LAB MANUAL

CS8711 – CLOUD COMPUTING LABORATORY

Regulation 2017

Year / Semester : IV / VII

Sep 2021 – Nov 2021

LIST OF EXPERIMENTS

1. Install Virtualbox/VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
3. Install Google App Engine. Create *hello world* app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
8. Install Hadoop single node cluster and run simple applications like wordcount.

TABLE OF CONTENTS

S.NO.	DATE	EXPERIMENT TITLE	MARKS /10	SIGN.
1.		Install Virtualbox / VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.		
2.		Install a C compiler in the virtual machine created using virtual box and execute Simple Programs		
3.		Install Google App Engine. Create *hello world* app and other simple web applications using python/java.		
4.		Use GAE launcher to launch the web applications.		
5.		Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.		
6.		Find a procedure to transfer the files from one virtual machine to another virtual machine.		
7.		Find a procedure to launch		

		virtual machine using trystack (Online Openstack Demo Version)		
8.		Install Hadoop single node cluster and run simple applications like wordcount.		

EX NO. : 1

DATE:

Install Virtualbox / VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.

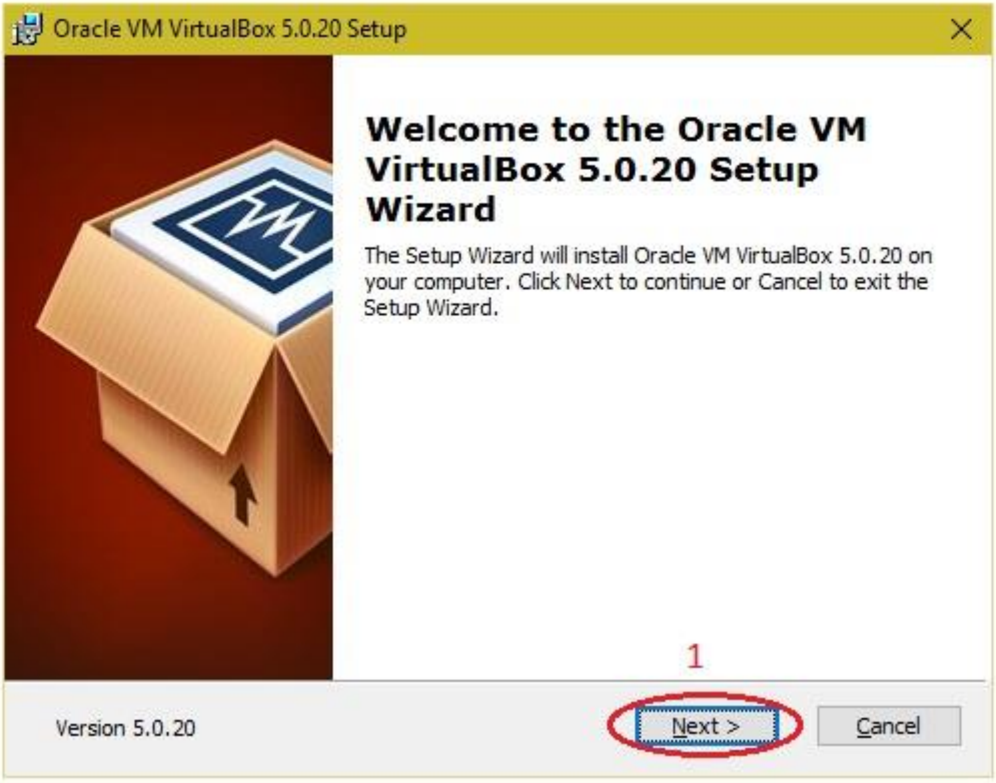
Aim:

To Install Virtualbox / VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.

PROCEDURE:

Steps to install Virtual Box:

