

Ex no : 1	INSTALL VIRTUAL BOX/VM WARE
Date :	

AIM:

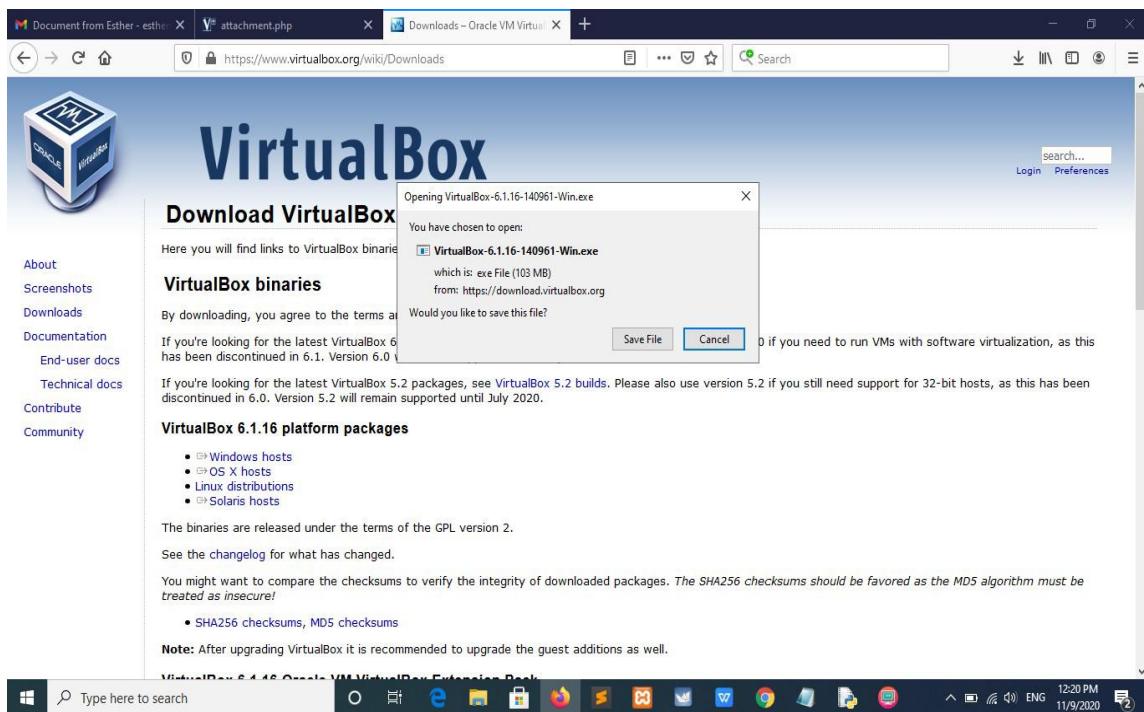
To install Virtual box/VM ware Workstation with different flavours of linux or windows OS on top of windows7 or 8.

REQUIREMENTS:

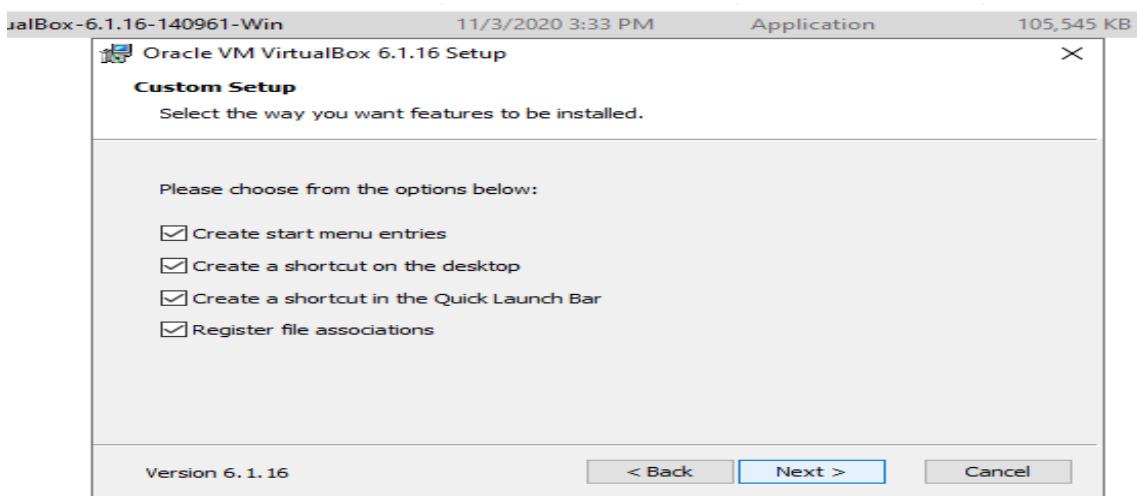
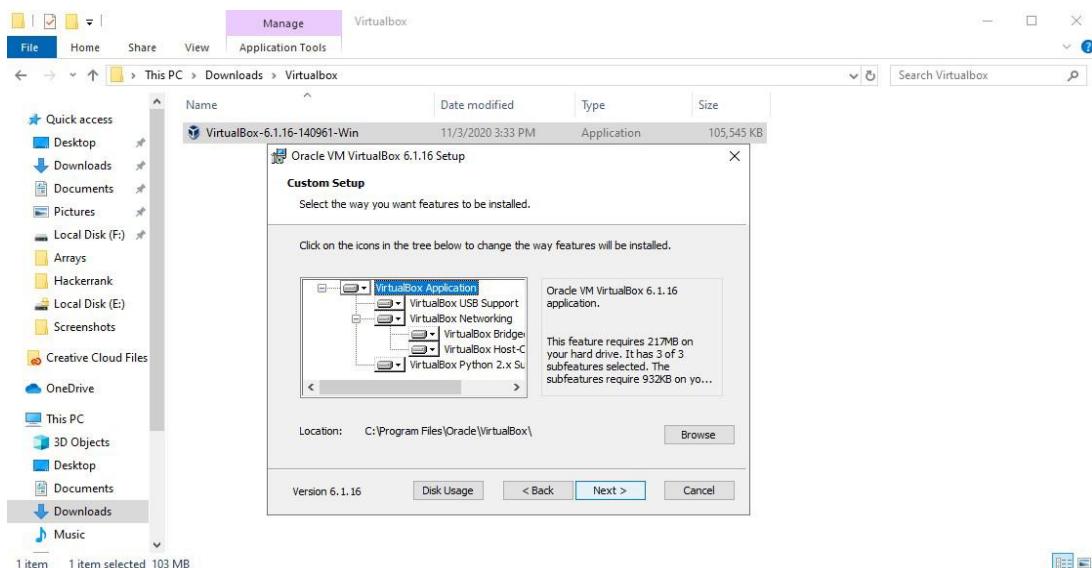
- Windows 7 or 8 operatingsystem.
- Virtualbox.
- Enough memory(RAM)to run your OS.

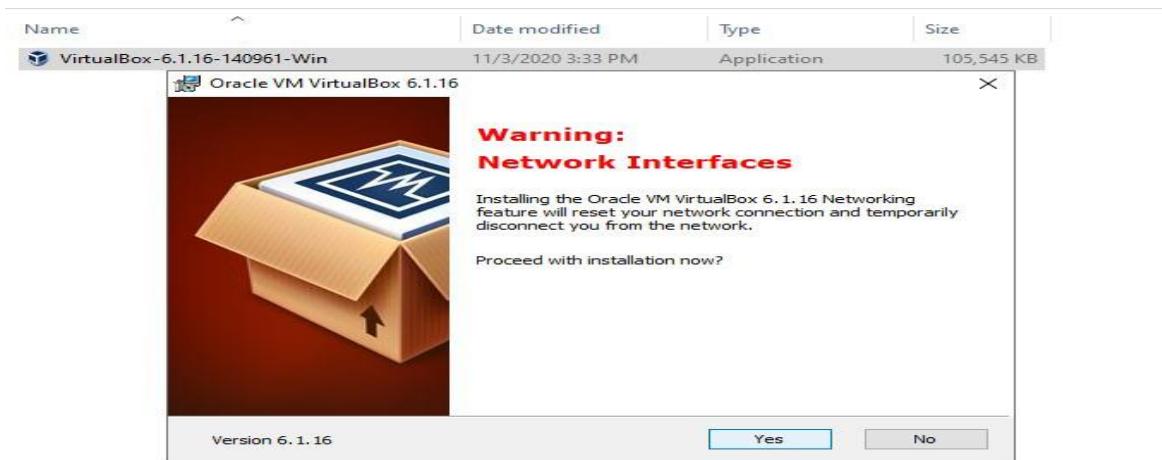
PROCEDURE:

- 1) Click <https://www.virtualbox.org/>
- 2) In Oracle VM,Click downloads and select the windows host then download with extension.

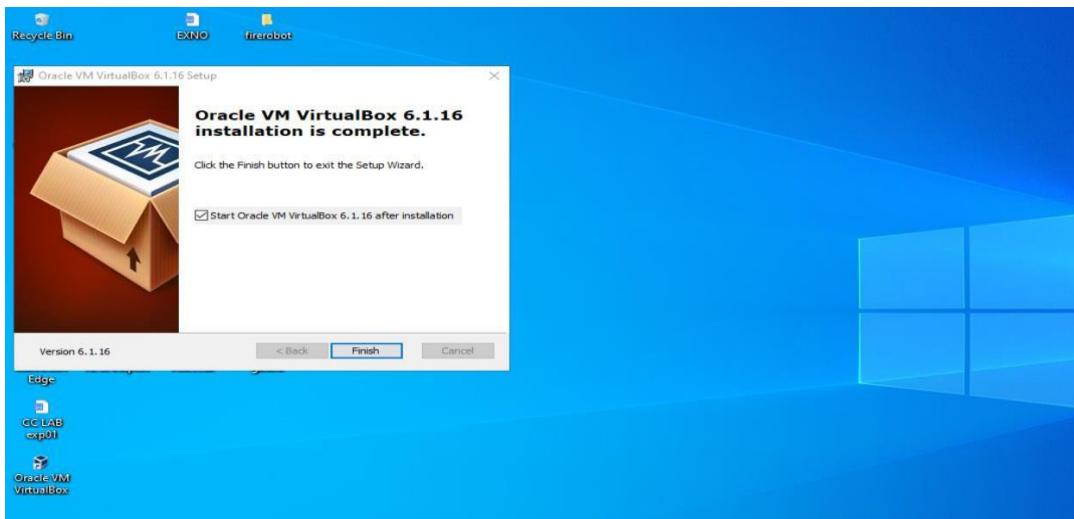
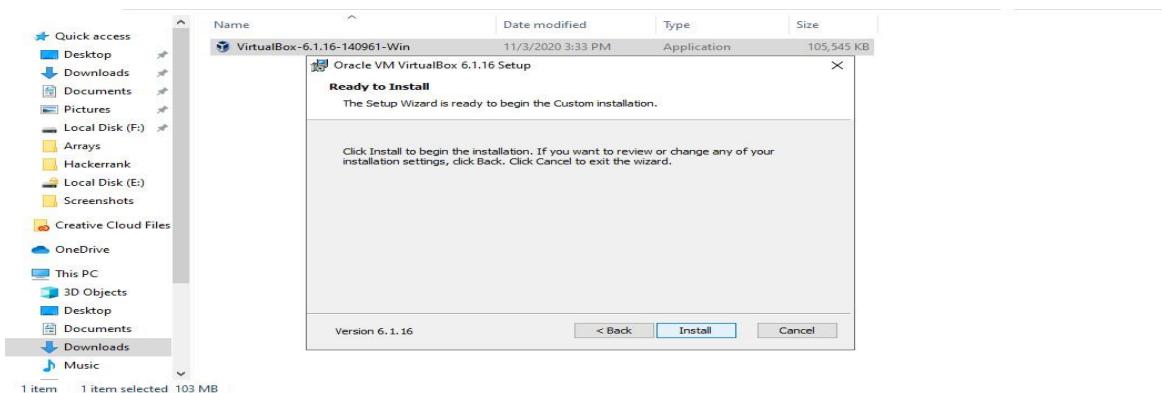


- 3) Open the extension file then click Next->Next->Next->Next->Yes->Install->Finish to install your virtualbox.





VIB



//Download the .iso image for Ubuntu-16.04-desktop version.

ubuntu-16.04.2-desktop-amd64 19-04-2017 21:26 Disc Image File 15,17,760 KB

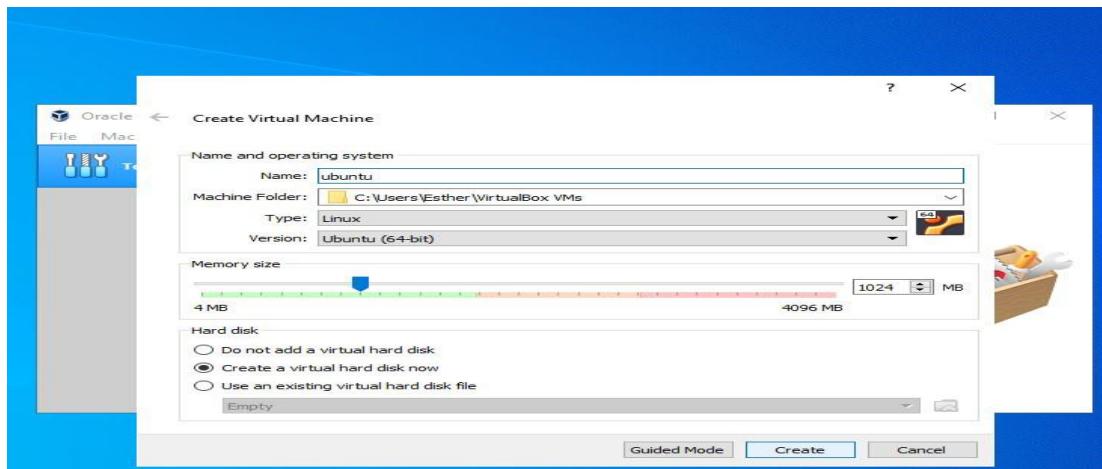
- 4) Open the Oracle Virtual Box and click new to create your virtualmachine.



- 5) Give your virtual machine name as ubuntu and select the type aslinux.

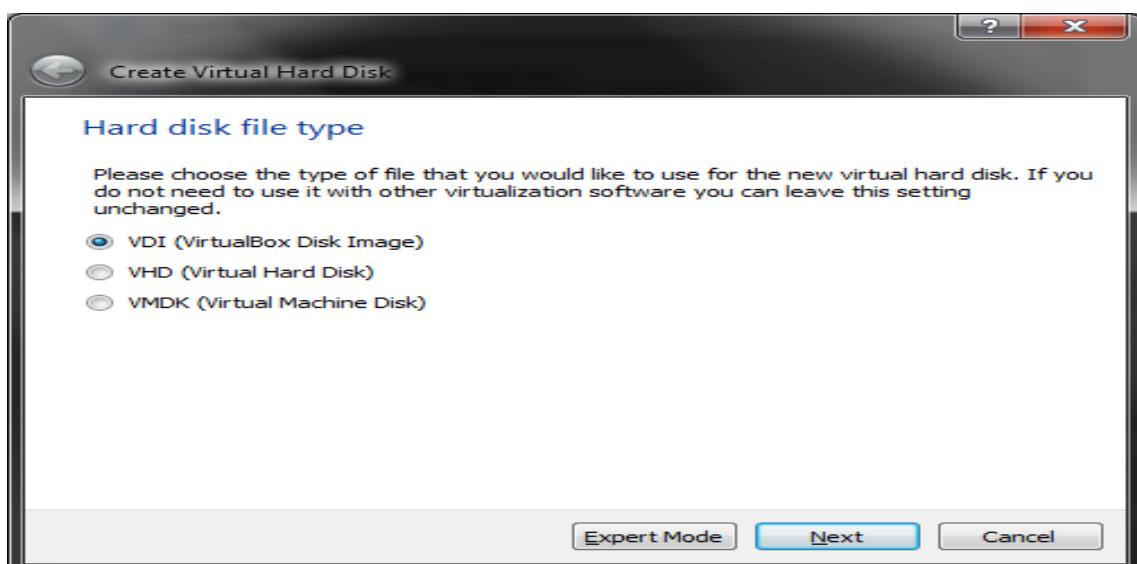
Then allocate **RAM** as per your usage(**1024 MB** is the recommended memory size).

Select “Create a virtual hard disk now”then click to create.

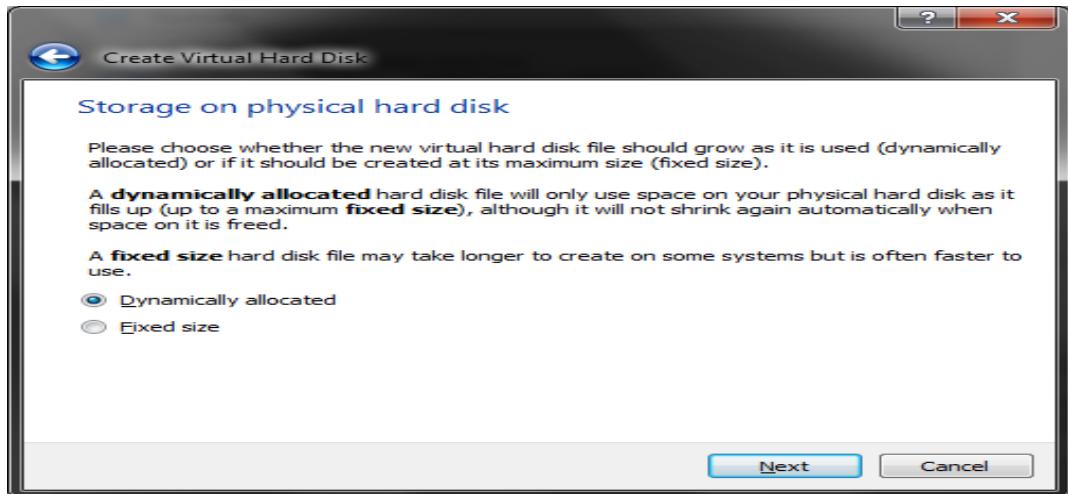


- 6) Select “**VDI(Virtual Box Disk Image)**” as the type for your Virtual Hard Disk file.

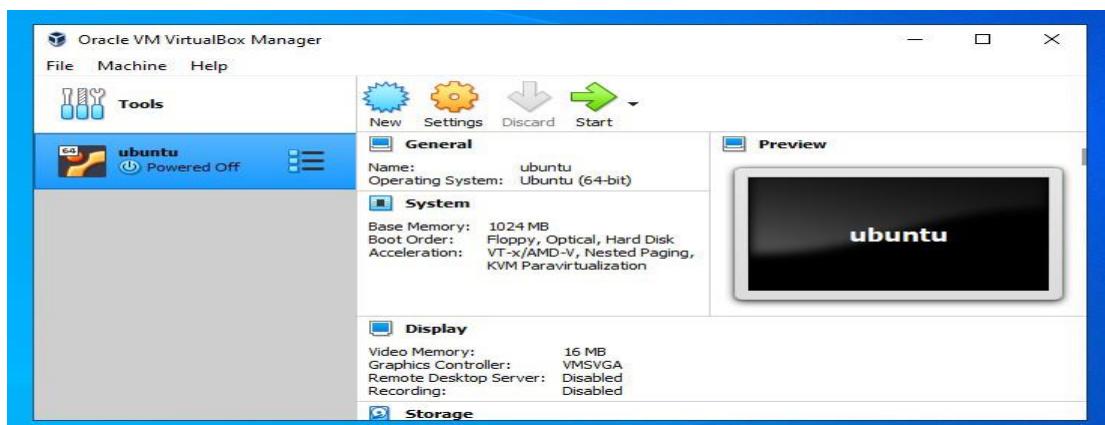
- 7)



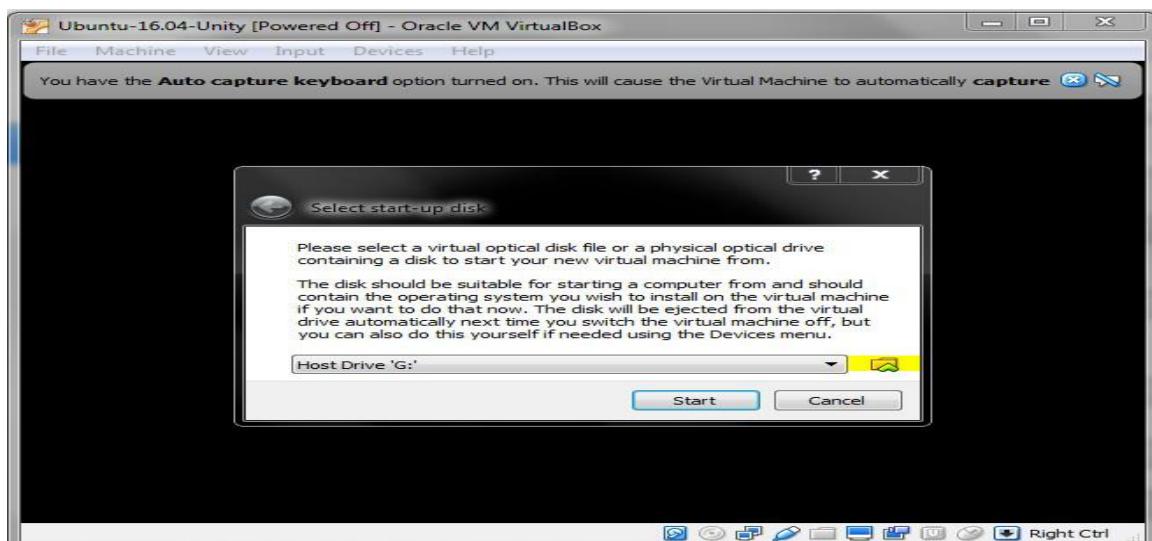
- 8) Select dynamically allocated as storage on physical harddisk.



