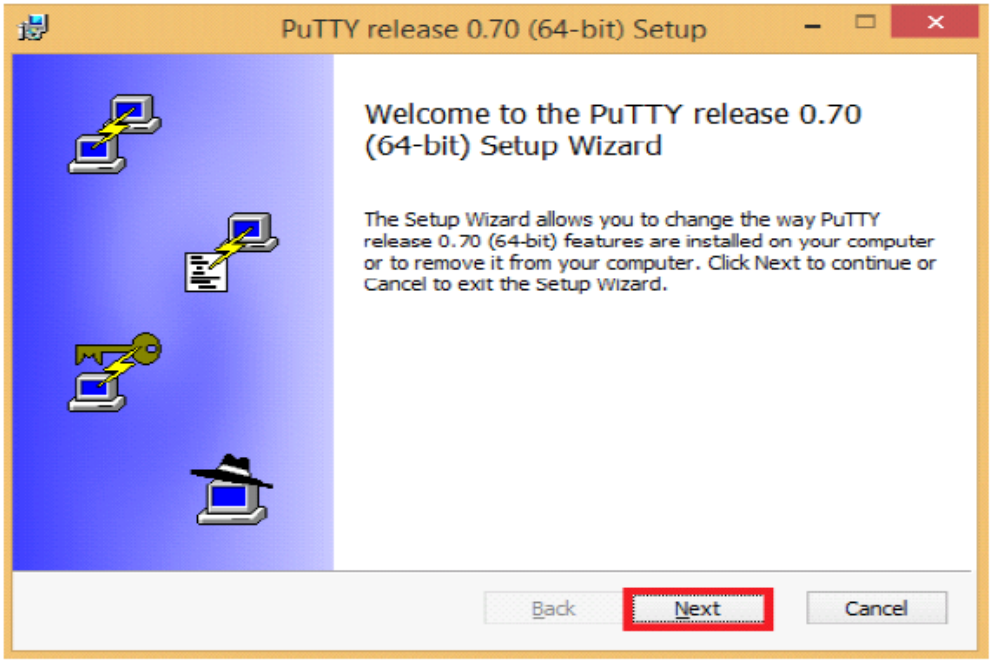
Goto **https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html** and download **putty msi installer** by clicking on the download link.



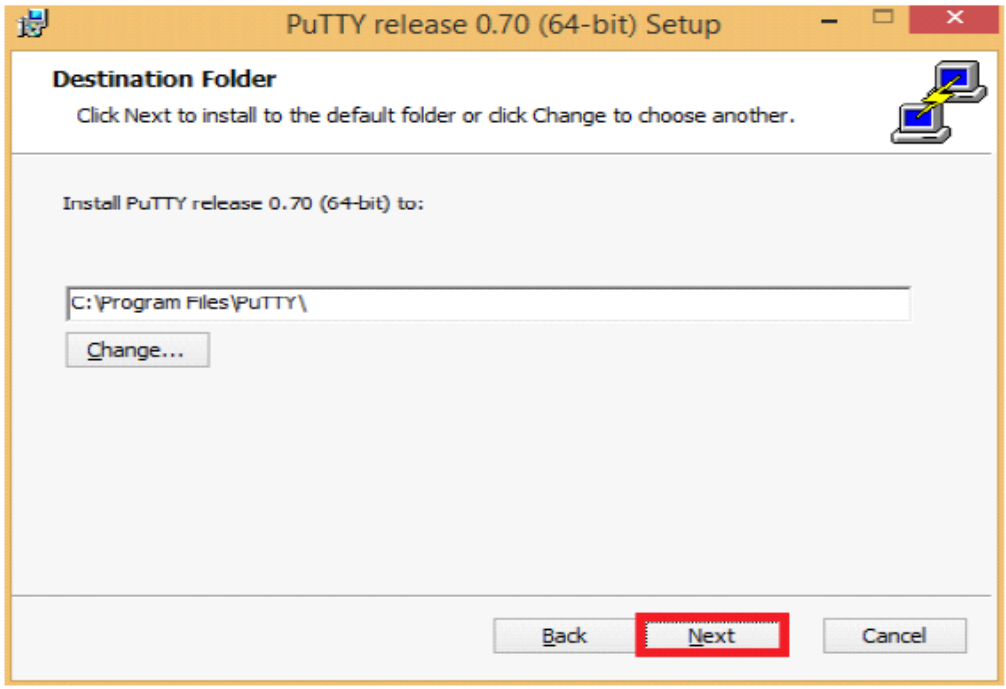
Right click on the Installer and select **Install** option.



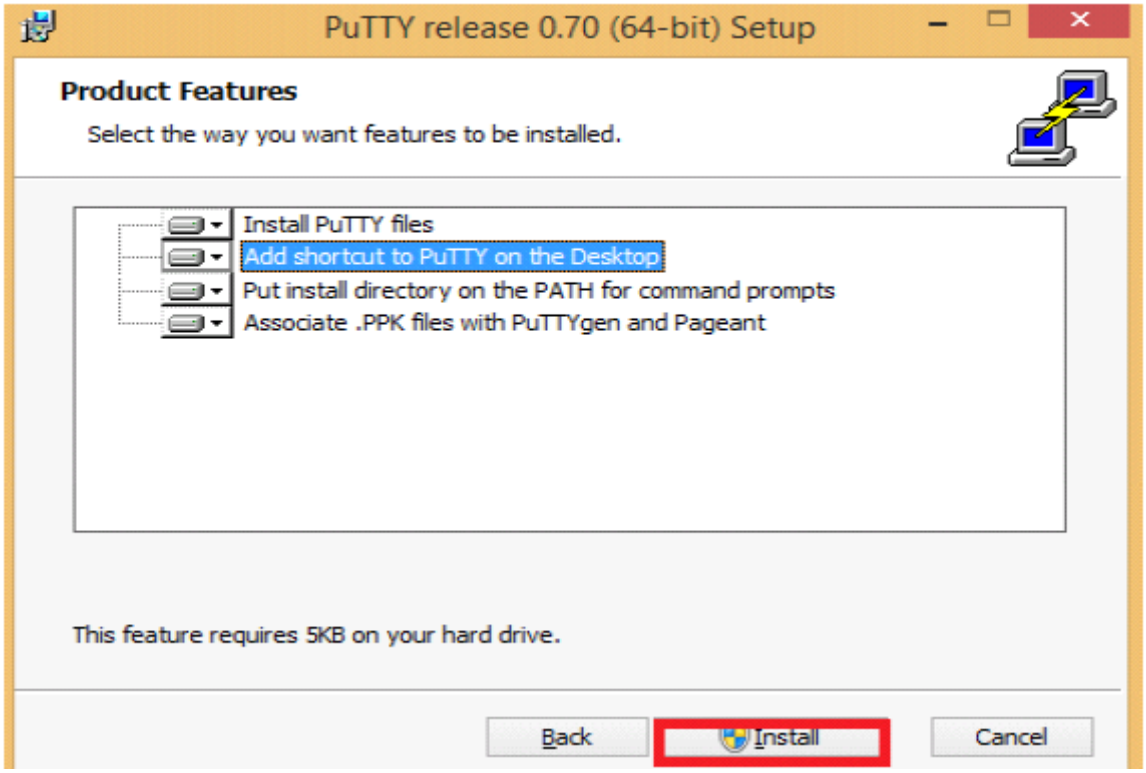
After selecting install option, the setup page will be displayed. Click on **Next**.



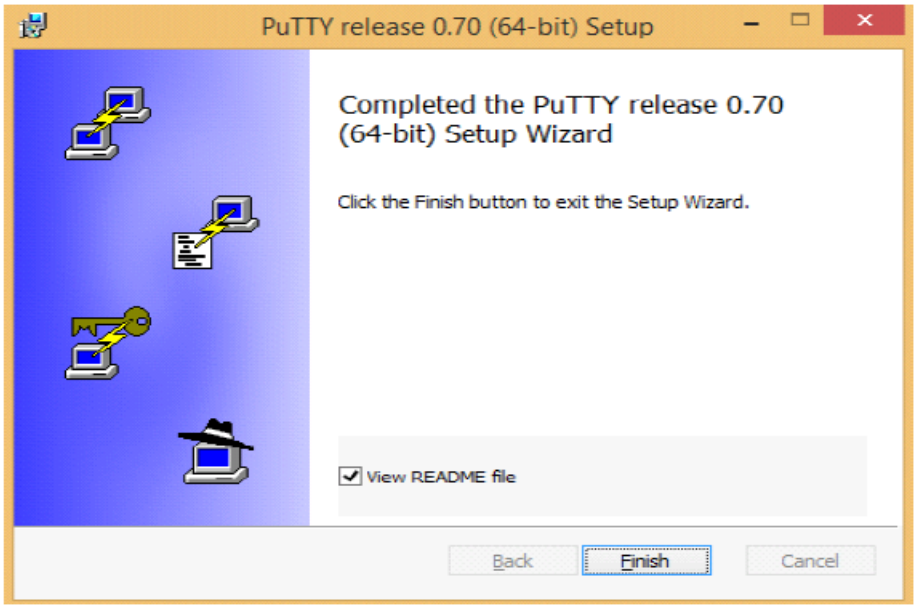
Select the location where putty has to be installed and click on **Next**.



Now select install option to install the application as shown below.



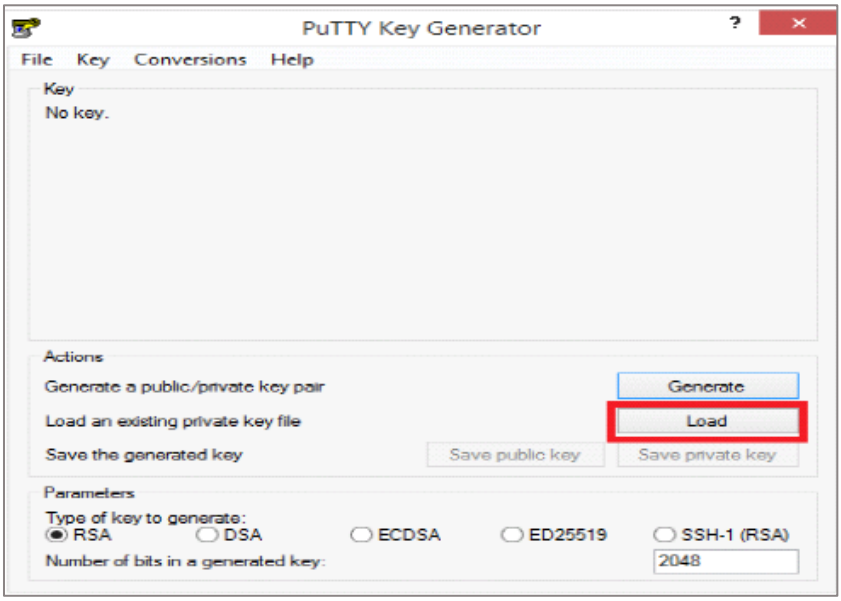
Select **Install** option it will prompt you to confirm installation with yes or no. Select **yes** to continue. Once the installation is completed, click on Finish.



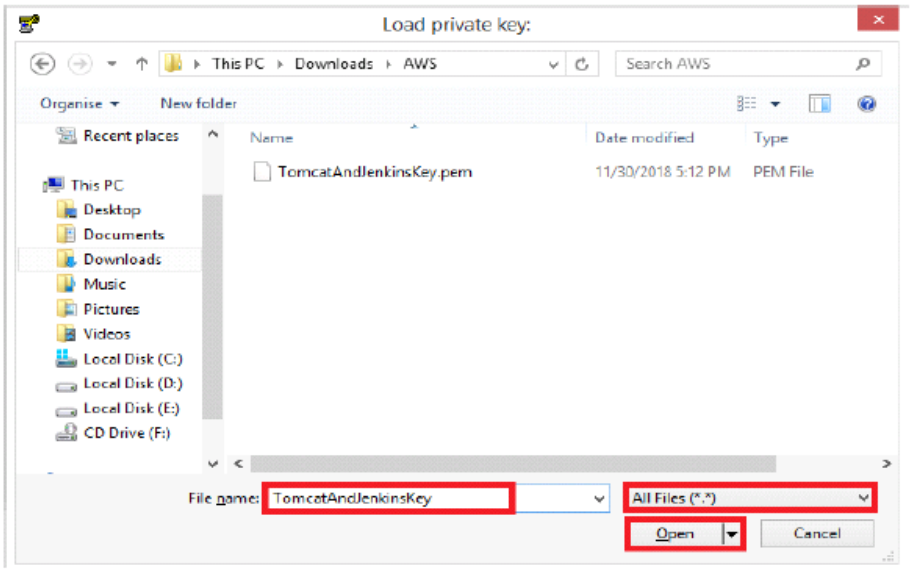
Step #3 | Conversion of .pem to .ppk

If you are using windows to connect to the instance, you need to install PuTTYgen and convert .pem file to .ppk file as shown below.