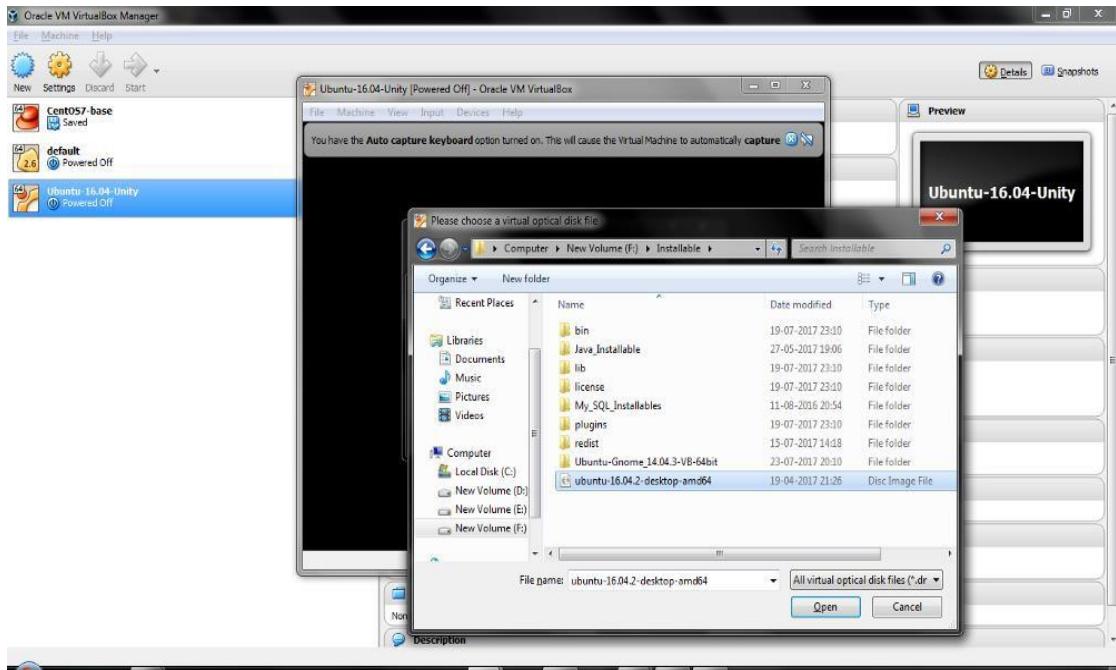
- 9) VM will now be created with the above configuration. Select the VM and click on "Start".



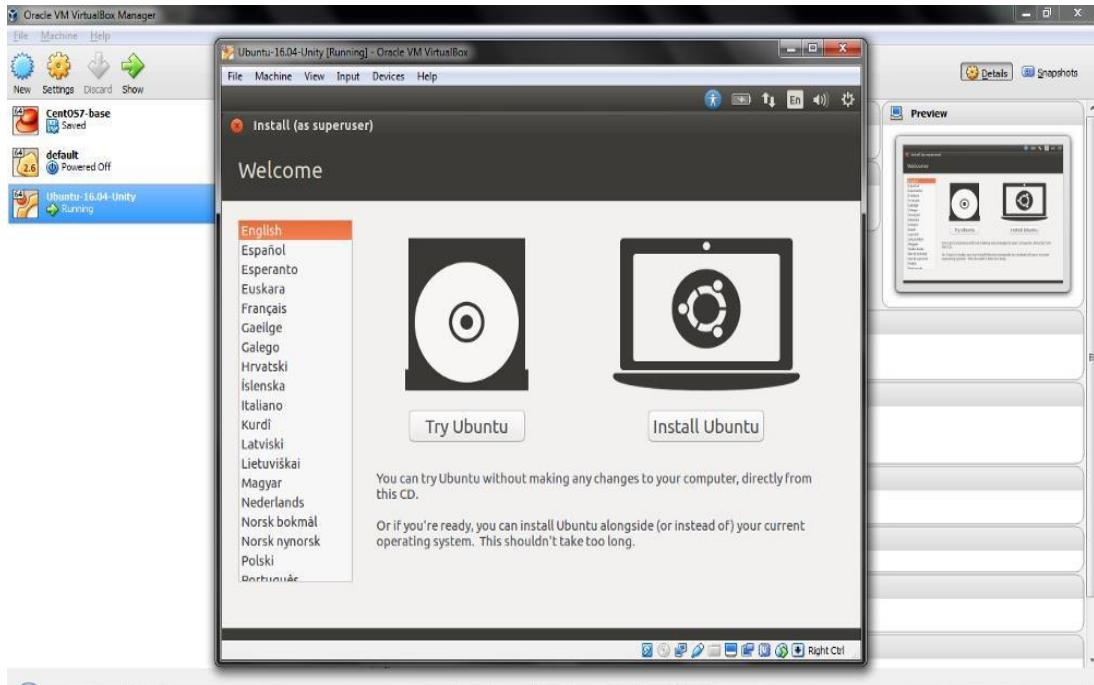
- 10) Now click the folder icon to select the virtual hard disk file then startit.



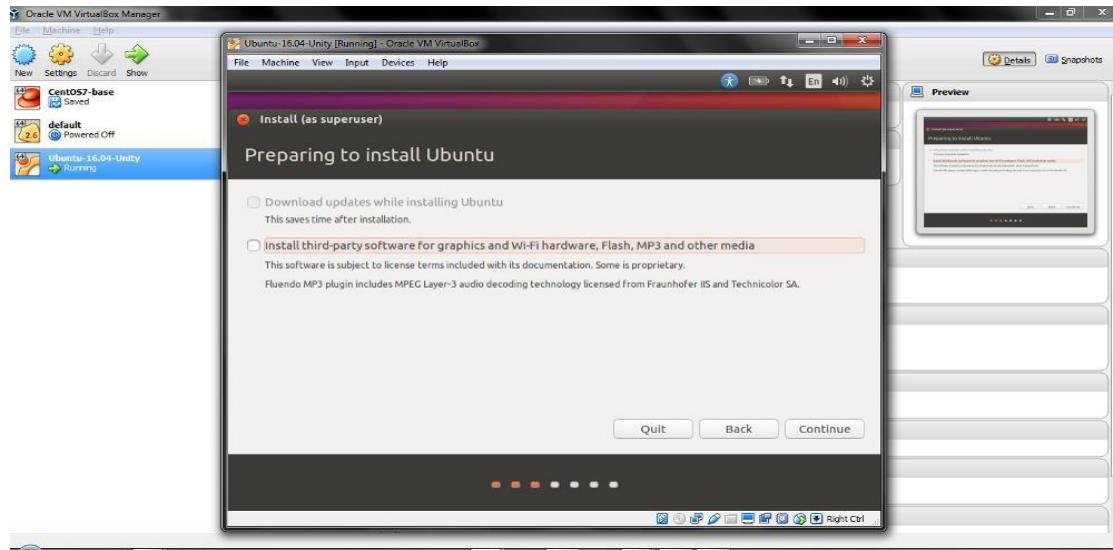
11)Go to select the path of your *Ubuntu-16.04-desktop-amd64.iso* image and Click“Open”.



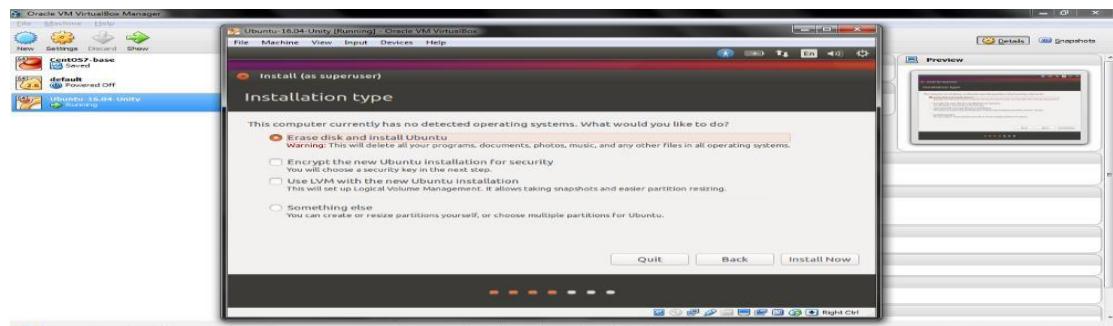
12)Now Ubuntu will boot up and ask for Installation details. Click on “InstallUbuntu”.



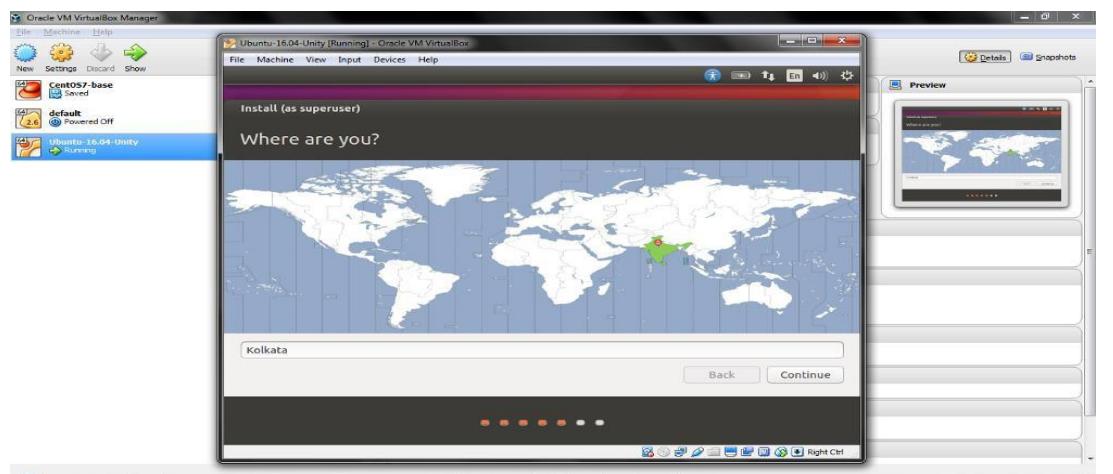
13) Don't select the checkbox as we don't want to install any 3rd party software while Ubuntu Installation.



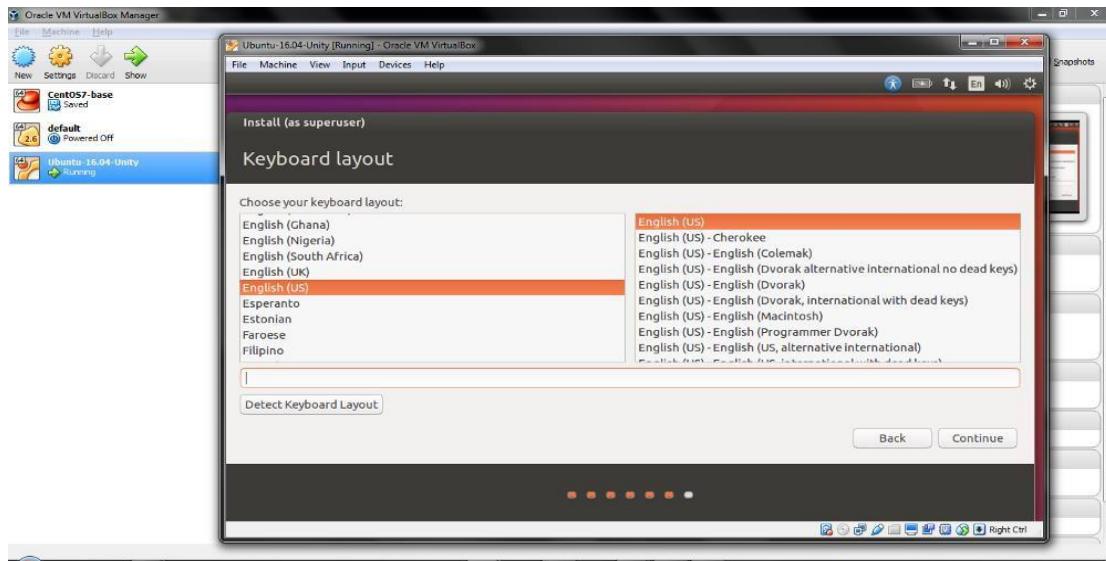
13) Select the “Erase disk and installUbuntu”.



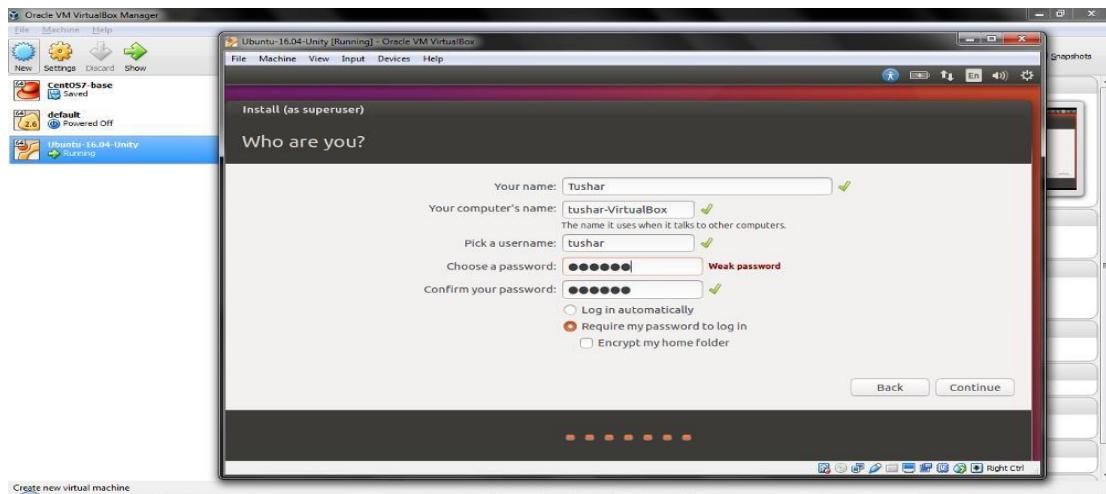
14) Select your location for time zonesettings(India)



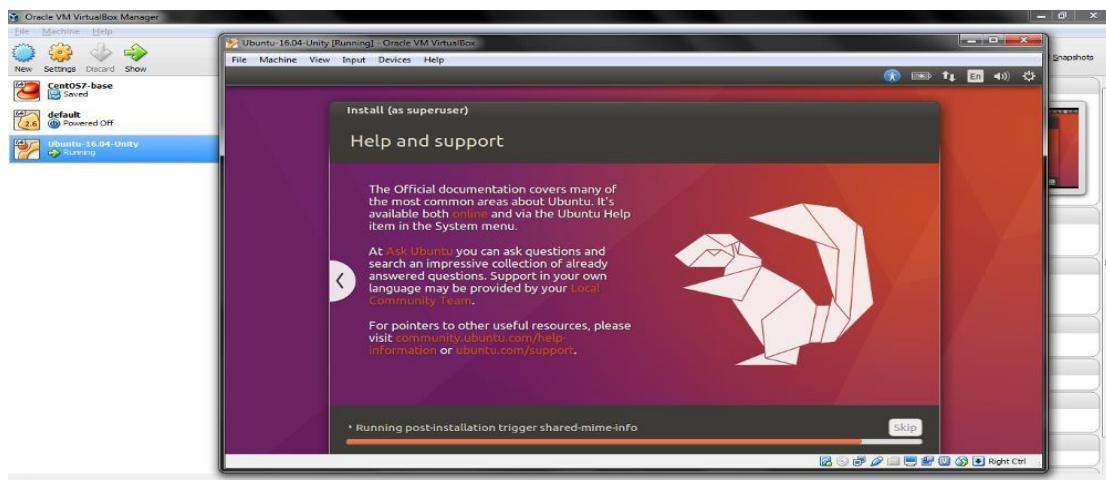
15) Select your keyboard layout as “English(US)”.



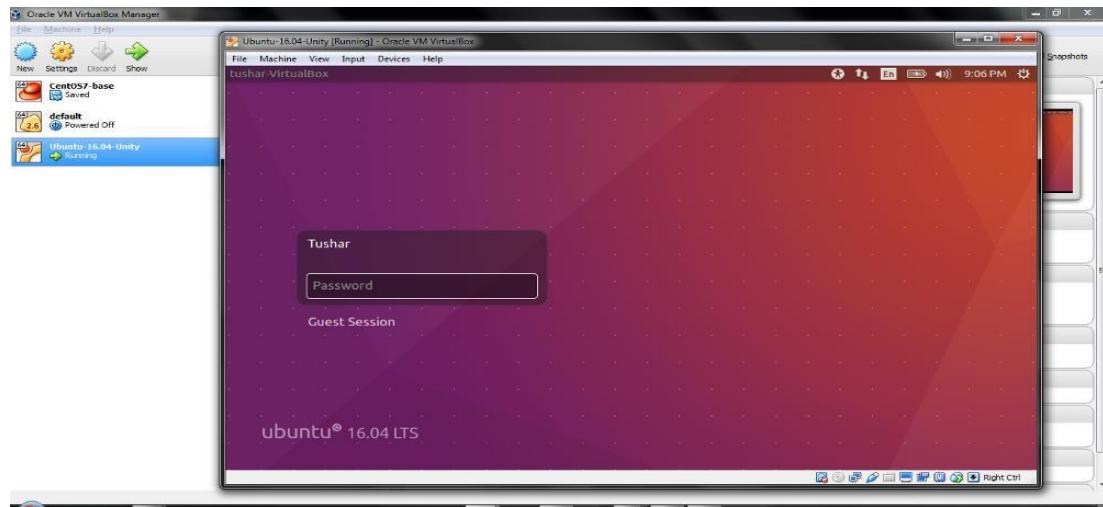
16) Fill in the below fields for Name, Username, Password and Computer's Name.



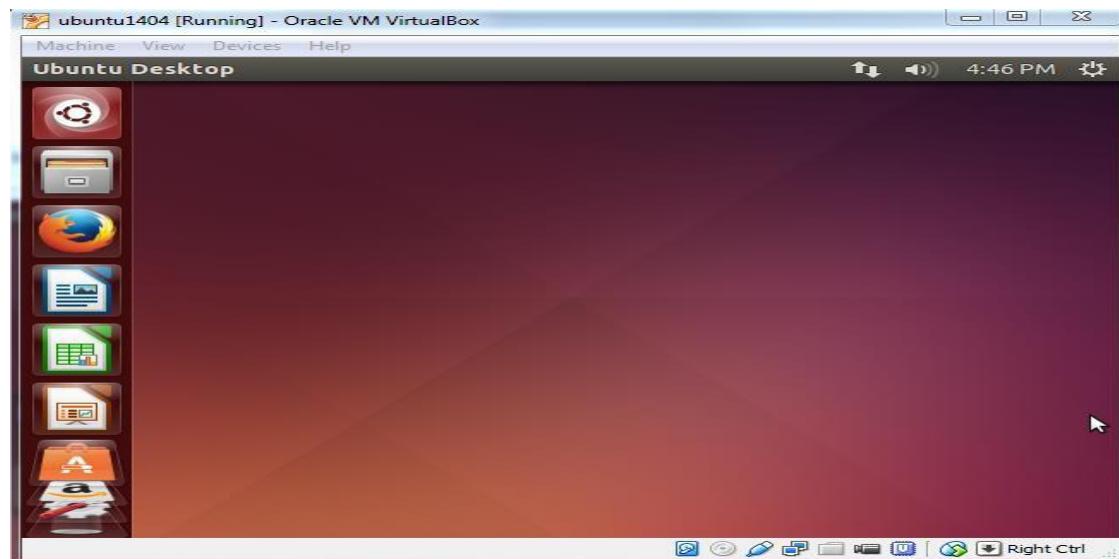
17) Wait for the Installation to complete.



18) Enter your **password** for logging into the Ubuntu-16.04 VMDesktop.



19) Now Virtual machine Ubuntu 16.04 LTS is running on your Oracle VirtualBox.



RESULT:

Thus the creation of virtual machine has been successfully installed.

Ex No : 2	Install a C compiler in the virtual
Date :	

AIM:

programs To Install a C compiler in the virtual machine created using virtual box and execute simple

REQUIREMENTS:

- Windows 7/8/10 Operating System.
- VirtualBox.
- Enough Memory (RAM) to run yourOS

PROCEDURE:

1. Open Terminal (Applications-Accessories-Terminal)
2. Type the command **Sudo apt-get install** to install packages
3. Open vi editor by using command **vi filename.c**
4. Type the program

```
#include<stdio.h>

main() {

printf("Hello World\n");

}
```
5. Save this file as “helloworld.c”
6. Type “ls” on Terminal to see all files under current folder
7. Confirm that “helloworld.c” is in the current directory. If not, type cd DIRECTORY_PATH to go to the directory that has “helloworld.c”
8. Type “gcc helloworld.c” to compile, and type “ls” to confirm that a new executable file “a.out” is created
9. Type “./a.out” on Terminal to run the program

```
yoonji@yoonji-VirtualBox:~$ ls  
Desktop Downloads helloworld.c Pictures Templates  
Documents examples.desktop Music Public Videos  
yoonji@yoonji-VirtualBox:~$ gcc helloworld.c  
yoonji@yoonji-VirtualBox:~$ ls  
a.out Documents examples.desktop Music Public Videos  
Desktop Downloads helloworld.c Pictures Templates  
yoonji@yoonji-VirtualBox:~$
```

10. If you see “Hello World” on the nextline, you just successfully ran your first Cprogram!

```
yoonji@yoonji-VirtualBox:~$ ls  
Desktop Downloads helloworld.c Pictures Templates  
Documents examples.desktop Music Public Videos  
yoonji@yoonji-VirtualBox:~$ gcc helloworld.c  
yoonji@yoonji-VirtualBox:~$ ls  
a.out Documents examples.desktop Music Public Videos  
Desktop Downloads helloworld.c Pictures Templates  
yoonji@yoonji-VirtualBox:~$ ./a.out  
Hello World  
yoonji@yoonji-VirtualBox:~$
```

RESULT:

Thus the creation of virtual machine has been successfully installed and executed simple programs.