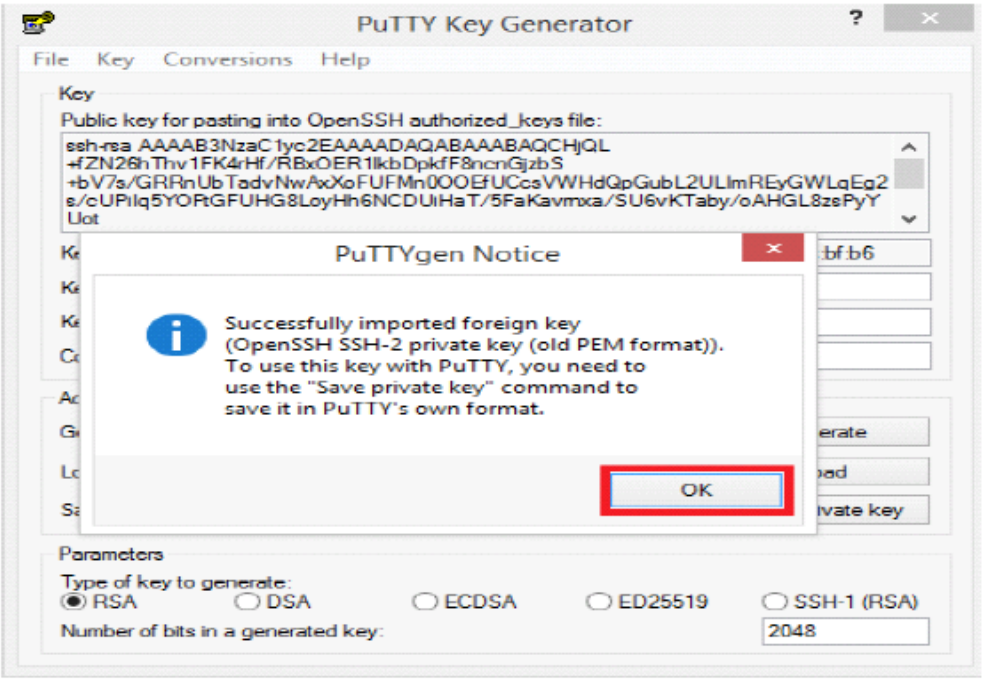
Open PuTTYgen and click on **Load.**



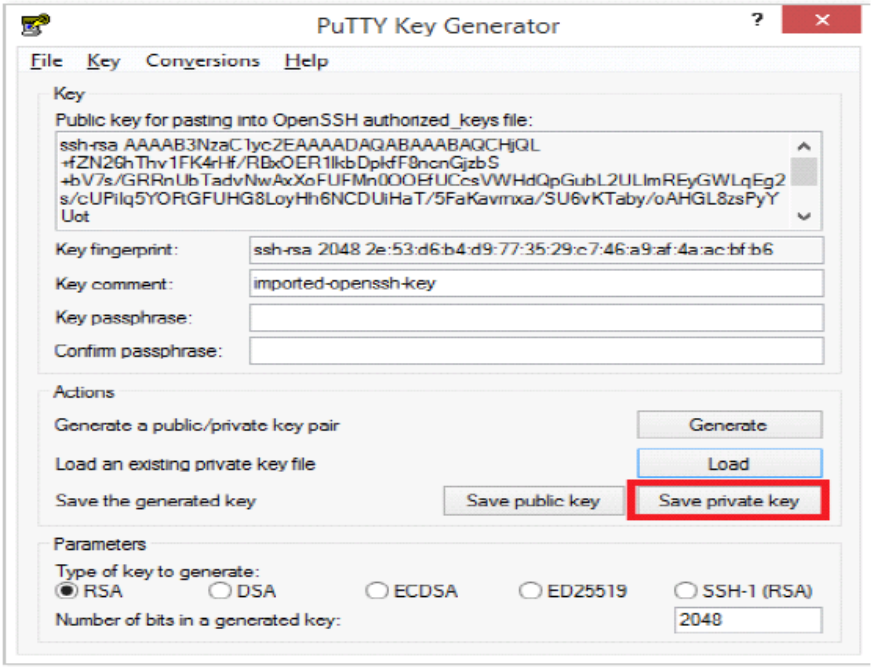
Search with your .pem file and change the file type to **All Files** as shown below and click on **Open.**



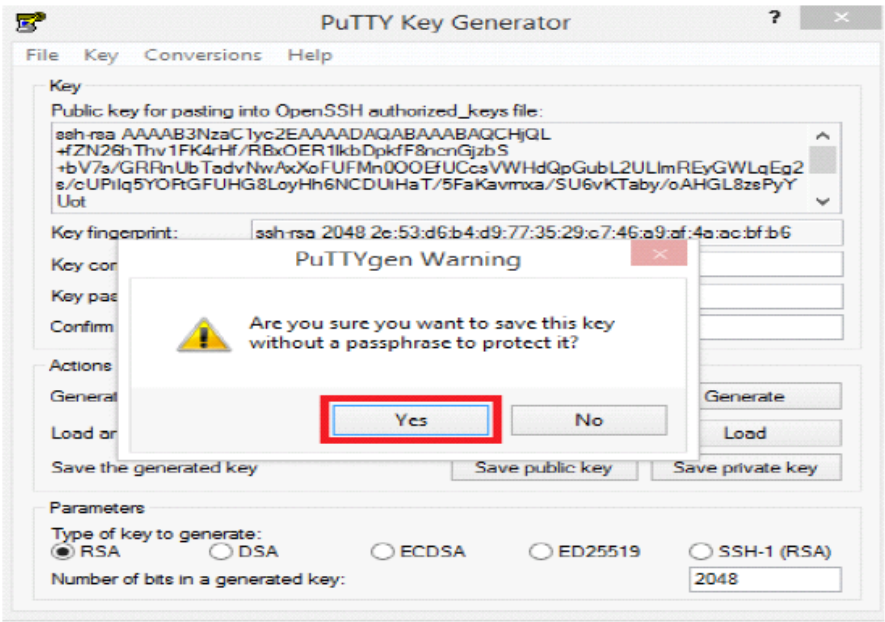
A popup is displayed on successful upload. Click on **OK** to continue**.**