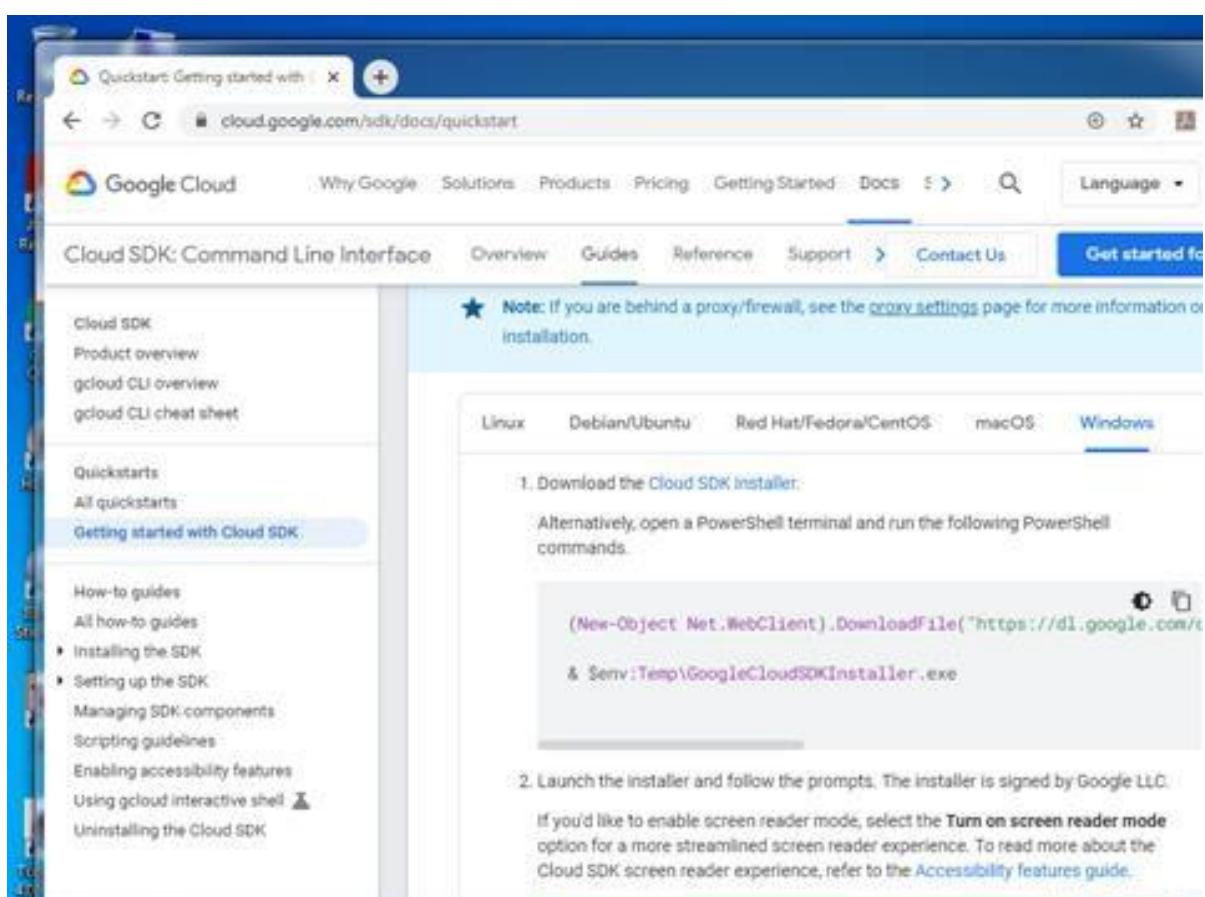
EX NO : 3	
DATE :	INSTALL GOOGLE APP ENGINE

AIM:

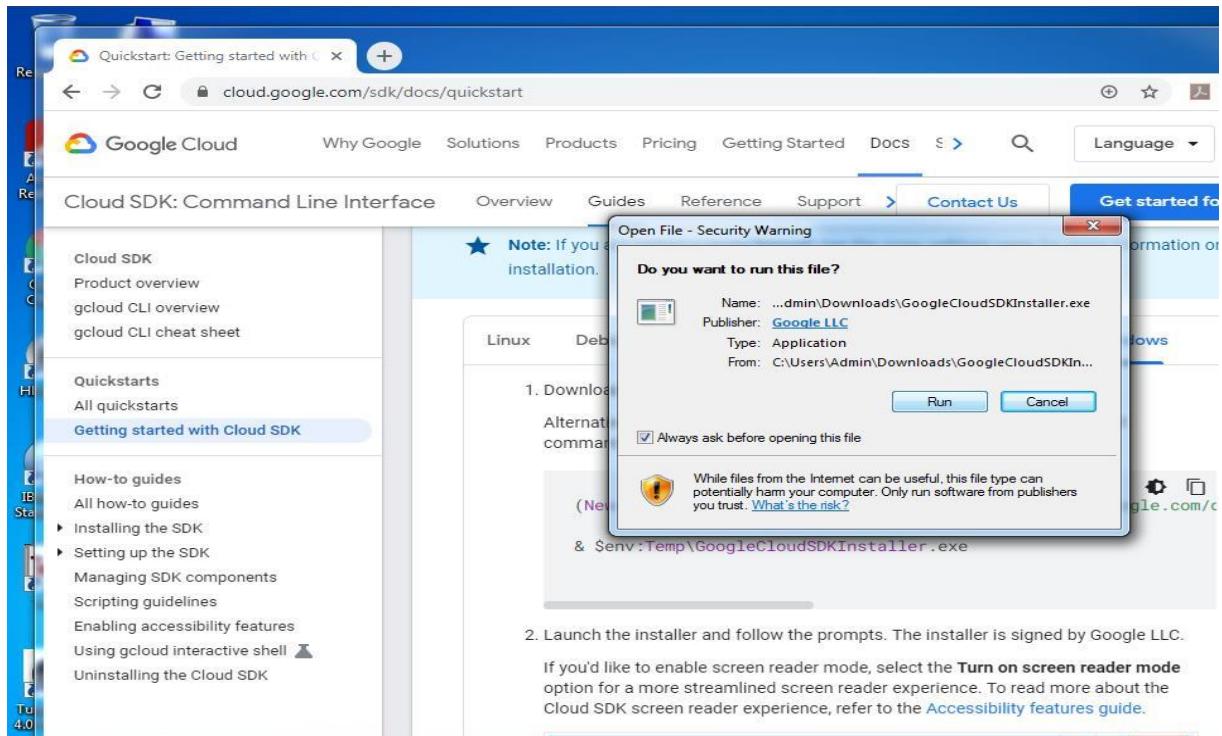
To install Google App Engine. Create hello world app and other simple web applications using python/java.

PROCEDURE:

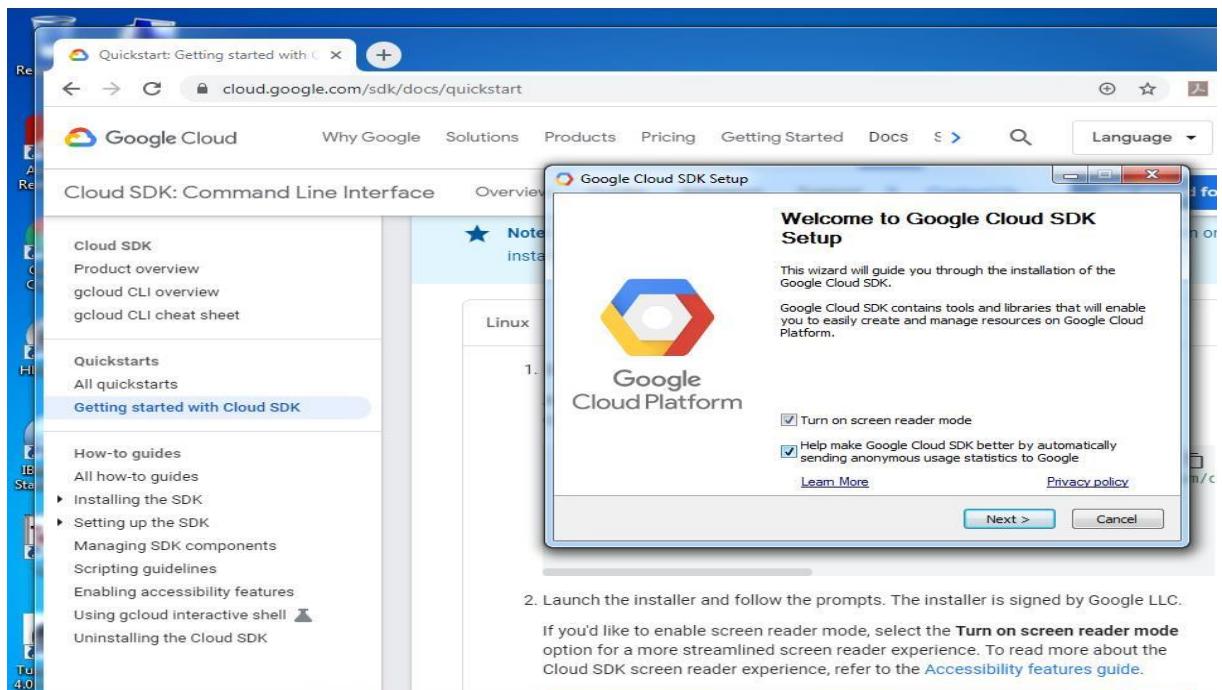
Step1: In Google, type cloud.google.com/sdk/docs/quickstart and download cloud SDK installer.



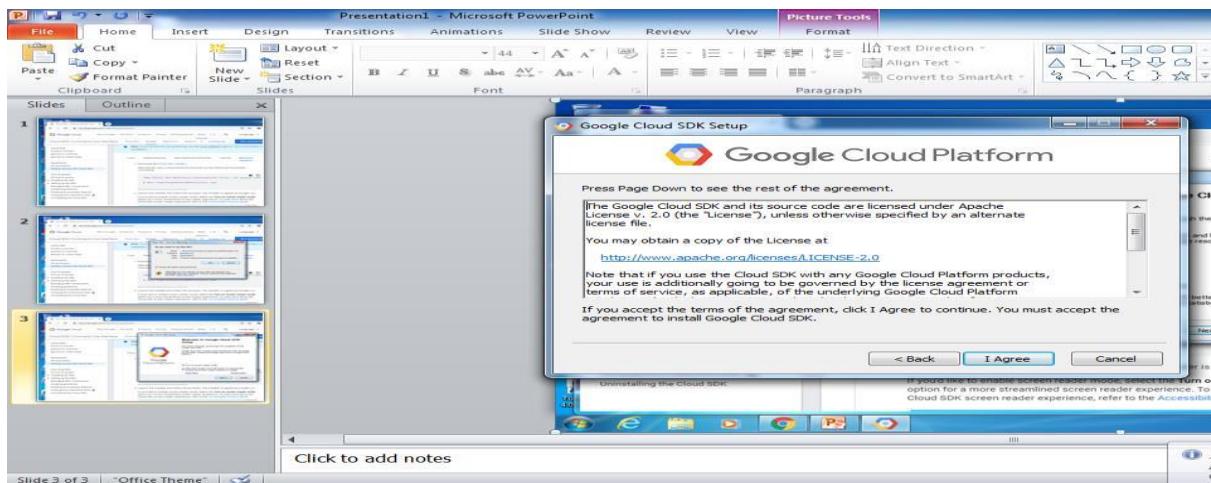
Step2: After installation, run the downloaded file.



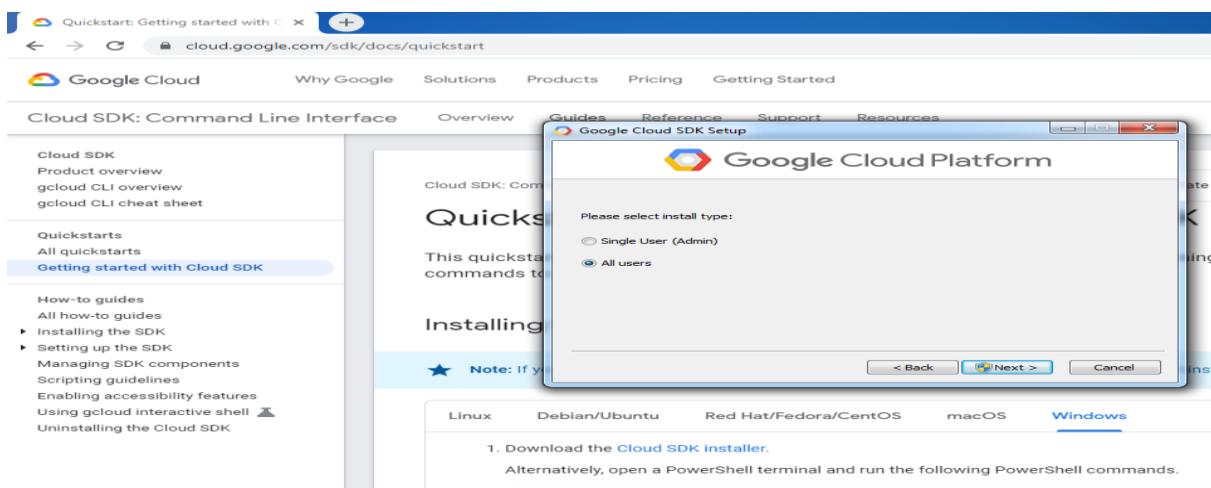
Step 3: Click Next.



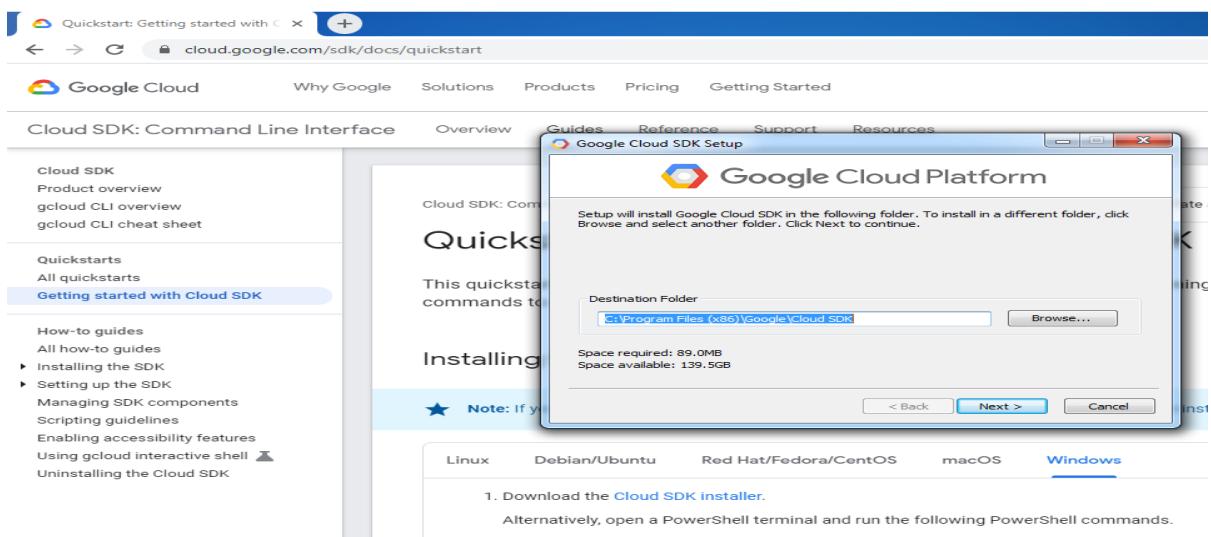
Step 4: Click I Agree button.



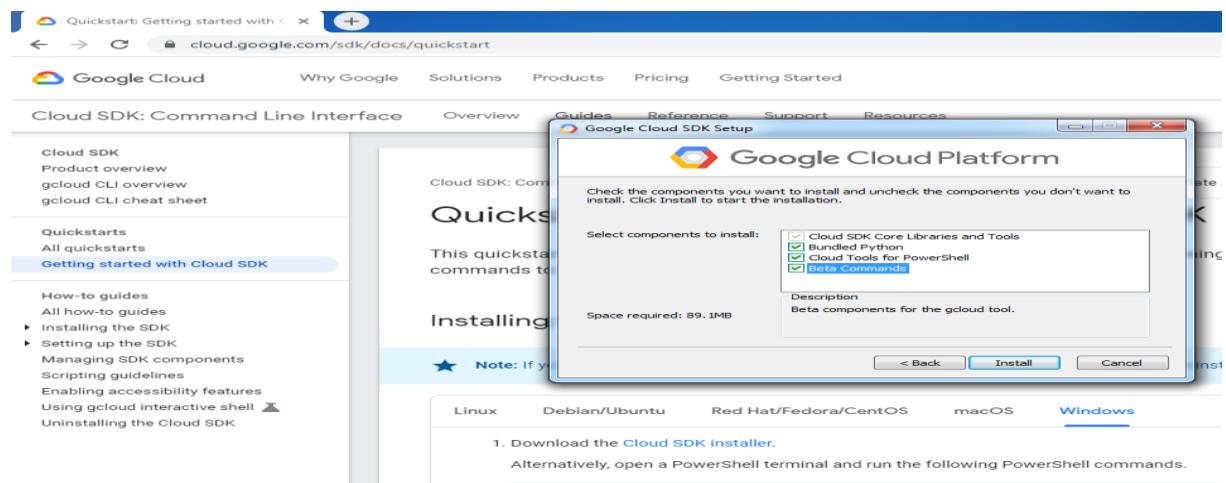
Step 5: select all user and click Next button.



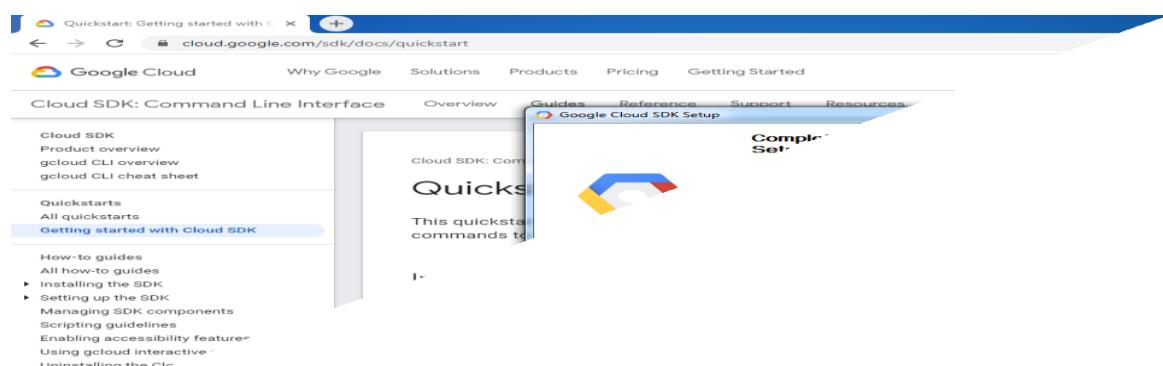
Step 6: Click Next.



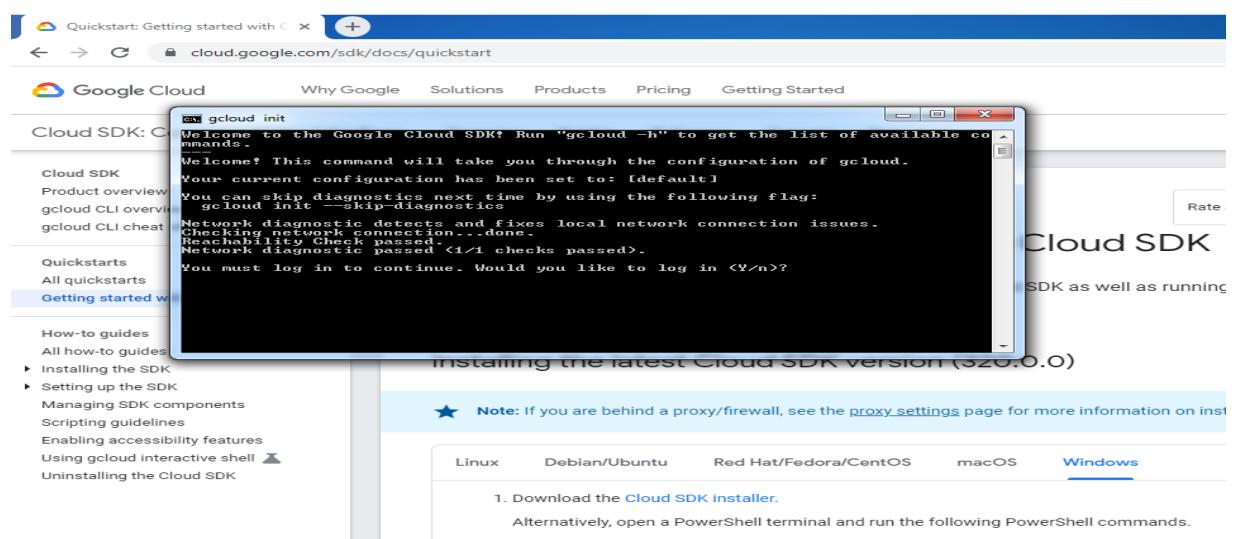
Step 7: Select the checkbox beta command and install.



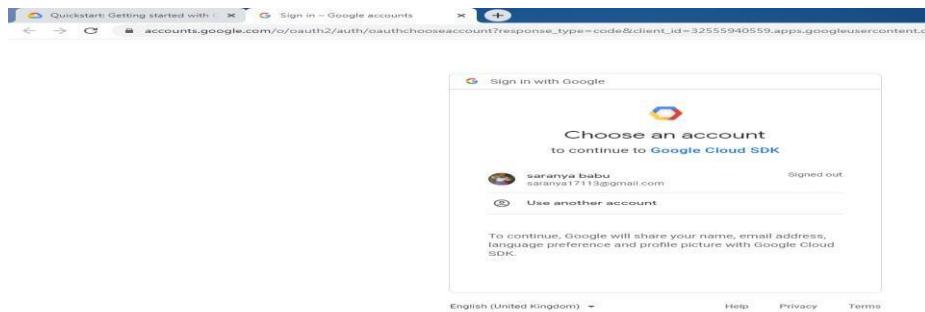
Step 8: Click Finish.



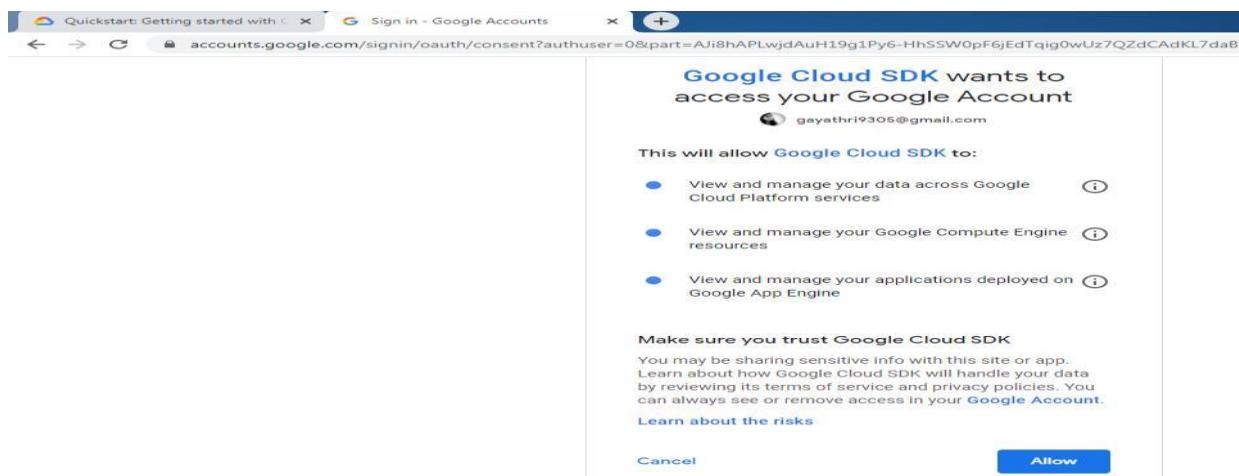
Step 9: Open command prompt.



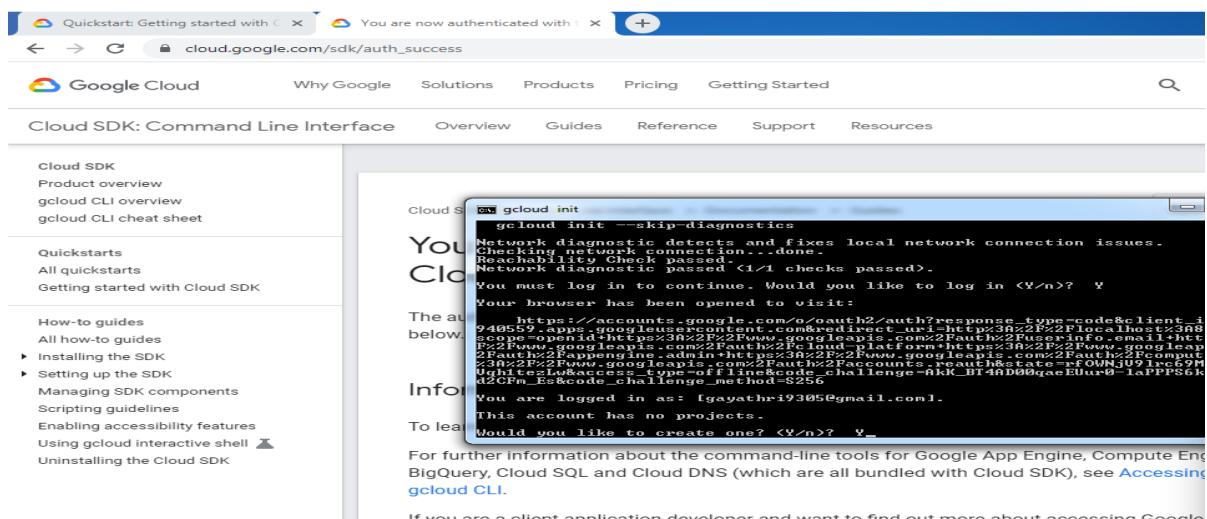
Step 10: Sign into the Gmail.



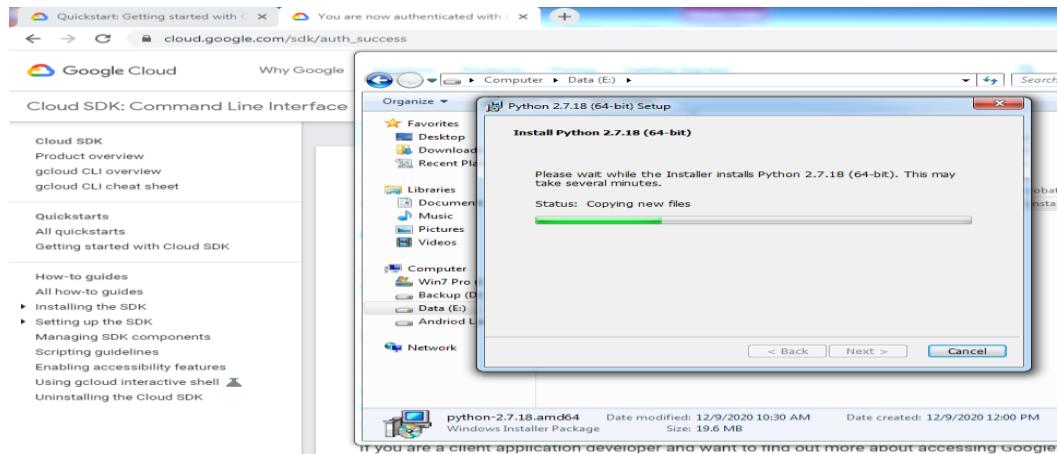
Step 11: Click Allow.



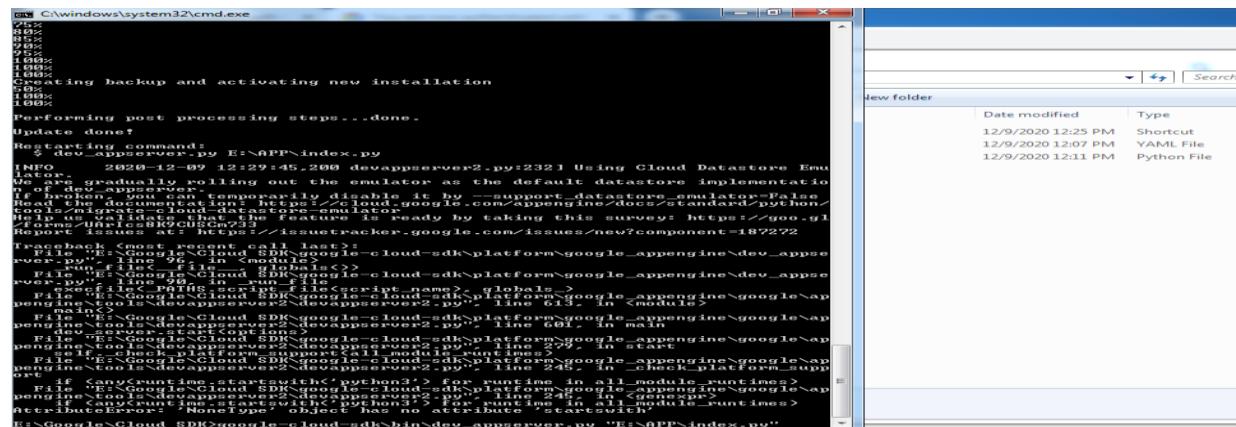
Step 12: Click Yes.



Step 13: Install Python.



Step 16: Select the location.



Step 17: IP address will obtain.



Step 18: Enter the localhost:8080/ in Google.



Step 19: Final output.



RESULT:

Thus the Google app engine has been installed and creation of hello world app, other simple web applications using python/java has been successfully verified.

EX NO : 4	
------------------	--

DATE :	Use GAE launcher to launch the web applications
---------------	--

AIM :

Pre--Requisites:Python 2.5.4

Download and Install Python 2.5.4 from:

<http://www.python.org/download/releases/2.5.4/>

Download and Install

You can download the Google App Engine SDK by going to:

<http://code.google.com/appengine/downloads.html>

and download the appropriate installpackage.

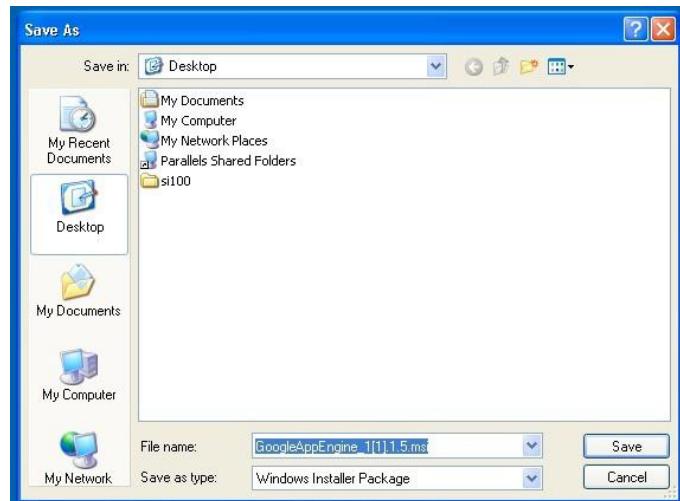
Download the Google App Engine SDK

Before downloading, please read the [Terms](#) that govern your use of the App Engine SDK.

Please note: The App Engine SDK is under **active development**, please keep this in mind as you explore its capabilities. See the [SDK Release Notes](#) for the information on the most recent changes to the App Engine SDK. If you discover any issues, please feel free to notify us via our [Issue Tracker](#).

Platform	Version	Package	Size	SHA1 Checksum
Windows	1.1.5 - 10/03/08	GoogleAppEngine_1.1.5.msi	2.5 MB	e974312b4aefc0b3873ff0d93eb4c525d5e88c30
Mac OS X	1.1.5 - 10/03/08	GoogleAppEngineLauncher-1.1.5.dmg	3.6 MB	f62208ac01c1b3e39796e58100d5f1b2f052d3e7
Linux/Other Platforms	1.1.5 - 10/03/08	google_appengine_1.1.5.zip	2.6 MB	cbb9ce817bdabf1c4f181d9544864e55ee253de1

Download the Windows installer – the simplest thing is to download it to your Desktop or another folder that you remember.



Double Click on the **GoogleApplicationEngine** installer.



Click through the installation wizard, and it should install the App Engine. If you do not have Python 2.5, it will install Python 2.5 as well.

Once the install is complete you can discard the downloaded installer



Making your First Application

Now you need to create a simple application. We could use the “+” option to have the launcher make us an application – but instead we will do it by hand to get a better sense of what is going on.

Make a folder for your Google App Engine applications. I am going to make the Folder on my Desktop called “**apps**” – the path to this folder is:

C:\Documents and Settings\csev\Desktop\apps

And then make a sub---folder in within **apps** called “**ae--01--trivial**” – the path to this folder would be:

C:\ Documents and Settings \csev\Desktop\apps\ae--01--trivial

Using a text editor such as JEdit (www.jedit.org), create a file called **app.yaml** in the **ae--01--trivial** folder with the following contents:

```
application: ae-01-trivial
version: 1
runtime:
python
api_version: 1

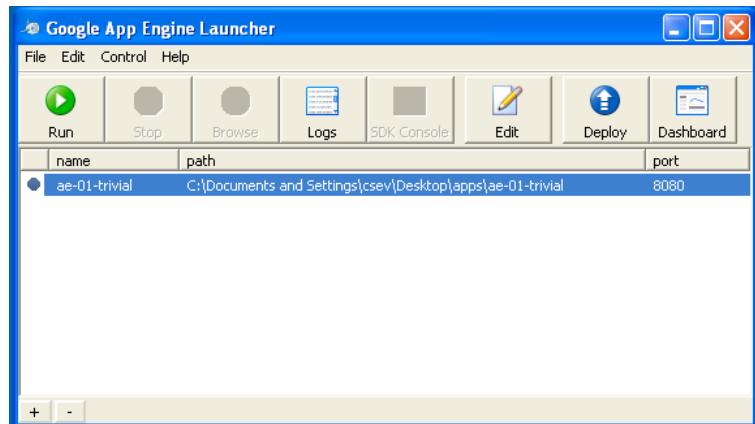
handlers:
- url: /.*
  script: index.py
```

Note: Please do not copy and paste these lines into your text editor – you might end up with strange characters – simply type them into your editor.

Then create a file in the **ae--01--trivial** folder called **index.py** with three lines in it:

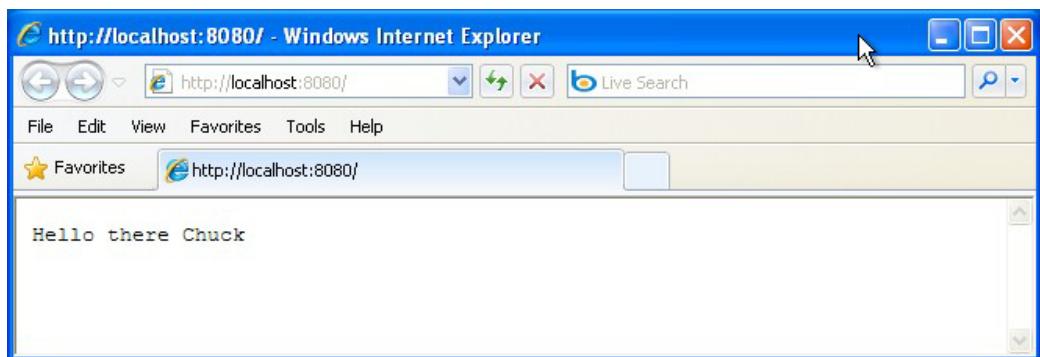
```
print 'Content-Type: text/plain'
print ''
print 'Hello there Chuck'
```

Then start the **GoogleAppEngineLauncher** program that can be found under **Applications**. Use the **File -->Add Existing Application** command and navigate into the **apps** directory and select the **ae--01--trivial** folder. Once you have added the application, select it so that you can control the application using the launcher.



Once you have selected your application and press **Run**. After a few moments your application will start and the launcher will show a little green icon next to your application. Then press **Browse** to open a browser pointing at your application which is running at <http://localhost:8080>/

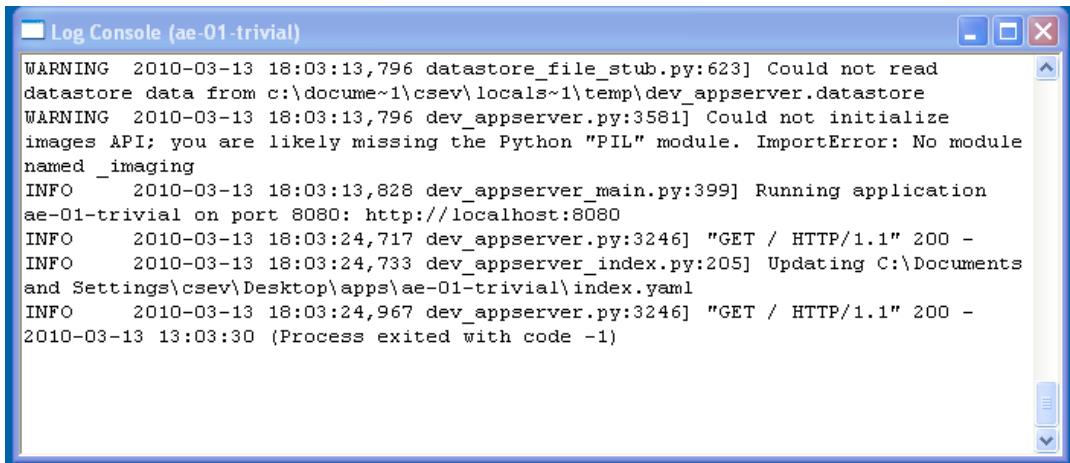
Paste <http://localhost:8080> into your browser and you should see your application as follows:



Just for fun, edit the **index.py** to change the name “Chuck” to your own name and press Refresh in the browser to verify your updates.

Watching the Log

You can watch the internal log of the actions that the web server is performing when you are interacting with your application in the browser. Select your application in the Launcher and press the **Logs** button to bring up a log window:



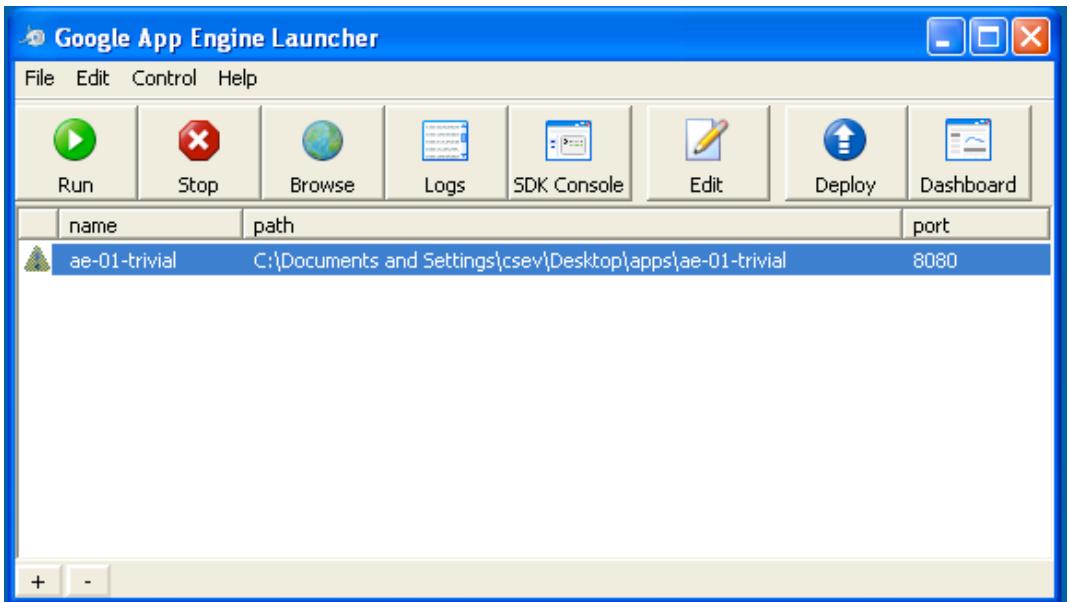
The screenshot shows a window titled "Log Console (ae-01-trivial)". The window contains a scrollable text area displaying application logs. The logs show several warning messages about missing modules and initialization issues, followed by informational messages about the application starting and receiving a GET request.

```
WARNING 2010-03-13 18:03:13,796 datastore_file_stub.py:623] Could not read
datastore data from c:\docume~1\csev\locals~1\temp\dev_appserver.datastore
WARNING 2010-03-13 18:03:13,796 dev_appserver.py:3581] Could not initialize
images API; you are likely missing the Python "PIL" module. ImportError: No module
named _imaging
INFO    2010-03-13 18:03:13,828 dev_appserver_main.py:399] Running application
ae-01-trivial on port 8080: http://localhost:8080
INFO    2010-03-13 18:03:24,717 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
INFO    2010-03-13 18:03:24,733 dev_appserver_index.py:205] Updating C:\Documents
and Settings\csev\Desktop\apps\ae-01-trivial\index.yaml
INFO    2010-03-13 18:03:24,967 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
2010-03-13 13:03:30 (Process exited with code -1)
```

Each time you press **Refresh** in your browser – you can see it retrieving the output with a **GET** request.