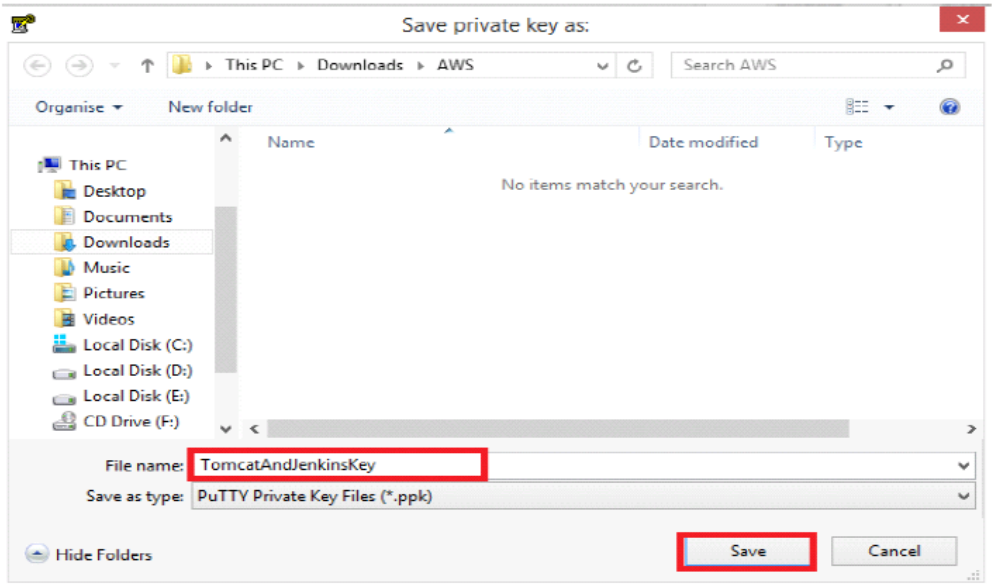
Click on Save private key to get .ppk file to your local machine.



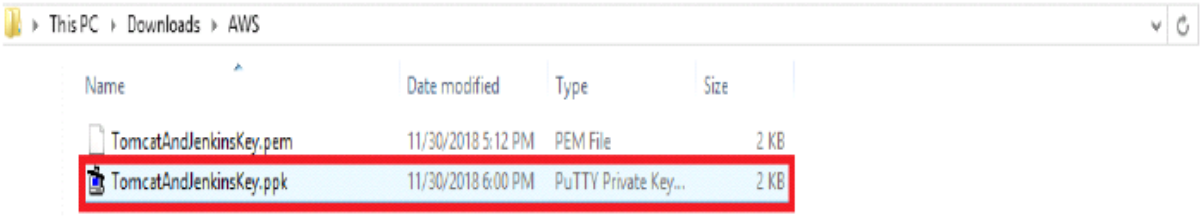
Confirmation popup is displayed before saving the .ppk file. Click on **Yes**.



Enter name for .ppk file and click on Save.



After downloading .ppk into your local machine, the file is as shown below.

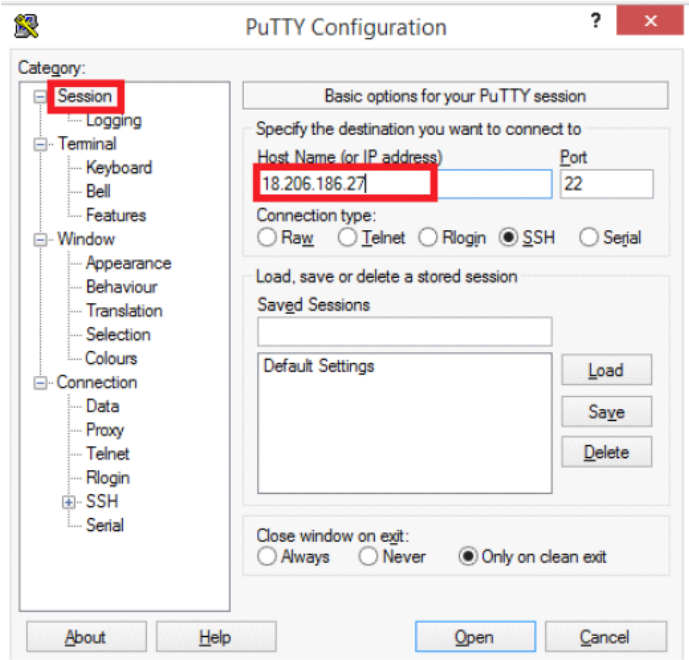


Step # 4| SSH to your Instance

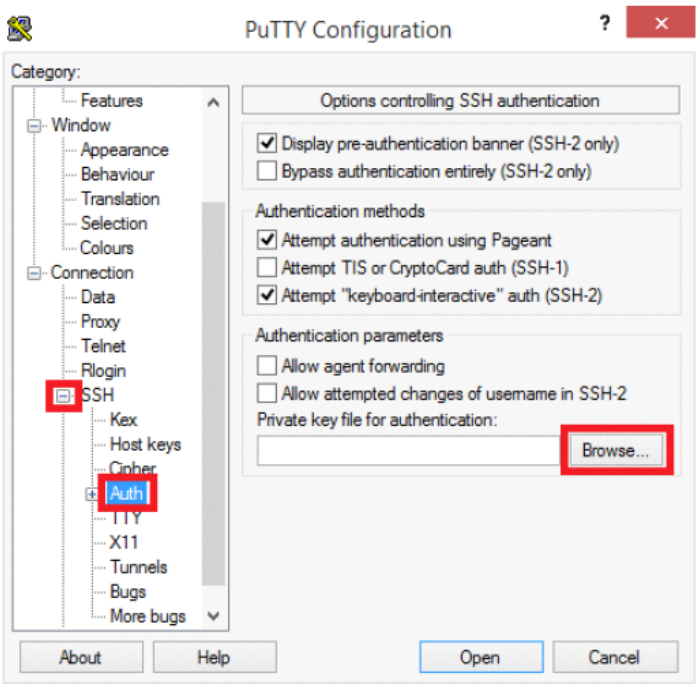
To connect to your instance, install PuTTY.

Open PuTTY and click on Session which is at the left side menu and enter Public IPV4 of the instance in Host Name (or IP address).

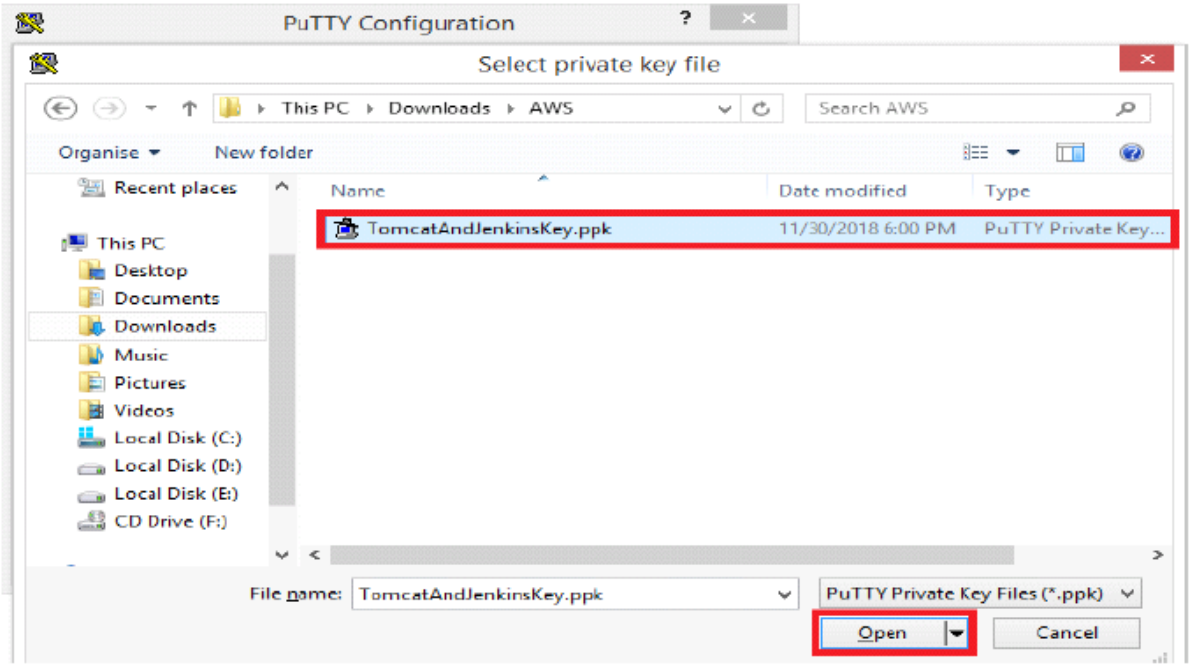
Note - Here Host Name means IPV4 Public IP which is discussed in #AWS EC2 instance creation.



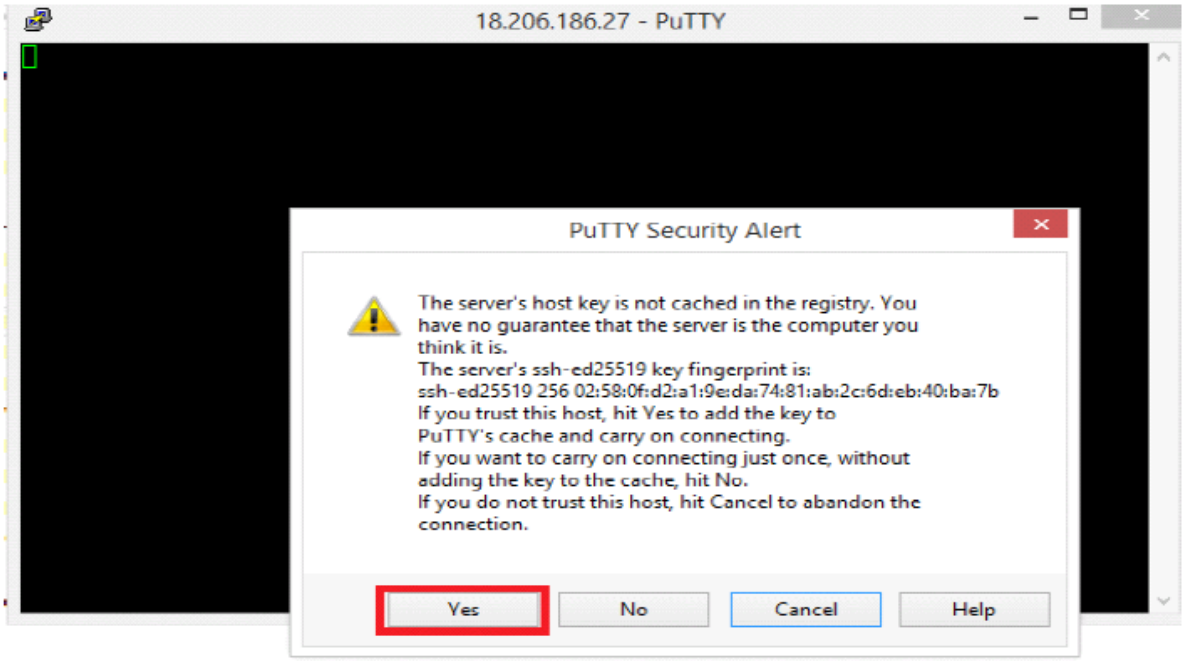
Click on **+SSH** in the left menu. Thereafter, click on **Auth.** You can select your .ppk file by clicking on **Browse.**



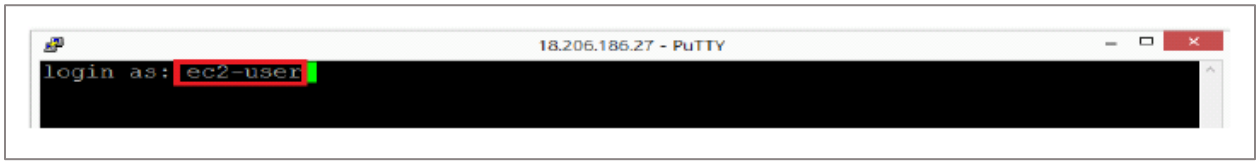
Select your .ppk file and click on **Open**.