Dealing With Errors

With two files to edit, there are two general categories of errors that you may encounter. If you make a mistake on the **app.yaml** file, the App Engine will not start and your launcher will show a yellow icon near your application:



To get more detail on what is going wrong, take a look at the log for the application:

The screenshot shows a window titled "Log Console (ae-01-trivial)". The error message is:

```
invalid object:  
Unknown url handler type.  
<URLMap  
    static_dir=None  
    secure=default  
    script=None  
    url='.*'  
    static_files=None  
    upload=None  
    mime_type=None  
    login=optional  
    require_matching_file=None  
    auth_fail_action=redirect  
    expiration=None  
>  
in "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\app.yaml", line 8,  
column 1
```

In this instance – the mistake is mis---indenting the last line in the **app.yaml** (line 8).

If you make a syntax error in the **index.py** file, a Python trace back error will appear in your browser.

The screenshot shows a browser window for "http://localhost:8080" in Internet Explorer. The error message is:

```
Traceback (most recent call last):  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    self._Dispatch(dispatcher, self.rfile, outfile, env_dict)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    base_env_dict=env_dict)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    base_env_dict=base_env_dict)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    self._module_dict)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    reset_modules = exec_script(handler_path, cgi_path, hook)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    handler_path, cgi_path, import_hook)  
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserver:  
    module_code = compile(source_file.read(), cgi_path, 'exec')  
  File "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\index.py", line 3  
[  print 'Hello, World!  
   ^  
SyntaxError: EOL while scanning single-quoted string
```

The error you need to see is likely to be the last few lines of the output – in this case I made a Python syntax error on line one of our one---line application.

Reference: http://en.wikipedia.org/wiki/Stack_trace

When you make a mistake in the **app.yaml** file – you must fix the mistake and attempt to start the application again.

If you make a mistake in a file like **index.py**, you can simply fix the file and press refresh in your browser – there is no need to restart the server.

Shutting Down the Server

To shut down the server, use the Launcher, select your application and press the **Stop** button.

RESULT :

Thus the GAE Lanucher was successfully installed and simple application was developed successfully.

EX NO : 5	SIMULATE A CLOUD SCENARIO USING CLOUDSIM
DATE :	

AIM :

To simulate a cloud scenario using cloudsim and run a scheduling algorithms that is not present in cloudsim

PROCEDURE :

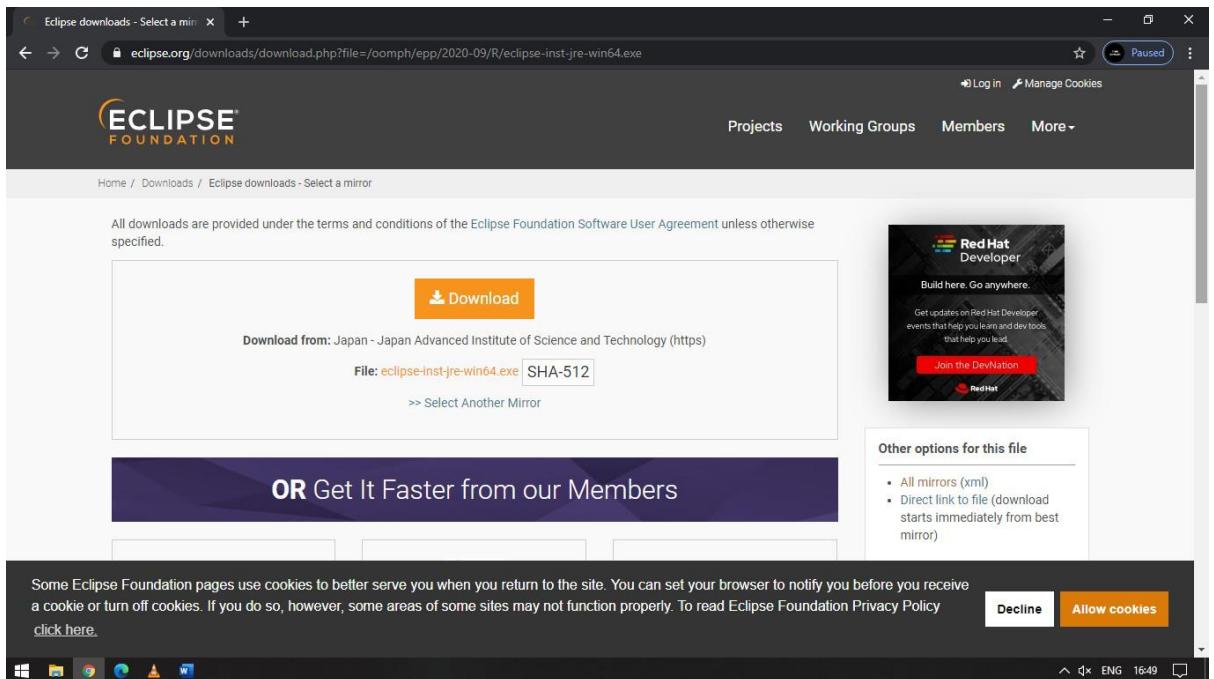
1. Download CloudSim installable files from <https://code.google.com/p/cloudsim/downloads/list> and unzip

The screenshot shows a web browser window with the following details:

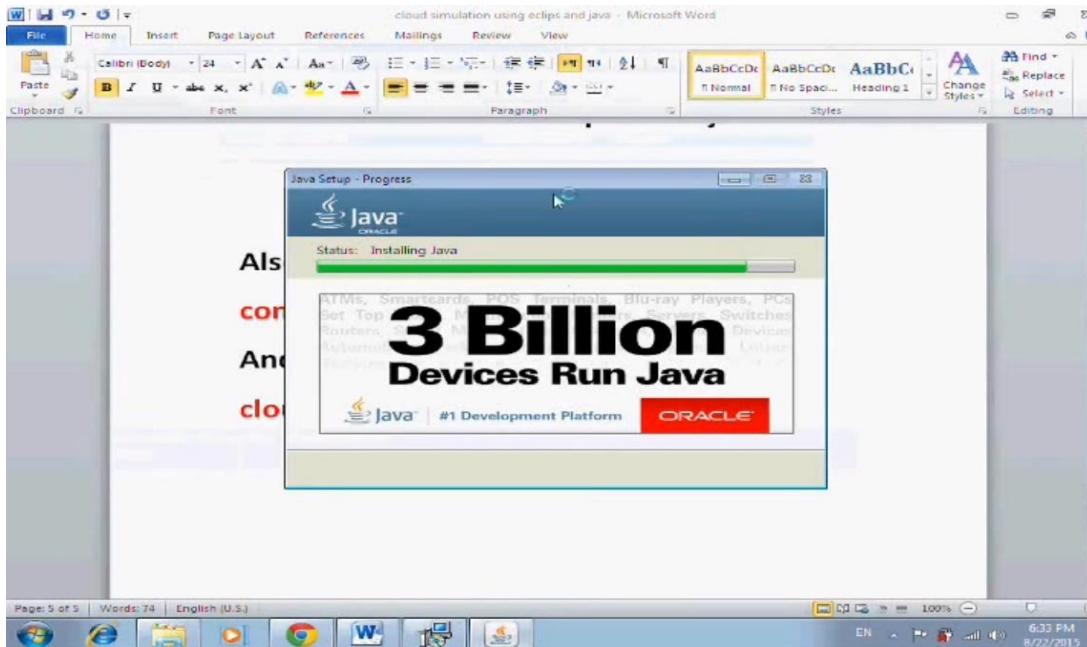
- Title Bar:** Google Code Archive - Long-term
- Address Bar:** code.google.com/archive/p/cloudsim/downloads
- Page Header:** Google Code Archive (Paused)
- Search Bar:** Search this site
- Project Navigation:** Projects, Search, About
- Project Name:** cloudsim
- File List:**

File	Summary + Labels	Uploaded	Size
cloudsim-3.0.3.tar.gz	CloudSim 3.0.3: bug fix release	May 2, 2013	9.9MB
cloudsim-3.0.3.zip	CloudSim 3.0.3: bug fix release	May 2, 2013	13.05MB
cloudsim-3.0.2.tar.gz	CloudSim 3.0.2: bug fix release	Nov 6, 2012	9.9MB
cloudsim-3.0.2.zip	CloudSim 3.0.2: bug fix release	Nov 6, 2012	13.05MB
cloudsim-3.0.1.tar.gz	CloudSim 3.0.1: bug fix release	Oct 16, 2012	9.89MB
cloudsim-3.0.1.zip	CloudSim 3.0.1: bug fix release	Oct 16, 2012	13.04MB
cloudsim-3.0.tar.gz	CloudSim 3.0	Jan 10, 2012	9.89MB
cloudsim-3.0.zip	CloudSim 3.0	Jan 10, 2012	13MB

2. OpenEclipse



3. Create a new Java Project: File ->New

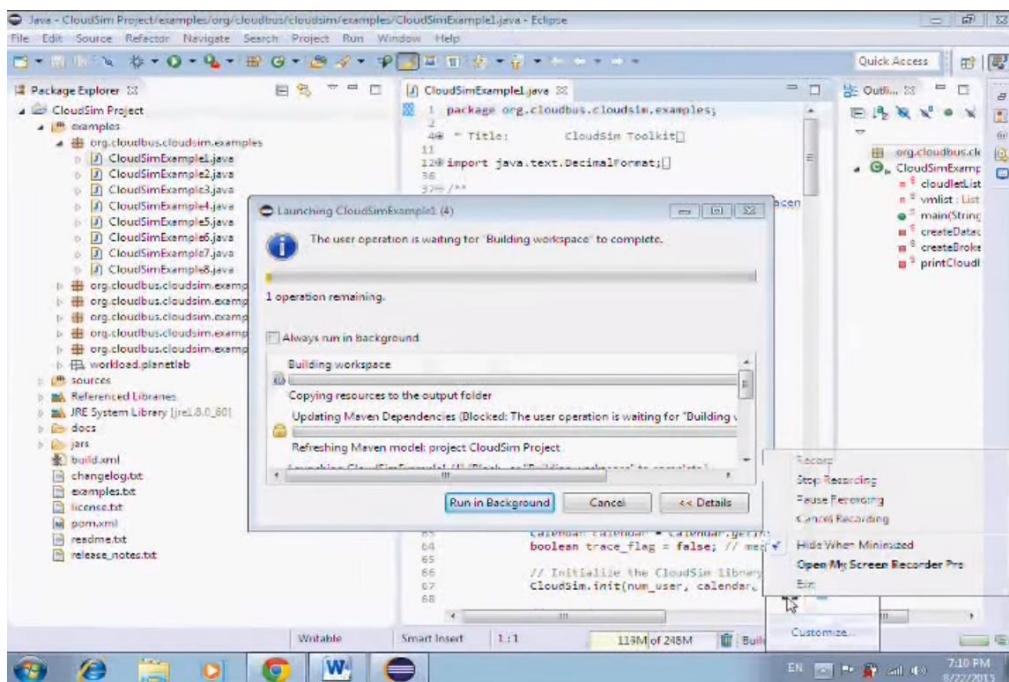
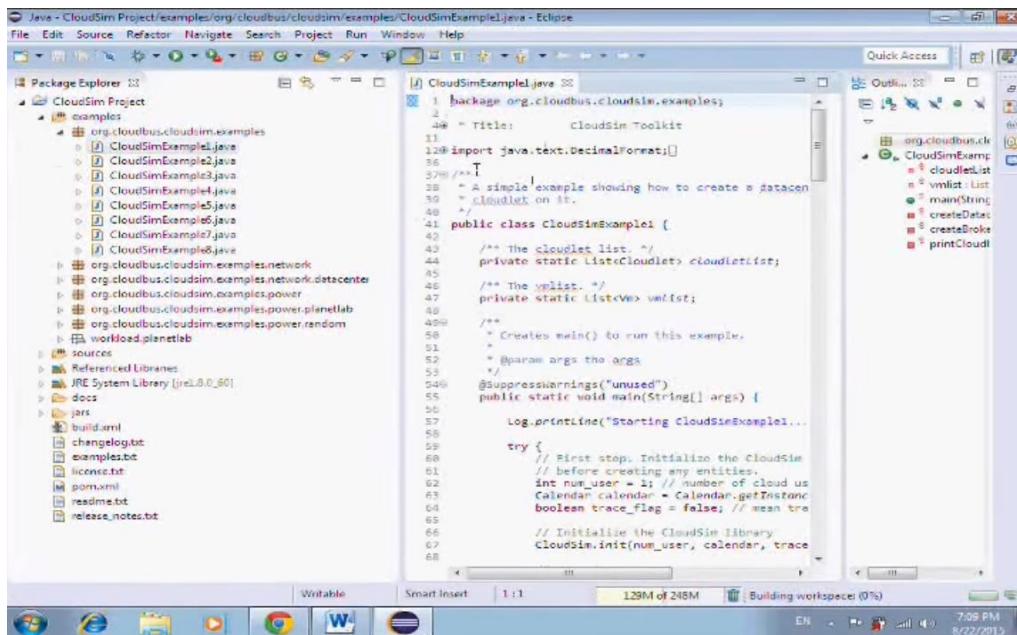


4. Import an unpacked CloudSim project into the new JavaProject

5. The first step is to initialise the CloudSim package by initialising the CloudSim library, as follows:
CloudSim.init(num_user, calendar,trace_flag)

6. Data centres are the resource providers in CloudSim; hence, creation of data centres is a second step. To create Datacenter, you need the DatacenterCharacteristics object that stores the properties of a data centre such as architecture, OS, list of machines, allocation policy that covers the time or spaceshared, the time zone and itsprice:

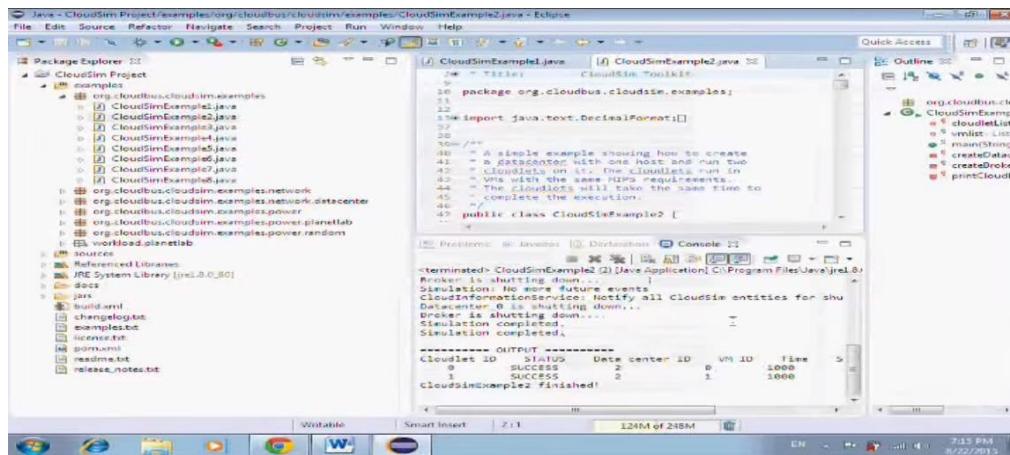
```
Datacenter datacenter9883 = new Datacenter(name, characteristics, new
VmAllocationPolicySimple(hostList),store
```



7. The third step is to create a broker: DatacenterBroker broker =createBroker();

8. The fourth step is to create one virtual machine unique ID of the VM, userId ID of the VM's owner, mips, number Of Pes amount of CPUs, amount of RAM, amount of bandwidth, amount of storage, virtual machine monitor, and cloudletScheduler policy forcloudlets:

```
Vm vm = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared())
```



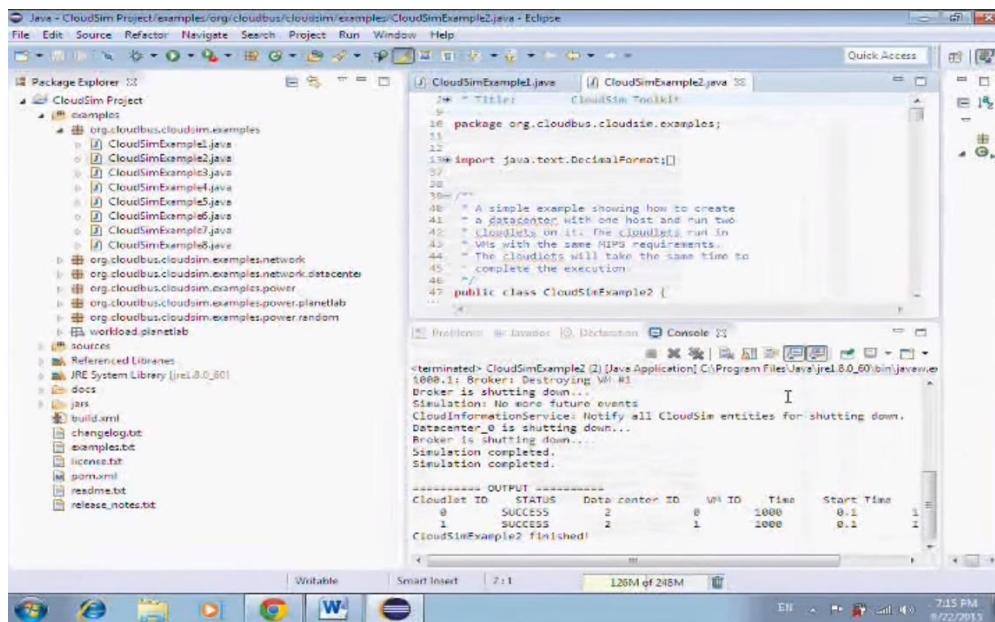
9. Submit the VM list to the broker: broker.submitVmList(vmList)

10. Create a cloudlet with length, file size, output size, and utilisationmodel:

```
Cloudlet cloudlet = new Cloudlet(id, length, pesNumber, fileSize, outputSize, utilizationModel, utilizationModel,
```

11. Submit the cloudlet list to the broker:

```
broker.submitCloudletList(cloudletList)
```



12. Start the simulation:

CloudSim.startSimulation() Sample Output from the Existing

Example: Starting CloudSimExample1...

Initialising...

Starting CloudSim version 3.0 Datacenter_0 is

Broker is starting...

Entities started.

0.0: Broker: Cloud Resource List received with 1 resource(s)

0.0: Broker: Trying to Create VM #0 in Datacenter_

0 0.1: Broker: VM #0 has been created in Datacenter #2, Host #

0 0.1: Broker: Sending cloudlet 0 to VM #0

400.1: Broker: Cloudlet 0 received

400.1: Broker: All Cloudlets executed. Finishing...

400.1: Broker: Destroying VM #0 Broker is shutting down...

Simulation: No more future events

CloudInformationService: Notify all CloudSim entities for shutting down.

Datacenter_0 is shutting down...

Broker is shutting down...

Simulation completed.

Simulation completed.

===== OUTPUT

CloudletID STATUS

0 SUCCESS 2 0 400 0.1 400.1

***** Datacenter: Datacenter_0 ***** User id Debt 3 35.6 CloudSimExample1 finished!

RESULT :

Thus the Simulating a cloud scenario using cloudsim and run a scheduling algorithms that is not present in cloudsim has been successfully completed

EX NO: 6

Find a procedure to transfer a file from one virtual machine

DATE :

to another virtual machine

Aim:

To find a procedure to transfer a file from one virtual machine to another virtual machine,

Procedure:

Step 1: Creating a Shared Folder in VirtualBox

Creating a shared folder between the guest and the host allows you to easily manage files which should be present on both machines.

By using the virtual machine on personal computer it will need to specify which folder to use as shared storage.

Z DRIVE gives you access to your Windows Account Z:\drive.

This is storage that is persistent to your SCS account and available as a network drive on the lab computers.

Temp gives you access to the folder found at D:\temp on the lab computer.

Files stored in this folder are local to the machine, meaning that they can be accessed **faster**, but will **delete** from the system when you log out.

Shared Folders on Personal Computers

If you are using your own personal machine, you will need to configure VirtualBox to look in the right place for your shared files.

Click on the guest machine to share files with. Select the guest *Settings* and **navigate** to *Shared Folders* on the left side menu. To create a new shared folder, either click the **New Folder** icon on the right menu **or** right click the empty list of shared folders and click **Add Shared Folder**.

- **Folder Path:** The folder name on the **host** machine. Click the drop down menu and navigate to the folder you would like to share.
- **Folder Name:** This is the name of the folder as it will appear on the **guest** machine.
- **Read-Only:** If you check read-only, the **guest** machine will be unable to write changes to the folder. This is valuable when you only want to send files *to* the virtual machine, but do not want to risk having the files modified by the guest.
- **Auto-Mount:** When any external storage is connected to a computer it must be *mounted* in order to be used. It is recommended that you turn on auto-mounting, unless you are familiar with the process of mounting a drive yourself.
- **Mount Point:** Unless you already know about mount points, leave this blank.
- **Make Permanent:** If you check this, the shared folder will be a permanent **machine folder**. If it is not checked, the folder will not be shared after a shutdown.

Step 2: Dragging and Dropping Files in VirtualBox

To transfer a few files, simply drag and drop the files in. On the top bar of the running guest machine, click on ***Devices > Drag and Drop*** and make sure that ***Bidirectional*** is selected.

This means that will be able to drag files from the host to the guest and from the guest to the host. Once bidirectional drag and drop is checked, and should be able to begin dragging and dropping files.

Step 3: Managing Files with NextCloud

On any virtual machine, including VirtualBox, VMWare, or the virtual machines hosted on the SCS OpenStack, can access the SCS NextCloud services to move files between multiple machines and SCS Windows Account storage.