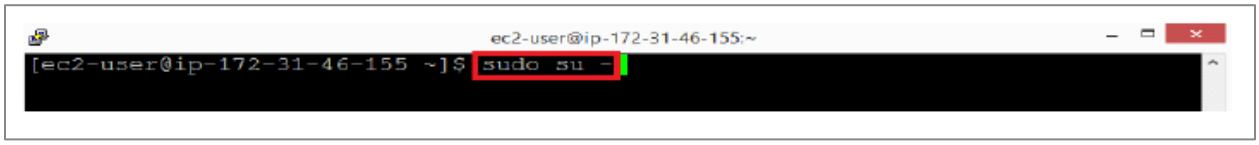
A security alert will be displayed, click on **Yes.**



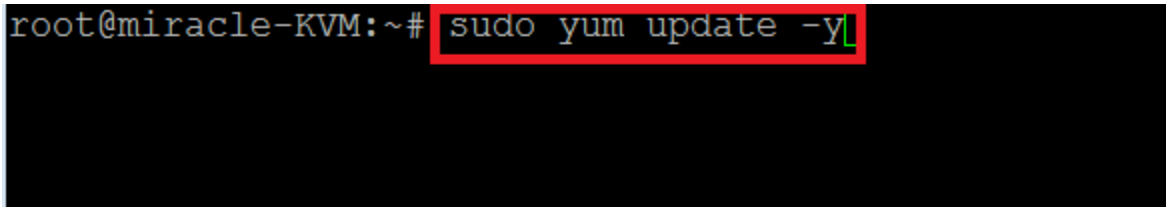
Enter **ec2-user** as user name as it is a Linux instance.



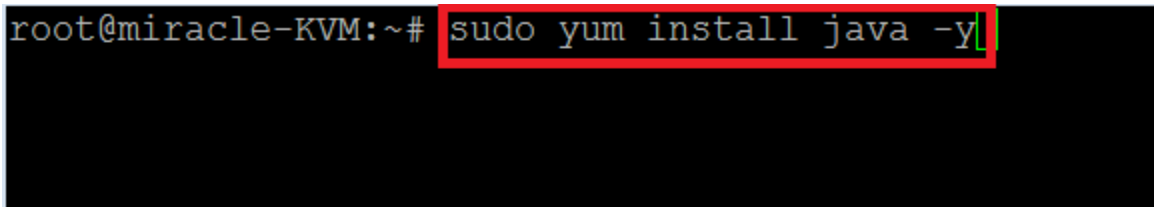
To switch to root user enter **sudo su -**



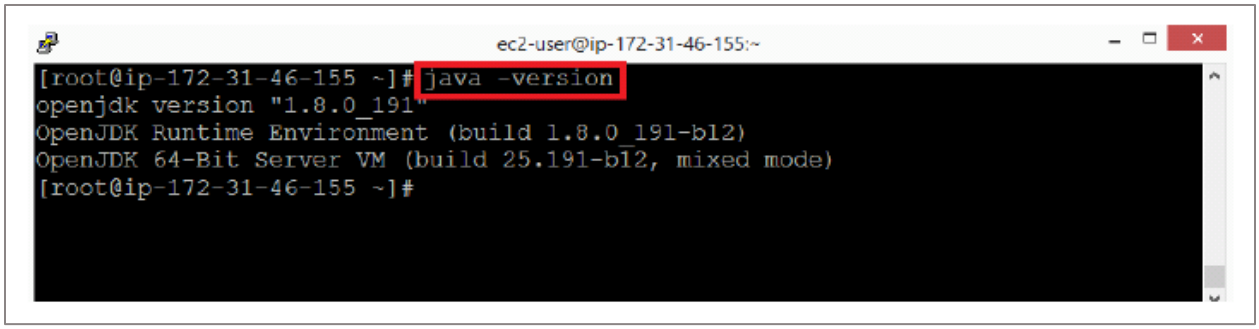
To check for updates, enter **sudo yum update -y**



Install Java by using this command: **sudo yum install java -y**



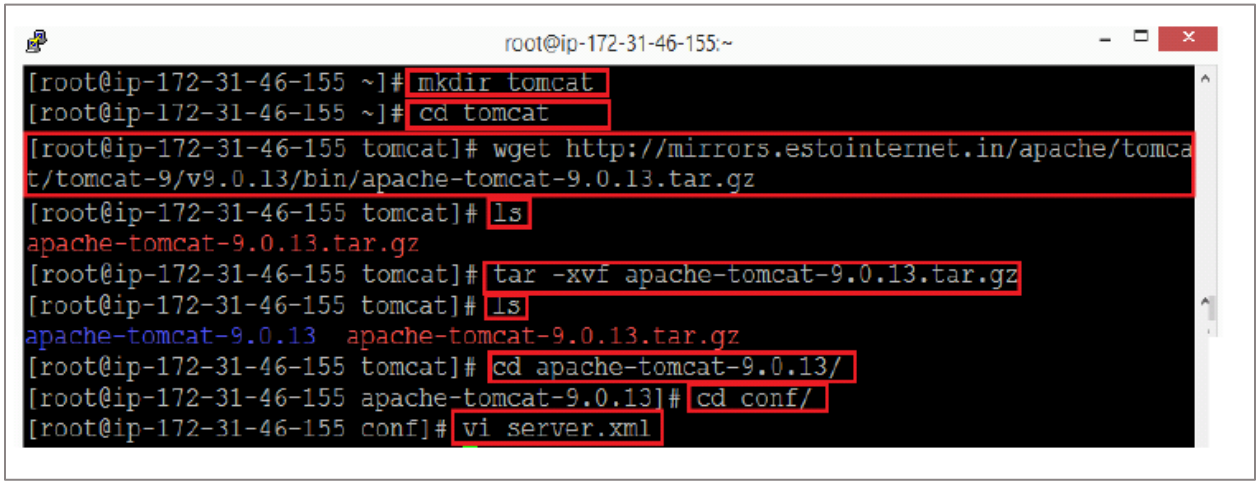
Check the version of Java by giving this command **java –version** as shown below



Step #5 | Tomcat Installation

To place Tomcat on your instance execute the following commands,

- Create folder : **mkdir folderName**(give any folderName name you wish to have)
  Example: mkdir tomcat
- Set the path for your folder : **cd folderName**
- Download the Tomcat file from official tomcat website,
  Wget http://mirrors.estointernet.in/apache/tomcat/tomcat-9/v9.0.13/bin/apache-tomcat-9.0.13.tar.gz
- Check whether tomcat tar file in that path by giving this command : **ls**
- Unzip the tar file : **tar -xvf apache-tomcat-9.0.13.tar.gz**
- Check whether unzipped tomcat file got placed in that path by giving this command : **ls**
- Goto that unzipped folder : **cd apache-tomcat-9.0.13**
- Goto conf folder : **cd conf**
- Open server.xml to change the port number of tomcat: **vi server.xml**



After opening **server.xml**, click on **insert** option on your keyboard and search for <connector port> by pressing the down arrow. Change the port to **80** and save the file by clicking the esc key on your keyboard and enter **: wq**

Note - Here, you can give any port number for Tomcat server. To access easily we gave it as 80

```
root@ip-172-31-46-155:~/tomcat/apache-tomcat-9.0.13/conf
    <!-- A "Connector" represents an endpoint by which requests are received
         and responses are returned. Documentation at :
         Java HTTP Connector: /docs/config/http.html
         Java AJP  Connector: /docs/config/ajp.html
         APR (HTTP/AJP) Connector: /docs/apr.html
         Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
    -->
    <Connector port="80" protocol="HTTP/1.1"
               connectionTimeout="20000"
               redirectPort="8443" />
    <!-- A "Connector" using the shared thread pool-->
    <!--
    <Connector executor="tomcatThreadPool"
               port="8080" protocol="HTTP/1.1"
               connectionTimeout="20000"
               redirectPort="8443" />
    -->
-- INSERT --
```

Open tomcat-users.xml : **vi tomcat-users.xml**

```
root@ip-172-31-46-155:~/tomcat/apache-tomcat-9.0.13/conf
[root@ip-172-31-46-155 conf]# vi tomcat-users.xml
```

D:\Praveen Kommula\Docs\Formatted\DigitalSummit\DS'18\Images\39.

Make sure that the below lines are added before **</tomcat-users>**

<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<user username="tomcat" password="tomcat" roles="manager-gui,manager-script"/>

Save the file by hitting ESC key and enter **: wq**

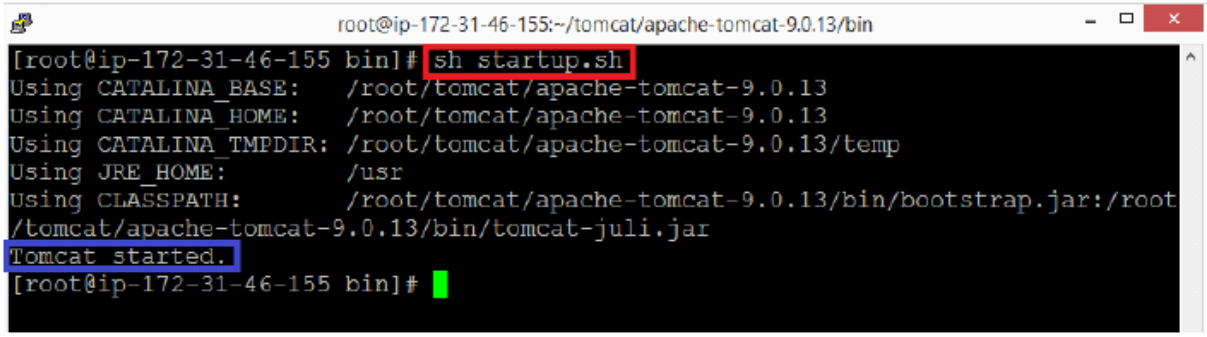
```
root@ip-172-31-46-155:~/tomcat/apache-tomcat-9.0.13/conf
<!--
  NOTE:  The sample user and role entries below are intended for use with the
  examples web application. They are wrapped in a comment and thus are ignored
  when reading this file. If you wish to configure these users for use with the
  examples web application, do not forget to remove the <!.. ..> that surrounds
  them. You will also need to set the passwords to something appropriate.
-->

  <role rolename="manager-gui"/>
  <role rolename="manager-script"/>
  <user username="tomcat" password="tomcat" roles="manager-gui,manager-script"/>


</tomcat-users>
-- INSERT --
```

Restart the Tomcat Server by giving restart command or by shut down and start again. To do that execute the below commands,

- Go back from the current folder : **cd ..**
- Go to bin folder : **cd bin**
- Shutdown Tomcat server : **sh shutdown.sh**
- Start Tomcat server : **sh startup.sh**

```
root@ip-172-31-46-155:~/tomcat/apache-tomcat-9.0.13/bin
[root@ip-172-31-46-155 conf]# cd ..
[root@ip-172-31-46-155 apache-tomcat-9.0.13]# cd bin/
[root@ip-172-31-46-155 bin]# sh shutdown.sh
Using CATALINA_BASE:   /root/tomcat/apache-tomcat-9.0.13
Using CATALINA_HOME:   /root/tomcat/apache-tomcat-9.0.13
Using CATALINA_TMPDIR: /root/tomcat/apache-tomcat-9.0.13/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /root/tomcat/apache-tomcat-9.0.13/bin/bootstrap.jar:/root
/tomcat/apache-tomcat-9.0.13/bin/tomcat-juli.jar
[root@ip-172-31-46-155 bin]#
```

You can see **Tomcat started** message on the screen.