NextCloud offers all of SCS storage in one remote location, similar to use other file hosting services like Dropbox or Google Drive. Before trying to use NextCloud, check that whether it access the service by logging in here.

various file storage services available:

- **Linux Home:** These are the files from your SCS Linux Account
- **Windows Home:** These are the files from your SCS Windows Account and your lab Z:\ drive.
- **NextCloud:** In addition to the other storage accounts provided to you by the SCS, you can also upload up to 20GB of files directly to NextCloud.

With NextCloud, upload files from any machine with an internet connection and download them onto any other machine with an internet connection. Alternatively,

Upload files from personal PC onto the NextCloud storage, place it into the ***WindowsHome*** folder, and access those files from either the lab Z:\ drive or download them on a virtual machine like VirtualBox or OpenStack

Result:

Thus the procedure to transfer a file from one virtual machine to another virtual machine was executed successfully.

EX.NO: 7	Find a procedure to launch virtual machine using trystack
DATE :	(Online Openstack Demo Version)

AIM:

To install opennebula in ubuntu operating system for creating virtualization in cloud.

PROCEDURE:

Step 1. Installation of opennebula in the Frontend

1.1. *Install therepo*

1. Open Terminal (ctrl+alt+t) or from dashboard typeterminal
2. Here # indirectly tells to work onroot.
\$ indirectly tells to work on normal user

Add the OpenNebula repository: # - root user

```
# wget -q -O- http://downloads.opennebula.org/repo/Ubuntu/repo.key | apt-key add -
# echo "deb http://downloads.opennebula.org/repo/4.12/Ubuntu/14.04/ stable opennebula" \
> /etc/apt/sources.list.d/opennebula.list
```

1.2. Install the requiredpackages

```
# apt-get update
# apt-get install opennebula opennebula-sunstone nfs-kernel-server
```

1.3. Configure and Start theservices

There are two main processes that must be started, the main OpenNebula daemon: **oned**, and the graphical user interface: **sunstone**.

Sunstone listens only in the loopback interface by default for security reasons. To change it edit

```
# gedit /etc/one/sunstone-server.conf
and change :host: 127.0.0.1 to :host: 0.0.0.0.
Now we must restart Sunstone:
# /etc/init.d/opennebula-sunstone restart
```

1.4. Configure SSH Public Key

OpenNebula will need SSH for passwordless from any node (including the frontend) to any other node.

To do so run the following commands:

```
# su - oneadmin
$ cp ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
```

Add the following snippet to `~/.ssh/config` so it doesn't prompt to add the keys to the `known_hosts` file:

```
$ cat<< EOT > ~/.ssh/config#Type the below commands
Host *
```

EOT

S
t
r
i
c

t
H
o
s
t
K
e
y
C
h
e
c
k
i
n
g
n
o
U
s
e
r
K

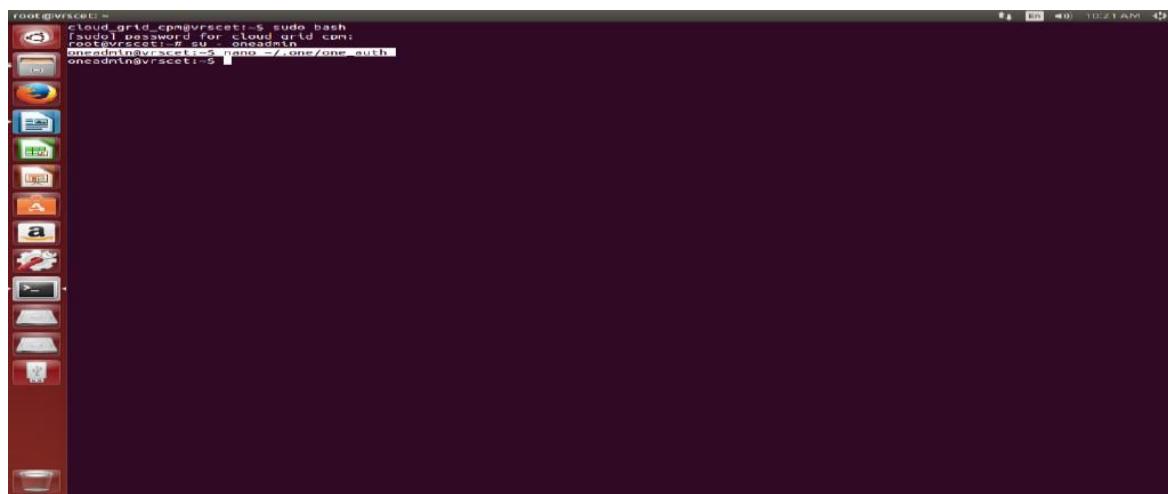
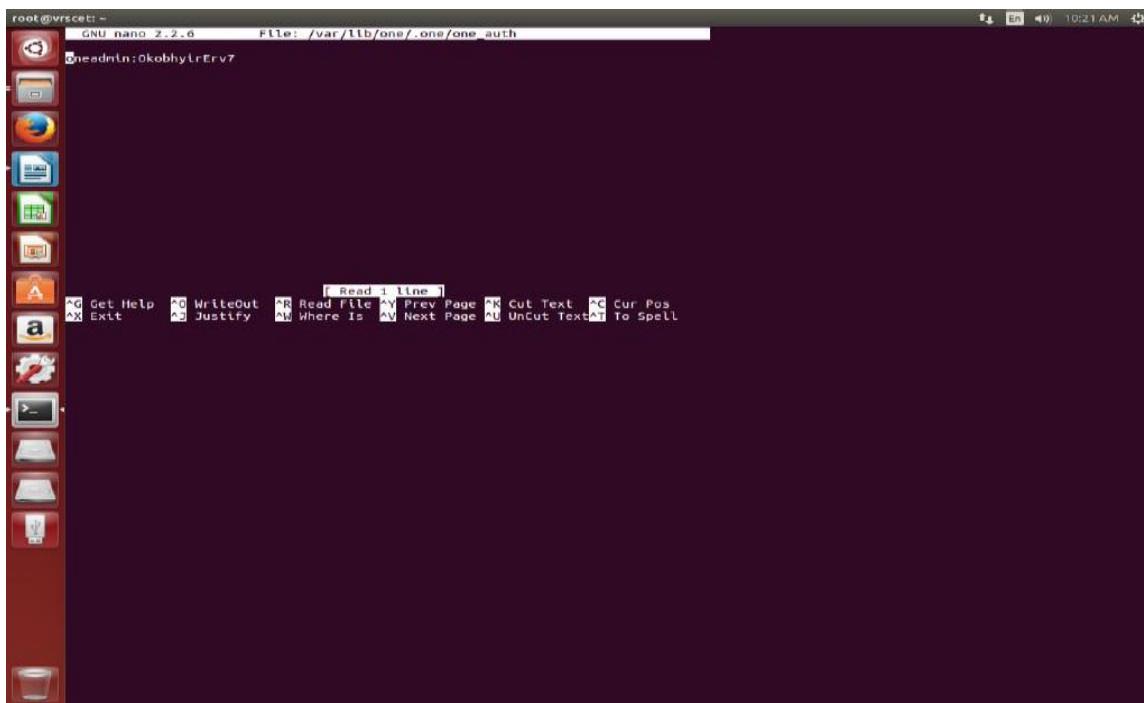
n
o
w
n
H
o
s
t
s
F
i
l
e
/
d
e
v
/
n
u
l
l

\$ chmod 600 ~/.ssh/config

Step 2. Basic Usage

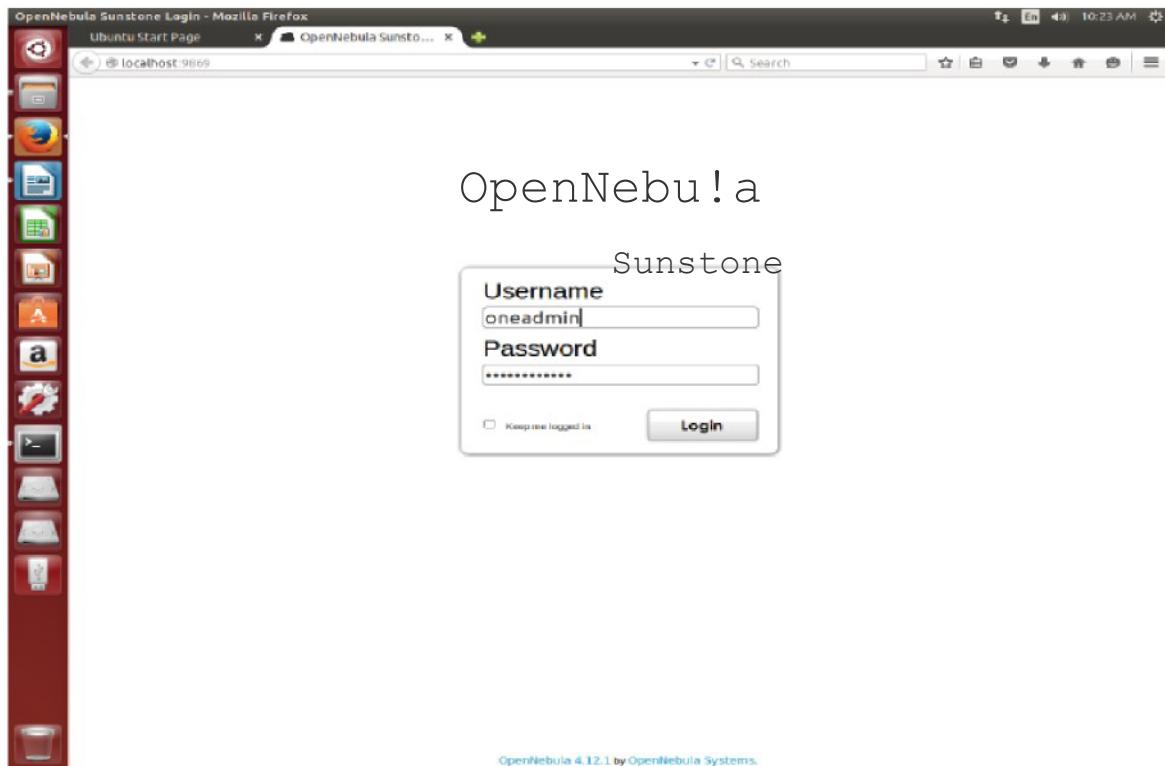
The default password for the oneadmin user can be found in `~/.one/one_auth` which is randomly generated on every installation.

```
$ nano ~/.one/one_auth
```



**Open mozilla
firefox
localhost:9869**

**Enter Username : oneadmin
Password : from `~/.one/one_auth` (file)**



The screenshot shows a Firefox browser window titled "OpenNebula Sunstone: Cloud Operations Center - Mozilla Firefox". The address bar displays "localhost:9869". The main content area is the "Cloud Operations Center" dashboard. On the left, there is a sidebar with navigation links: "Dashboard", "system", "Virtual Resources", "Infrastructure", "Marketplace", "OneFlow", "Support" (with a note "Not connected"), and a "Sign in" button. The main dashboard has three main sections: "Virtual Machines", "Hosts", and "Users".
Virtual Machines: Shows 0 TOTAL VMs. CPU hours: 0 / 0. Memory GB hours: 0 / 0. Status: 0 ACTIVE, 0 PENDING, 0 FAILED. Note: "There is no information available". Action: "+ Create".
Hosts: Shows 0 TOTAL hosts. ALLOCATED CPU: 0 % / 0 %. ALLOCATED MEMORY: 0 % / 0 KB. REAL CPU: 0 %. REAL MEMORY: 0 %. Status: 0 ON, 0 OFF, 0 ERROR. Action: "+ Create".
Users: Shows 2 USERS and 2 GROUPS. CPU hours: 0 / 0. Memory GB hours: 0 / 0. Status: There is no information available. Action: "+ Create".

Creating Oneimage, onetemplate and one vm

Step 1.

Move to the datastores folder.

```
o  
n  
e  
a  
d  
m  
i  
n  
@  
l  
i  
n  
u  
x  
:  
~  
$  
c  
d  
d  
a  
t  
a  
s  
t  
o  
r  
e  
s  
o  
n  
e  
a  
d  
m  
i  
n  
@  
l  
i  
n  
u  
x  
:  
~  
/  
d  
a  
t  
a  
s  
t  
o
```

r
e
s
\$

Creating oneimage

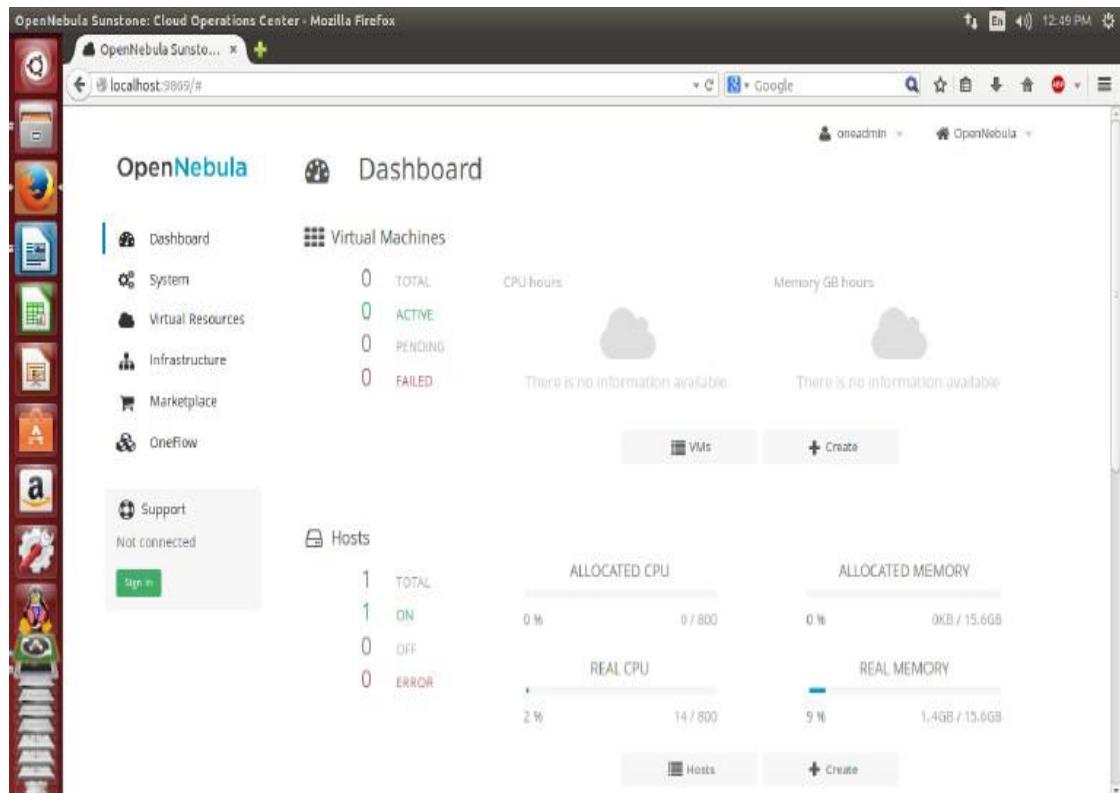
```
oneadmin@linux:~/datastores$  
oneimage create --name  
"ttylinux" --path  
"/home/linux/Downloads/source/  
ttylinux.img" --driver raw --  
datastore default
```

Creating One Template:

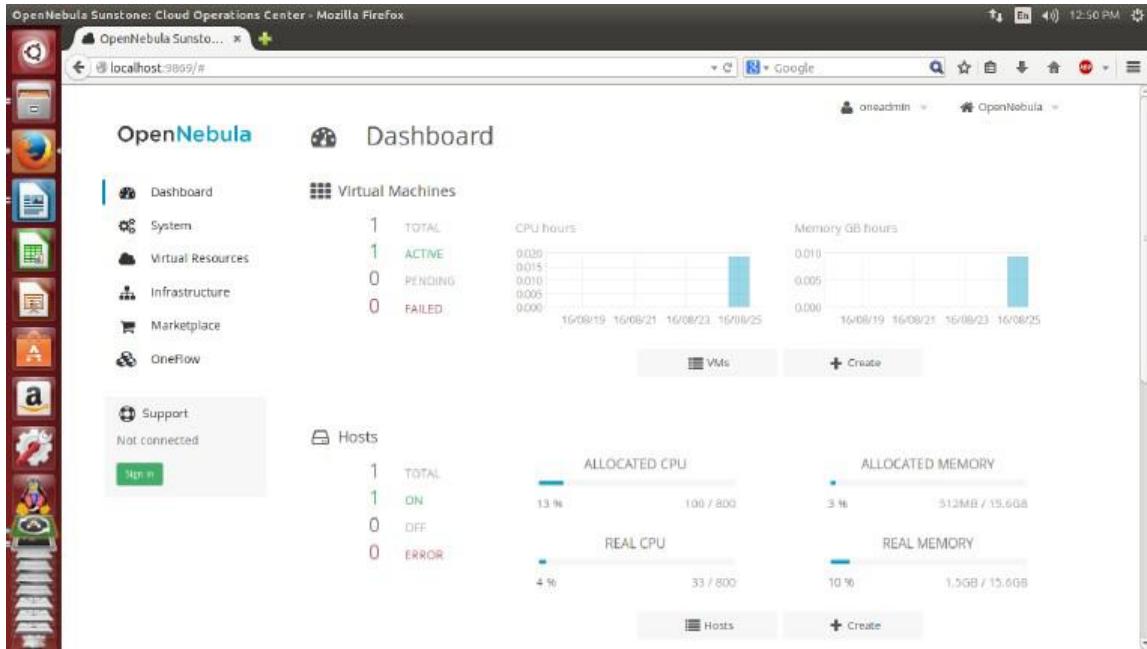
```
oneadmin@linux:~/datastores$  
onetemplate create --name "ttylinux" --  
cpu 1 --vcpu 1 --memory 512 --arch  
x86_64 --disk "ttylinux" --nic "private" --  
vnc --ssh
```

Instantiating OneVm (oneemplate)

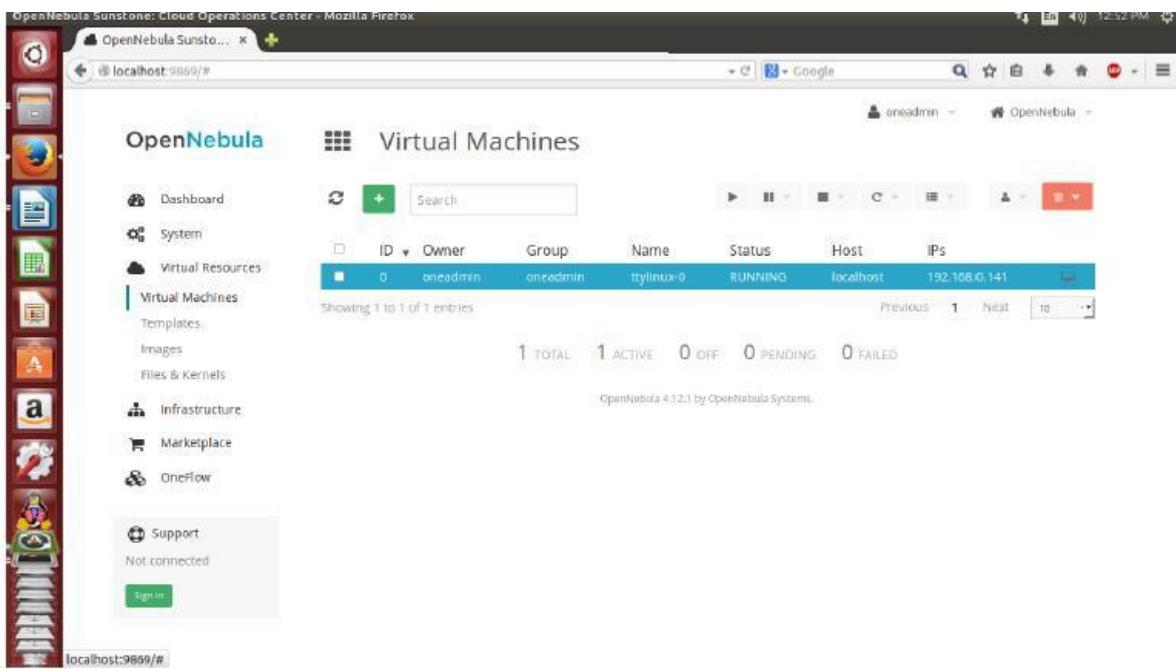
```
oneadmin@linux:~/datastores$ onetemplate  
instantiate "ttylinux"
```



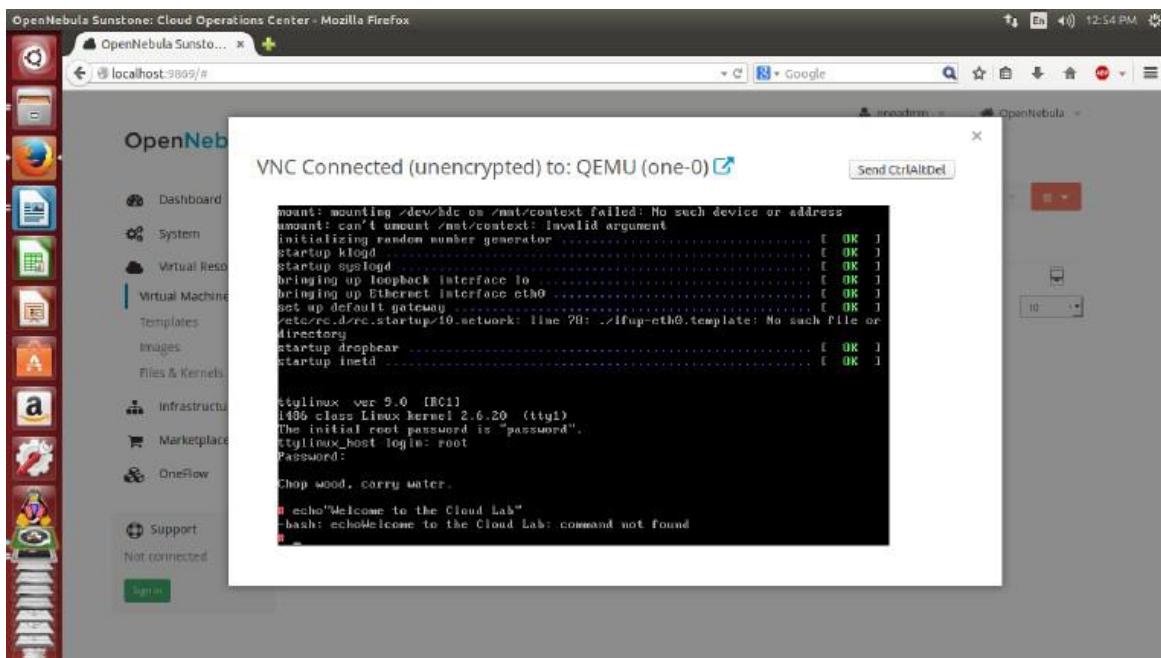
The above image before creating the vm. Refresh and check once the above commands are executed.