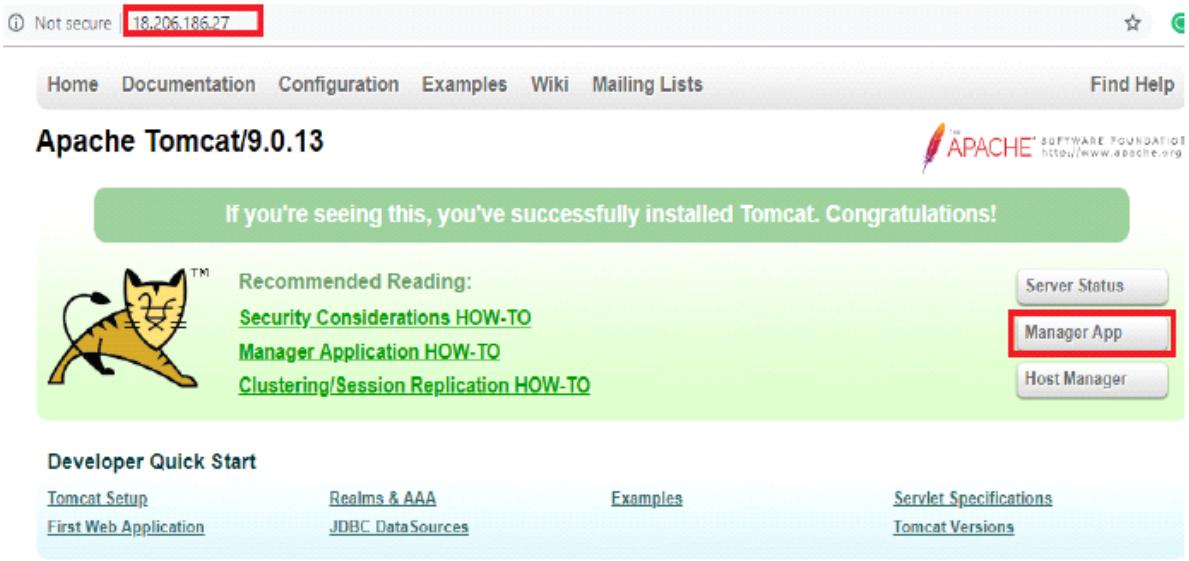


To access your application through Tomcat do the following process.

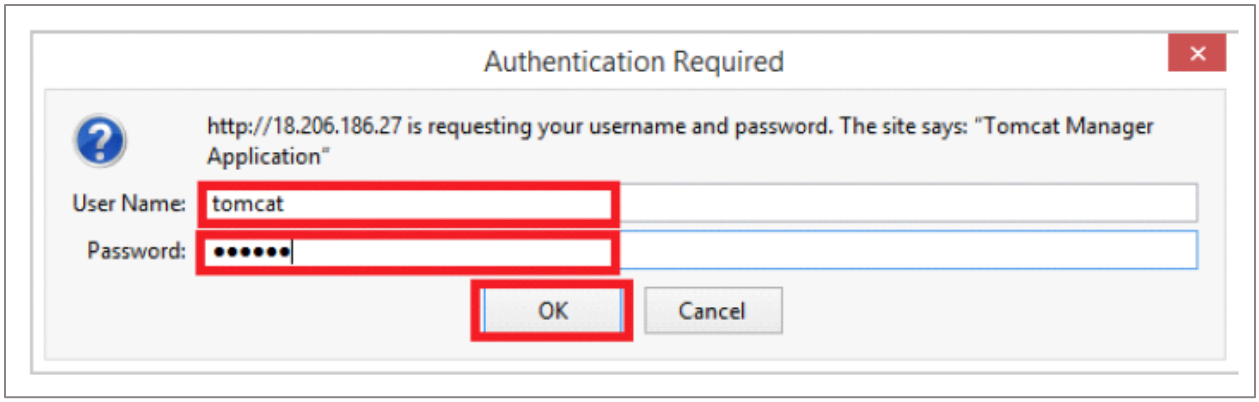
Step #6 | Login to Tomcat Manager Console

Enter your instance public IPV4 address in browser and hit enter. For example: 18.206.186.27

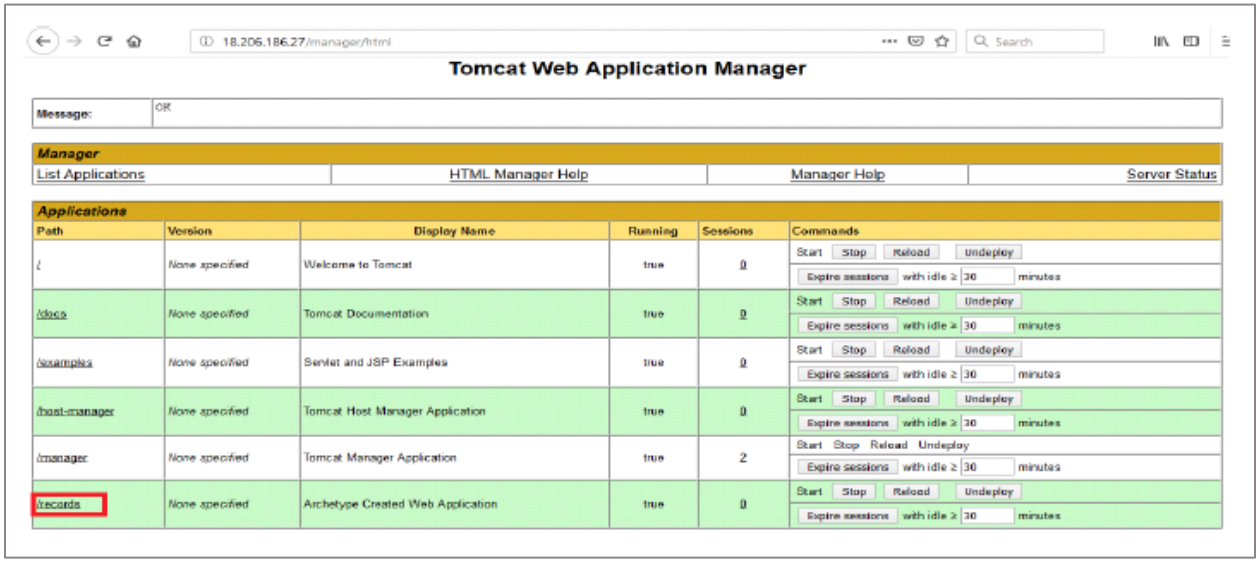
Click on **Manager App** and it asks for Username and Password.



Enter User Name and Password and click on **OK.**



After opening **ManagerApp** page, you will get the list of all deployed contexts as shown below.



If you get the 403 access denied error after opening the Tomcat Manager App do the following steps,

- Give the command to go the previous directory **cd ..**
- Goto the directory by using the command **cd webapps/manager/META-INF**
- List the files in that directory using l**s**
- Open the file **context.xml** by using this command **vi context.xml**
- Click the **insert** option on your keyboard
- Change the allow value as **\d+\.\d+\.\d+\.\d+** within the **double quotes ("")**. Save the file by selecting the **esc** key on your keyboard and enter : **wq**

- Come back to your Apache folder by giving the command **cd ..** (until you get the Apache folder)
- Open bin folder by using **cd bin**
- Once shut down your Tomcat Server with **sh shutdown.sh**
- Start Tomcat by using command **sh startup.sh**
- After completion of above procedure you have to open the manager app and you can deploy the application

For any questions regarding the lab please feel free to reach out to **innovation@miraclesoft.com**. **We hope you enjoyed this!**