Step 2: Opening the VM through opennebula.



Click the corner computer symbol icon, it will ask for the username and password. By default the username is root and password is password.



Through terminal you can access the vm by
 oneadmin@linux:~/datastores\$ sshroot@192.168.0.141
 and give password

```

root@linux:/var/lib/one
onedeaminglinux:~$ cd datastores
onedeaminglinux:~/datastores$ ls
0 1 2
onedeaminglinux:~/datastores$ onelimage create --name "ttylinux" --path "/home/linux/Downloads/source/ttylinux.img" --driver raw --datastore default
ID: 0
onedeaminglinux:~/datastores$ onetemplate create --name "ttylinux" --cpu 1 --vcpu 1 --memory 512 --arch x86_64 --disk "ttylinux" --nic "private"
--vnc --ssh
ID: 0
onedeaminglinux:~/datastores$ onelimage list
+---+ ID USER GROUP NAME DATASTORE SIZE TYPE PER STAT RVMS
| 0 | onedeamin onedeamin ttylinux default 40M OS No ready 0
onedeaminglinux:~/datastores$ onetemplate list
+---+ ID USER GROUP NAME REGTIME
| 0 | onedeamin onedeamin ttylinux 08/25 12:46:17
onedeaminglinux:~/datastores$ onetemplate instantiate "ttylinux"
VM ID: 0
onedeaminglinux:~/datastores$ ssh root@192.168.0.141
Warning: Permanently added '192.168.0.141' (RSA) to the list of known hosts.
root@192.168.0.141's password:
*Chop wood, carry water.

# echo "Welcome"
-bash: echoWelcome: command not found
# logout
Connection to 192.168.0.141 closed.
onedeaminglinux:~/datastores$ 

```

Step 3: Similarly you can create as much vm your machine supports and can access only 5vm at a time since we limited our ip range upto 5 in mynetwork.one
 You can install ubuntu, centos and etc.,

Change the unbold data in the below command and install for various vm size.

Creating One Template:

```
oneadmin@linux:~/datastores$ onetemplate create --name "ttylinux" --cpu1 --vcpu 1 --memory  
512 --arch x86_64 --disk "ttylinux" --nic "private" --vnc --ssh
```

Instantiating OneVm (oneemplate)

```
oneadmin@linux:~/datastores$ onetemplate instantiate "ttylinux"
```

RESULT:

Thus the opennebula was installed and virtual machines were created successfully.

EX.NO :8	Find procedure to set up the one node Hadoop cluster
DATE :	

AIM:

To find the procedure for setting up the one hadoop cluster in the linux platform.

PROCEDURES:

Step 1: Download the latest sun java and apache hadoop from the official website.

Step 2: To install Java and Hadoop follow the below lines

1. InstallJava

a. Extract the Downloaded java tar.gz file in Downloads / Documentsfolder

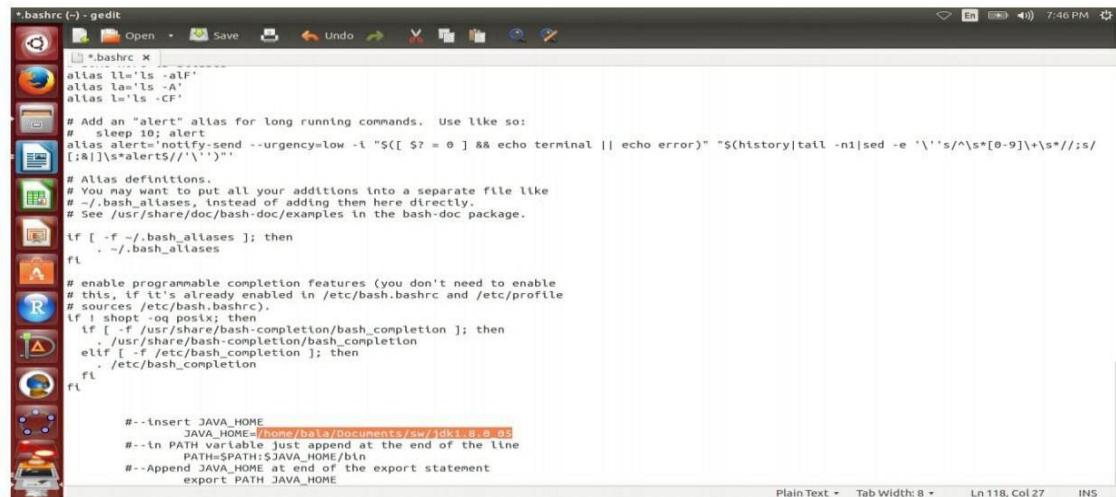
b. Open Terminal by pressing **ctrl+alt+t**

c. In Terminal, type **\$gedit~/.bashrc**

d. At the bottom paste the following lines by changing the path alone

```
--insert JAVA_HOME JAVA_HOME=/opt/jdk1.8.0_05
```

```
--in PATH variable just append at the end of the line PATH=$PATH:$JAVA_HOME/bin
```



```
*-bashrc (-) - gedit
  Open  Save  Undo  Redo  Cut  Copy  Paste  Find  Selection  Help  7:46 PM  ↻
* * bashrc x
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $(($? = 0)) && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''$'\''s/^/\''$'\''\s*[0-9]\+\s*//;s/\''$'\''/\'`\'')"'"

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.
if [ -f ~/.bash_aliases ]; then
  . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -q posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

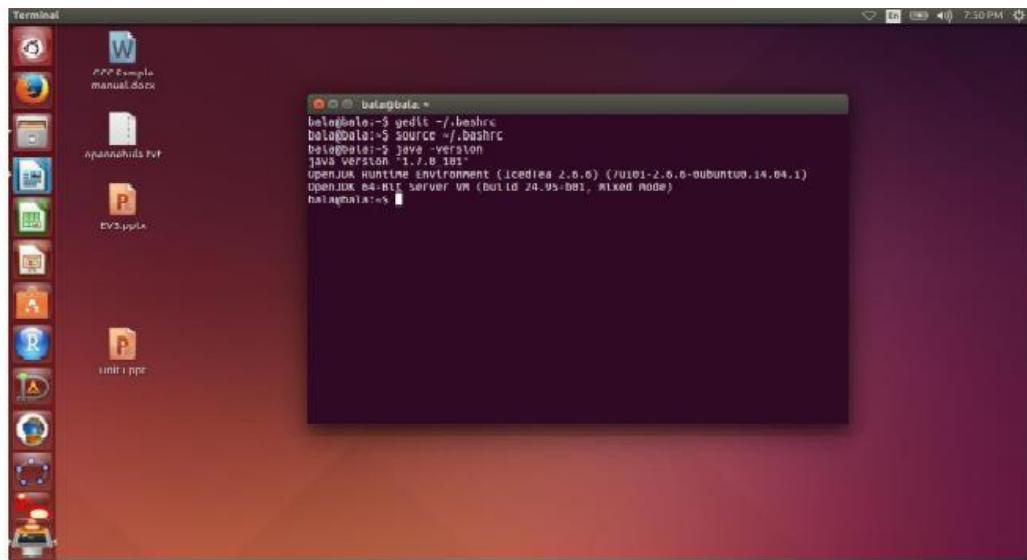
--insert JAVA_HOME
JAVA_HOME=/home/bala/Documents/sw/jdk1.8.0_05
--in PATH variable just append at the end of the line
PATH=$PATH:$JAVA_HOME/bin
--Append JAVA_HOME at end of the export statement
export PATH JAVA_HOME

Plain Text  Tab Width: 8  Ln 118, Col 27  INS
```

--Append JAVA_HOME at end of the export statement **export PATH JAVA_HOME**

e. Save the configuring by giving command as **\$ source~/.bashrc**

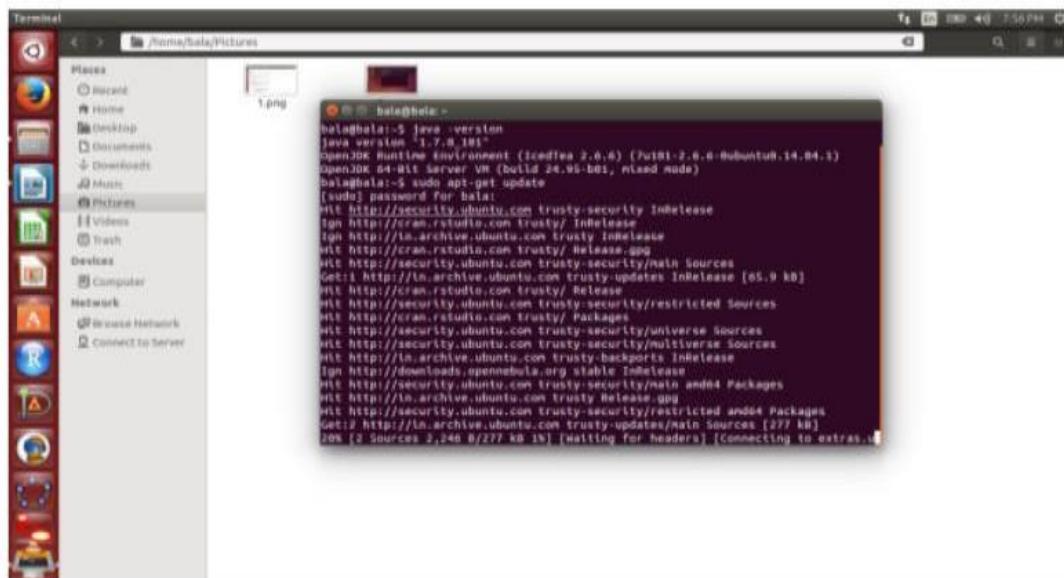
f. Check java has been successfully installed by typing **\$java--version**

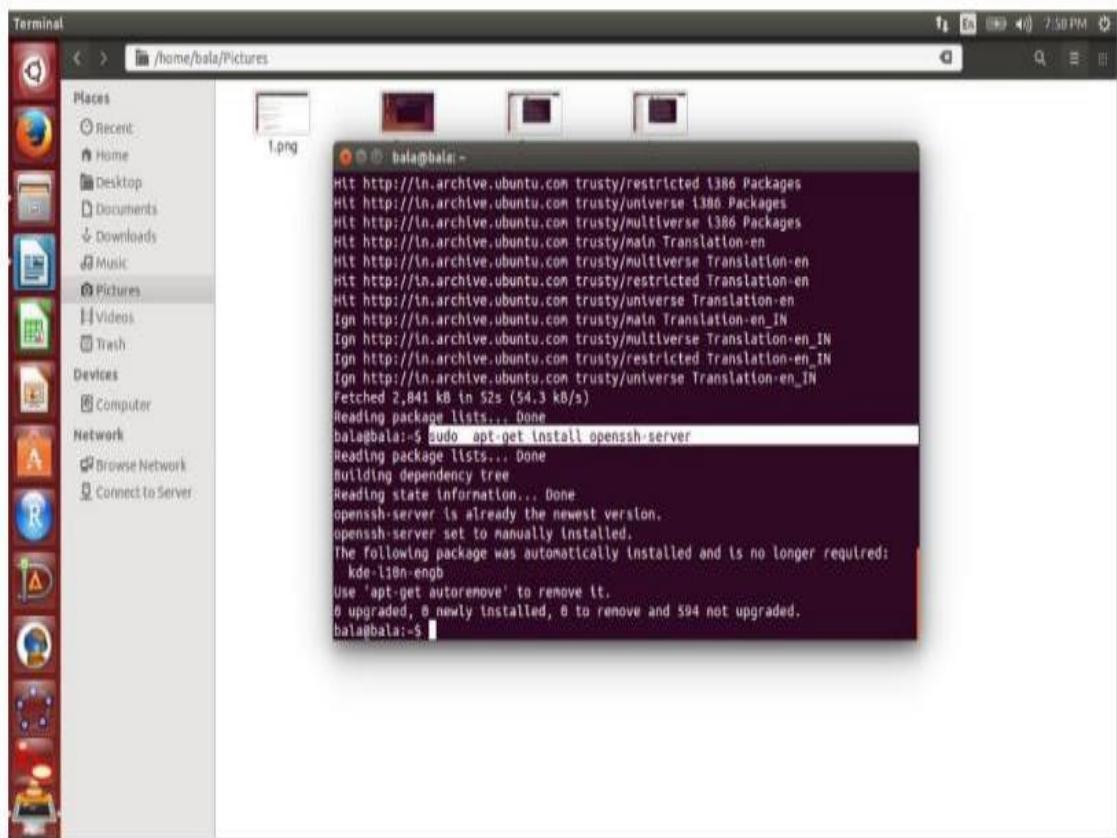
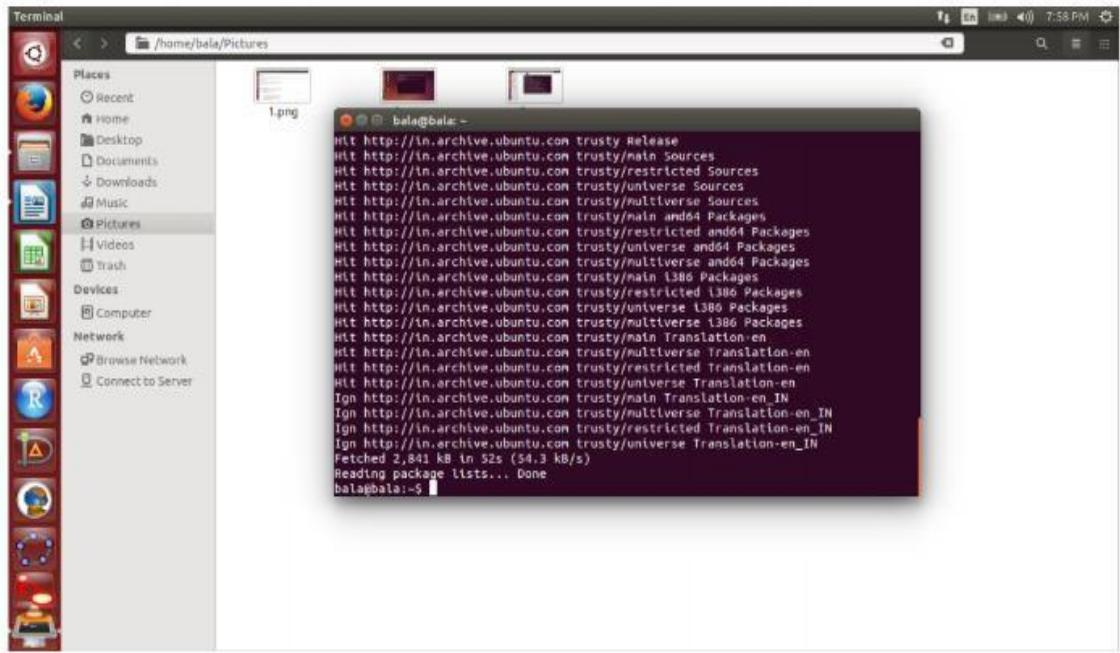


Step 3. Install ssh for passwordless authentication For passwordless authentication we need to do certain changes by following the below procedure and we need internet connection.

In Terminal: copy and the paste the below lines \$ sudo apt-get update .It will ask your root password. Give it

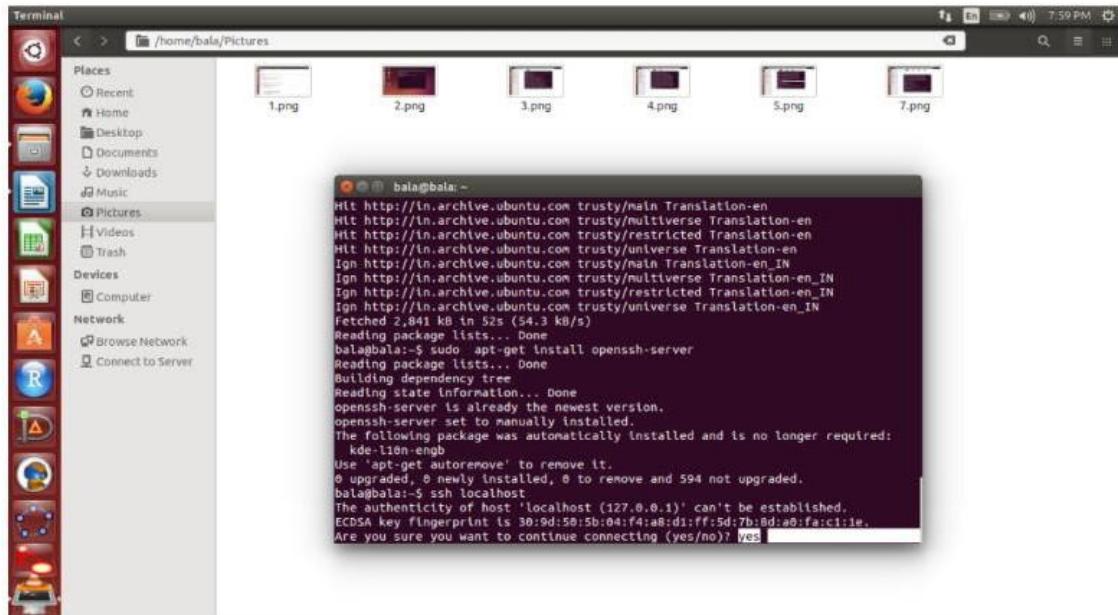
\$ sudo apt-get install openssh-server



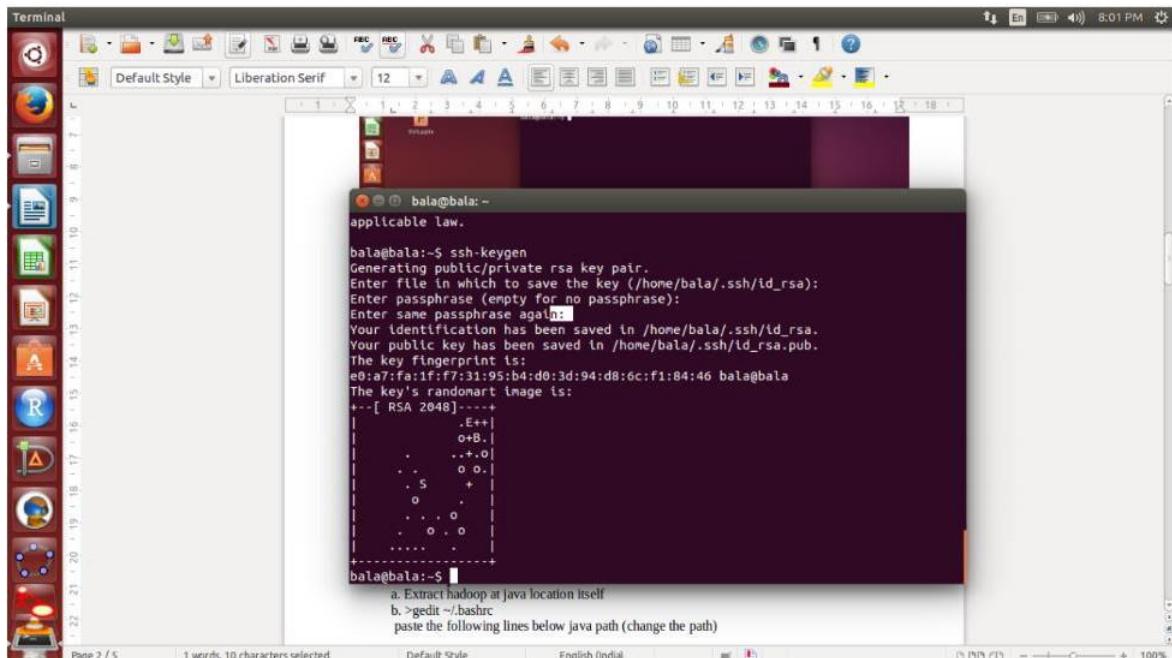


\$ sshlocalhost

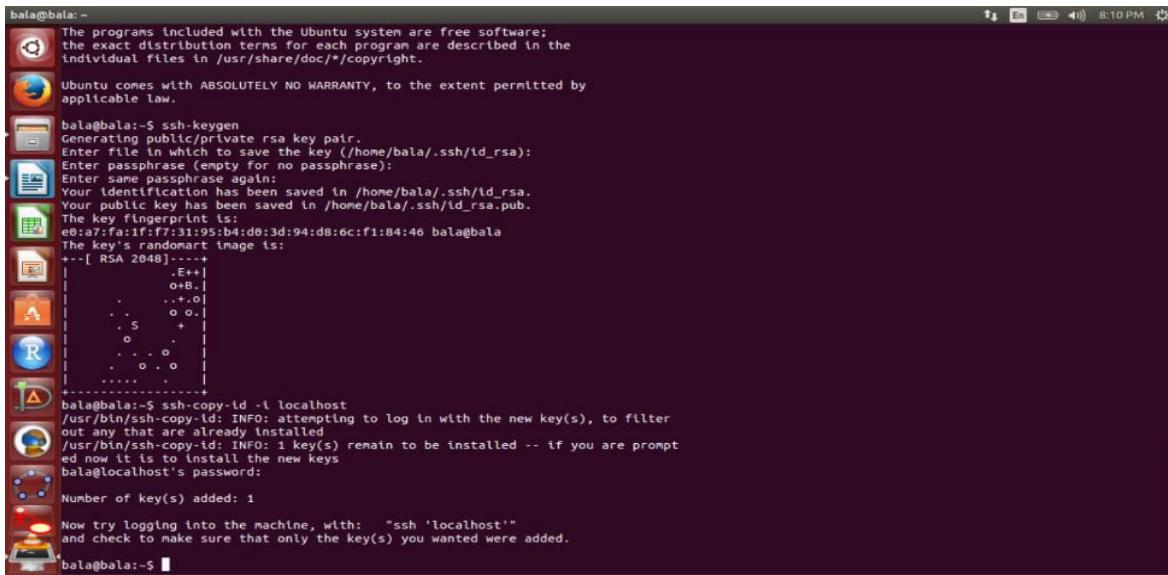
It also will ask ask root password



\$ ssh-keygen(Don't mention any path during key generation)



\$ ssh-copy-id -ilocalhost



```

bala@bala:~ 
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

bala@bala:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/bala/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/bala/.ssh/id_rsa.
The key fingerprint is:
e0:a7:fa:1f:f7:31:95:b4:d0:3d:94:d0:6c:f1:84:46 bala@bala
The key's randomart image is:
++-[ RSA 2048]----+
 .E+o| 
 o+B.| 
 ..+o| 
 . S +| 
 o o .| 
 . . o| 
 . o . o| 
 | . . .| 
 | . . .| 
 | . . .| 
bala@bala:~$ ssh-copy-id -i localhost
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
bala@localhost's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'localhost'"
and check to make sure that only the key(s) you wanted were added.

bala@bala:~$ 

```

Step 4. Installation Procedure of Hadoop

As like java, extract hadoop tar.gz file also and do the changes in bashrc file by copy and paste the following line

a. Extract hadoop at java located folder itself (Downloads orDocuments)

b. \$ gedit~/.bashrc

paste the following lines below java path (change the path)

#--insert HADOOP_PREFIX

HADOOP_PREFIX=/opt/hadoop-2.7.0

#--in PATH variable just append at the end of the line

PATH=\$PATH:\$HADOOP_PREFIX/bin

#--Append HADOOP_PREFIX at end of the export statement

export PATH JAVA_HOME HADOOP_PREFIX

c. save it by typing the below command interminal

\$ source ~/.bashrc

d. To check the installed path of Hadoop. Type thecommand

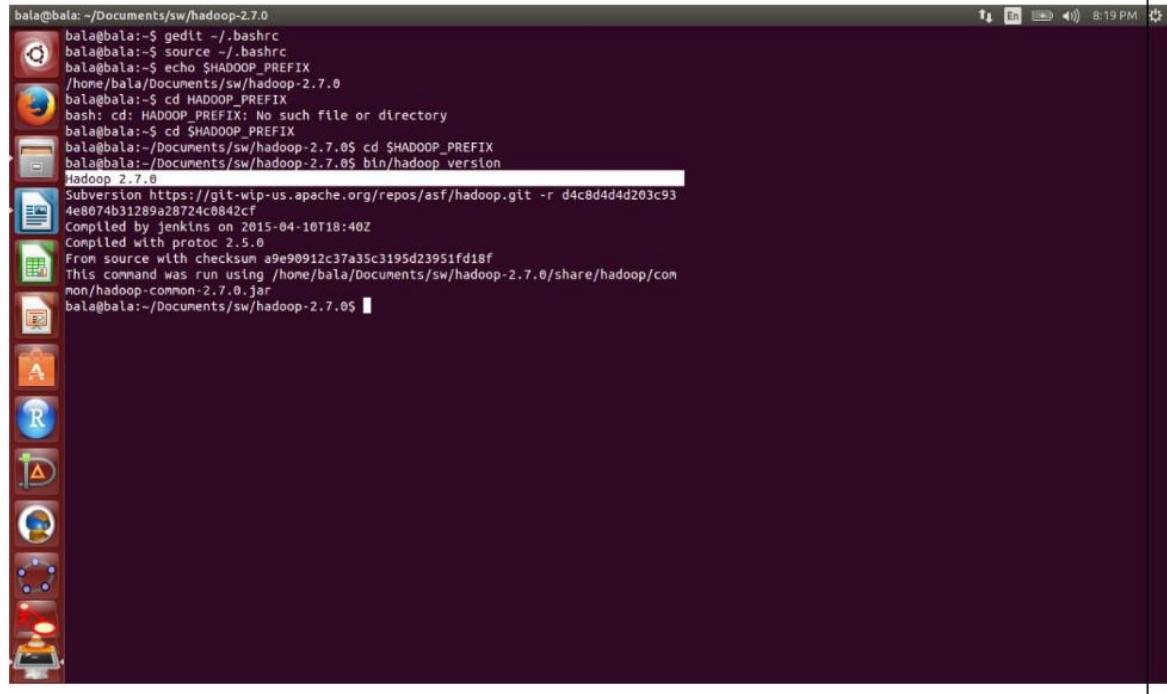
\$ echo \$HADOOP_PREFIX

e. Command is to get into the hadoop directoryis

\$ cd \$HADOOP_PREFIX

f. To check the installed hadoopversion

\$bin/hadoop version



```
bala@bala: ~/Documents/sw/hadoop-2.7.0
bala@bala: ~$ gedit ~/.bashrc
bala@bala: ~$ source ~/.bashrc
bala@bala: ~$ echo $HADOOP_PREFIX
/home/bala/Documents/sw/hadoop-2.7.0
bala@bala: ~$ cd HADOOP_PREFIX
bash: cd: HADOOP_PREFIX: No such file or directory
bala@bala: ~$ cd $HADOOP_PREFIX
bala@bala: ~/Documents/sw/hadoop-2.7.0$ bin/hadoop version
Hadoop 2.7.0
Subversion https://git-wip-us.apache.org/repos/asf/hadoop.git -r d4c8d4d4d203c93
4eb074b31289a28724c0842cf
Compiled by jenkins on 2015-04-10T18:40Z
compiled with protoc 2.5.0
From source with checksum a9e90912c37a35c3195d23951fd18f
This command was run using /home/bala/Documents/sw/hadoop-2.7.0/share/hadoop/com
mon/hadoop-common-2.7.0.jar
bala@bala: ~/Documents/sw/hadoop-2.7.0$
```

Step 5. Modifying the Hadoop configuration files

Do the things as like we did before using terminal

(i) cd\$HADOOP_PREFIX/etc/hadoop

\$ gedit hadoop-env.sh

(paste the java and hadoop path as the first two lines)

export JAVA_HOME=/usr/local/jdk1.8.0_05

export HADOOP_PREFIX=/opt/hadoop-2.7.0

(ii) Modify thecore-site.xml

\$ gedit core-site.xml

Paste the line within <configuration></configuration>

```
<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://localhost:9000</value>
</property>
</configuration>
```

```

core-site.xml (~Documents/sw/hadoop-2.7.0/etc/hadoop) - gedit
core-site.xml x
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://localhost:9000</value>
</property>
</configuration>

```

(iii) Modify thehdfs-site.xml

\$ gedit hdfs-site.xml

Paste the configuration file

```

<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
</configuration>

```

(iv) modify themapred-site.xml

\$ cpmapred-site.xml.template mapred-site.xml
\$ gedit mapred-site.xml

```

<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>

```

(v) Modiyarn-site.xml

```

>gedit yarn-site.xml
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
</configuration>

```

```

bala@bala: ~/Documents/sw/hadoop-2.7.0/etc/hadoop
balagbala:~$ gedit ~.bashrc
balagbala:~$ source ~.bashrc
balagbala:~$ echo $HADOOP_PREFIX
/home/bala/Documents/sw/hadoop-2.7.0
balagbala:~$ cd $HADOOP_PREFIX
bash: cd: HADOOP_PREFIX: No such file or directory
balagbala:~$ cd $HADOOP_PREFIX
balagbala:~/Documents/sw/hadoop-2.7.0$ cd $HADOOP_PREFIX
balagbala:~/Documents/sw/hadoop-2.7.0$ bin/hadoop version
Hadoop 2.7.0
Subversion https://git-wip-us.apache.org/repos/asf/hadoop.git -r d4c8d4d4d203c93
4e8074b31289a28724c0842cf
Compiled by jenkins on 2015-04-10T18:40Z
Compiled with protoc 2.5.0
From source with checksum a9e90912c37a35c3195d23951fd18F
This command was run using /home/bala/Documents/sw/hadoop-2.7.0/share/hadoop/com-
mon/hadoop-common-2.7.0.jar
balagbala:~/Documents/sw/hadoop-2.7.0$ gedit hadoop-env.sh
balagbala:~/Documents/sw/hadoop-2.7.0$ cd $HADOOP_PREFIX/etc/hadoop
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit hadoop-env
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit hadoop-env.sh
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit core-site.xml
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit hdfs-site.xml
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ cp mapred-site.xml.template mapred-site.xml
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit mapred-site.xml
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ gedit yarn-site.xml
balagbala:~/Documents/sw/hadoop-2.7.0/etc/hadoop$ 

```

Step 6. Formatting the HDFS file-system via the NameNode

```

$ cd $HADOOP_PREFIX
$ bin/hadoop namenode -format

```

```

bala@bala: ~/Documents/sw/hadoop-2.7.0
16/07/04 20:34:24 INFO blockmanagement.BlockManager: replicationRecheckInterval = 3000
16/07/04 20:34:24 INFO blockmanagement.BlockManager: encryptDataTransfer = false
16/07/04 20:34:24 INFO blockmanagement.BlockManager: maxNumBlocksToLog = 1000
16/07/04 20:34:24 INFO namenode.FSNamesystem: fsowner = bala (auth:SIMPLE)
16/07/04 20:34:24 INFO namenode.FSNamesystem: supergroup = supergroup
16/07/04 20:34:24 INFO namenode.FSNamesystem: isPermissionEnabled = true
16/07/04 20:34:24 INFO namenode.FSNamesystem: HA Enabled: false
16/07/04 20:34:24 INFO namenode.FSNamesystem: Append Enabled: true
16/07/04 20:34:25 INFO util.GSet: Computing capacity for map TNodeMap
16/07/04 20:34:25 INFO util.GSet: VH type = 64-bit
16/07/04 20:34:25 INFO util.GSet: 1.0% max memory 966.7 MB = 9.7 MB
16/07/04 20:34:25 INFO util.GSet: capacity = 2^20 = 1048576 entries
16/07/04 20:34:25 INFO namenode.FSDirectory: ACLs enabled? false
16/07/04 20:34:25 INFO namenode.FSDirectory: XAttrs enabled? true
16/07/04 20:34:25 INFO namenode.FSDirectory: Maximum size of an xattr: 16384
16/07/04 20:34:25 INFO namenode.NameNode: Caching file names occurring more than 10 times
16/07/04 20:34:25 INFO util.GSet: Computing capacity for map cachedblocks
16/07/04 20:34:25 INFO util.GSet: VH type = 64-bit
16/07/04 20:34:25 INFO util.GSet: 0.25K max memory 966.7 MB = 2.4 MB
16/07/04 20:34:25 INFO util.GSet: capacity = 2^18 = 262144 entries
16/07/04 20:34:25 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.9990000128746033
16/07/04 20:34:25 INFO namenode.FSNamesystem: dfs.namenode.safemode.mln.datanodes = 0
16/07/04 20:34:25 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension = 30000
16/07/04 20:34:25 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
16/07/04 20:34:25 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
16/07/04 20:34:25 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
16/07/04 20:34:25 INFO namenode.FSNamesystem: Retry cache on namenode ls enabled
16/07/04 20:34:25 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
16/07/04 20:34:25 INFO util.GSet: Computing capacity for map NameNodeRetryCache
16/07/04 20:34:25 INFO util.GSet: VH type = 64-bit
16/07/04 20:34:25 INFO util.GSet: 0.02999999932944746% max memory 966.7 MB = 297.0 KB
16/07/04 20:34:25 INFO util.GSet: capacity = 2^15 = 32768 entries
16/07/04 20:34:25 INFO namenode.FSImage: Allocated new BlockPoolId: BP-236706538-127.0.1.1-1467644665613
16/07/04 20:34:25 INFO common.Storage: Storage directory /tmp/hadoop-bala/dfs/name has been successfully formatted.
16/07/04 20:34:26 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 8
16/07/04 20:34:26 INFO util.ExitUtil: Exiting with status 0
16/07/04 20:34:26 INFO namenode.NameNode: SHUTDOWN_MSG:
*****SHUTDOWN_MSG: Shutting down NameNode at bala/127.0.1.1*****
bala@bala:~/Documents/sw/hadoop-2.7.0$ 

```

after formatting, start the services

Step 7. Starting the services.

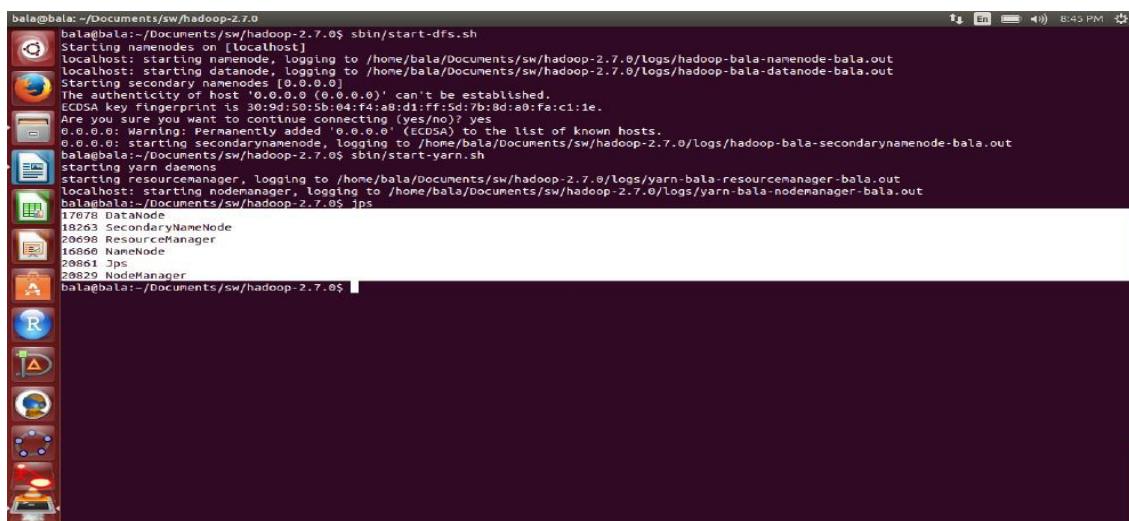
```
$sbin/start-dfs.sh#####it will start services by taking some time and it will ask permission  
give yes.  
$sbin/start-yarn.sh  
else  
$ sbin/start-all.sh  
to check running services  
>jps  
3200 DataNode  
12563 Jps  
4036 ResourceManager  
4172 NodeManager  
5158NameNode  
3685 SecondaryNameNode
```

Step 8 . Stopping Services

```
>sbin/stop-dfs.sh  
>sbin/stop-yarn.sh  
(or)  
>sbin/stop-all.sh
```

Once you start the services after stopped means it shows only 4 services.

```
>jps  
12563 Jps
```



```
balagbala:~/Documents/sw/hadoop-2.7.0$ sbin/start-dfs.sh  
balagbala:~/Documents/sw/hadoop-2.7.0$ Starting namenodes on [localhost]  
localhost: starting namenode, logging to /home/bala/Documents/sw/hadoop-2.7.0/logs/hadoop-bala-namenode-bala.out  
localhost: starting datanode, logging to /home/bala/Documents/sw/hadoop-2.7.0/logs/hadoop-bala-datanode-bala.out  
Starting secondarynamenode on [localhost]  
The authenticity of host '0.0.0.0 (0.0.0.0)' can't be established.  
ECDSA key fingerprint is 30:9d:50:5b:04:f4:a8:d1:ff:5d:7b:9d:a0:fe:c1:1e.  
Are you sure you want to continue connecting (yes/no)? yes  
0.0.0.0: Warning: Permanently added '0.0.0.0' (ECDSA) to the list of known hosts.  
0.0.0.0: starting secondarynamenode, logging to /home/bala/Documents/sw/hadoop-2.7.0/logs/hadoop-bala-secondarynamenode-bala.out  
balagbala:~/Documents/sw/hadoop-2.7.0$ sbin/start-yarn.sh  
starting yarn daemons  
starting resourcemanager, logging to /home/bala/Documents/sw/hadoop-2.7.0/logs/yarn-bala-resourcemanager-bala.out  
localhost: starting nodemanager, logging to /home/bala/Documents/sw/hadoop-2.7.0/logs/yarn-bala-nodemanager-bala.out  
balagbala:~/Documents/sw/hadoop-2.7.0$ jps  
17078 DataNode  
18263 SecondaryNameNode  
20860 ResourceManager  
16660 NameNode  
20861 Jps  
20829 NodeManager  
balagbala:~/Documents/sw/hadoop-2.7.0$
```

Step 9. only four services will run. To start datanode and name node we have to add some lines in **hdfs-site.xml**

In Terminal

```
$ cd $HADOOP_PREFIX/etc/hadoop
```

```
$gedit hdfs-site.xml      (Paste the belowlines)
<property>
<name>dfs.namenode.name.dir</name>
<value>/opt/name</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/opt/data</value>
</property>
```

```

<?xml version="1.0" encoding="UTF-8"?>
<xm...-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at
    http://www.apache.org/licenses/LICENSE-2.0
  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>/opt/name</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>/opt/data</value>
  </property>
</configuration>
```

Next do these procedures, for creating permanent log storage to namenode and datanode.

```
$ sudo mkdir /opt/name
$ sudo mkdir /opt/data
$ ls /opt/
$ ls -l /opt/ ##To change the directory from root user to admin user
$ sudo chown vrscet:vrscet -R /opt (root should be replaced by your system username)
$ ls -l /opt/
```

Step 10. Format the namenode

```
$ cd $HADOOP_PREFIX
$ bin/hadoopnamenode –format
```

Step 11. Start services

```
$sbin/start-dfs.sh
/sbin/start-yarn.sh
$jps
```

```
3200 DataNode
12563 Jps
```

4036 ResourceManager
4172 NodeManager
5158 NameNode
3685 SecondaryNameNode

Step 12. To view in Browser (Open Firefox and enter the below address)

localhost:50070
localhost:8088

The screenshot shows the 'Namenode Information' page from a Firefox browser. The URL is `localhost:9000/d/health.html#tab-overview`. The page has a green header bar with tabs for Hadoop, Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities. The main content area is titled 'Overview' for 'localhost:9000' (active). It contains several tables with cluster information:

Started:	Mon Jul 04 21:11:46 IST 2016
Version:	2.7.0. rd4c8d4d4d203c934e8074b31289a28724c0842cf
Compiled:	2015-04-10T18:40Z by Jenkins from (detached from d4c8d4d)
Cluster ID:	CID-5baebe8-89da-4dd6-a104-94ade0e21b65
Block Pool ID:	BP-24067461-127.0.1.1-1467646854273

Below this is a 'Summary' section with the following text:

Security is off.
Safermode is off.
1 files and directories, 0 blocks = 1 total filesystem object(s).
Heap Memory used 37.17 MB of 65.57 MB Heap Memory. Max Heap Memory is 966.69 MB.
Non Heap Memory used 45.86 MB of 46.46 MB Committed Non Heap Memory. Max Non Heap Memory is -1 B.

A note at the bottom states: 'Firefox automatically sends some data to Mozilla so that we can improve your experience.'

The screenshot shows the 'All Applications' page from a Firefox browser. The URL is `localhost:8088/cluster`. The page has a black header bar with tabs for Ubuntu Start Page, Namenode Information, and All Applications. The main content area is titled 'All Applications' and features a 'hadoop' logo. On the left, there is a sidebar with a tree view of the cluster, showing sections for Cluster, Scheduler, and Tools. The main area displays two tables: 'Cluster Metrics' and 'Scheduler Metrics'.

	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	Vcores Used	Vcores Total	Vcores Reserved	Active Nodes	Decommissioned Nodes
NEW	0	0	0	0	0	0 B	8 GB	0 B	0	8	0	1	0

The 'Scheduler Metrics' table shows the following data:

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:1>

Below these tables, there is a search bar and a message: 'Showing 0 to 0 of 0 entries'.

RESULT:

Thus single node hadoop cluster has been successfully created.