1. Download the Virtual box exe and click the exe file…and select next button..



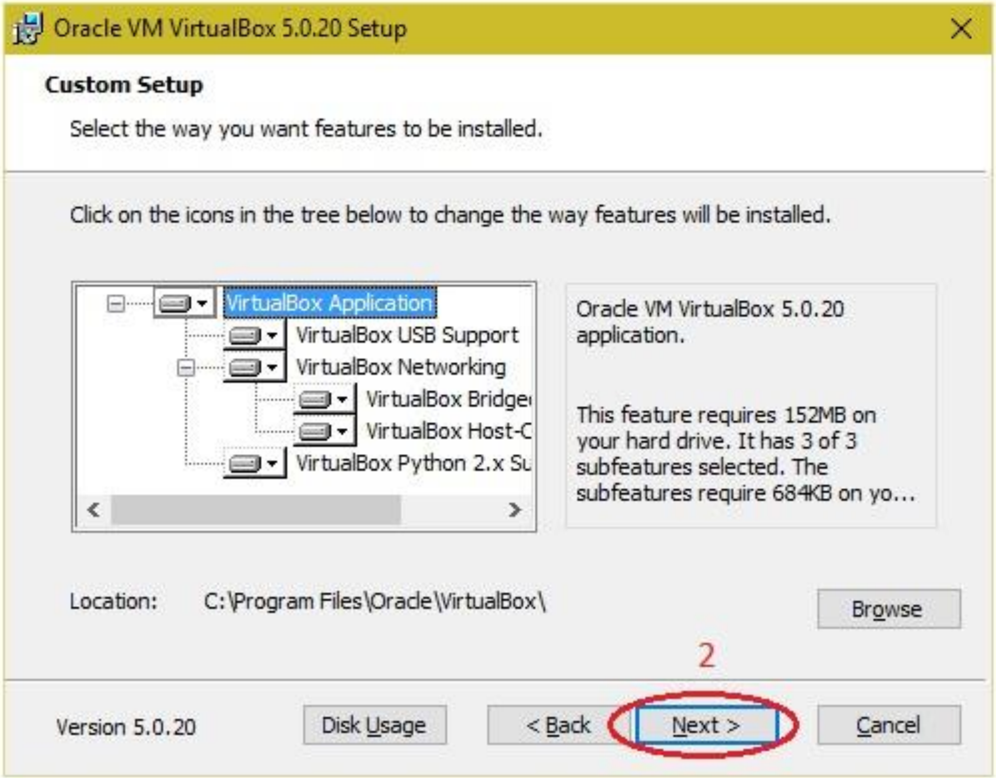
Oracle VM VirtualBox 5.0.20 Setup

Welcome to the Oracle VM VirtualBox 5.0.20 Setup Wizard

The Setup Wizard will install Oracle VM VirtualBox 5.0.20 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Version 5.0.20

2. Click the next button..



Oracle VM VirtualBox 5.0.20 Setup

Custom Setup

Select the way you want features to be installed.

Click on the icons in the tree below to change the way features will be installed.

- VirtualBox Application
  - VirtualBox USB Support
  - VirtualBox Networking
    - VirtualBox Bridge
    - VirtualBox Host-C
  - VirtualBox Python 2.x Su

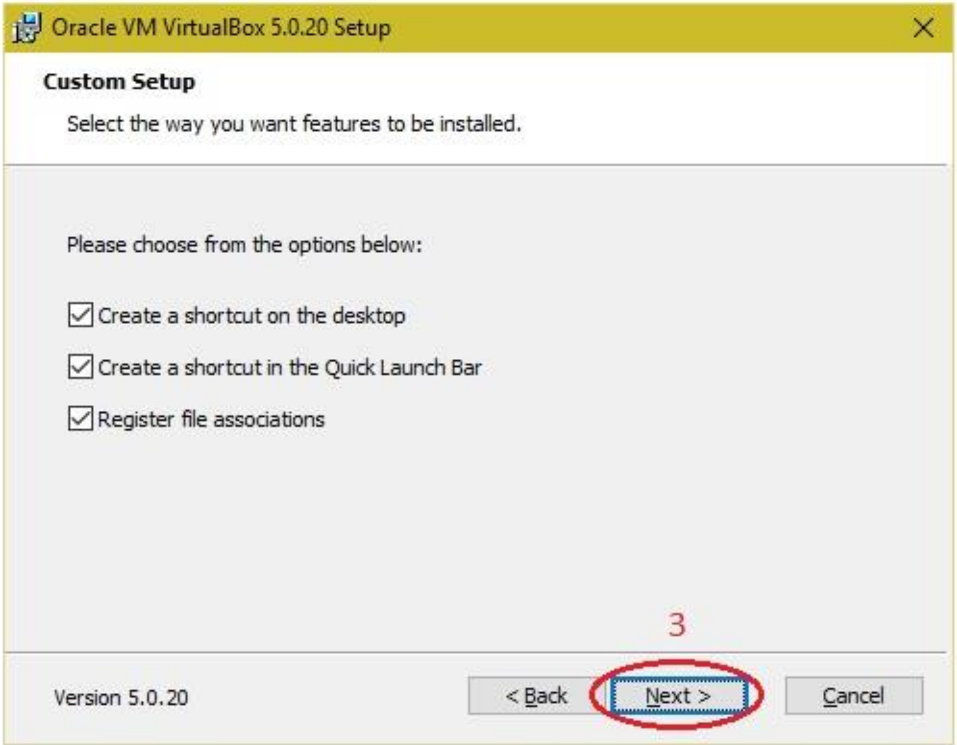
Oracle VM VirtualBox 5.0.20 application.

This feature requires 152MB on your hard drive. It has 3 of 3 subfeatures selected. The subfeatures require 684KB on yo...

Location: C:\Program Files\Oracle\VirtualBox\

Version 5.0.20

3. Click the next button



Oracle VM VirtualBox 5.0.20 Setup

Custom Setup

Select the way you want features to be installed.

Please choose from the options below:

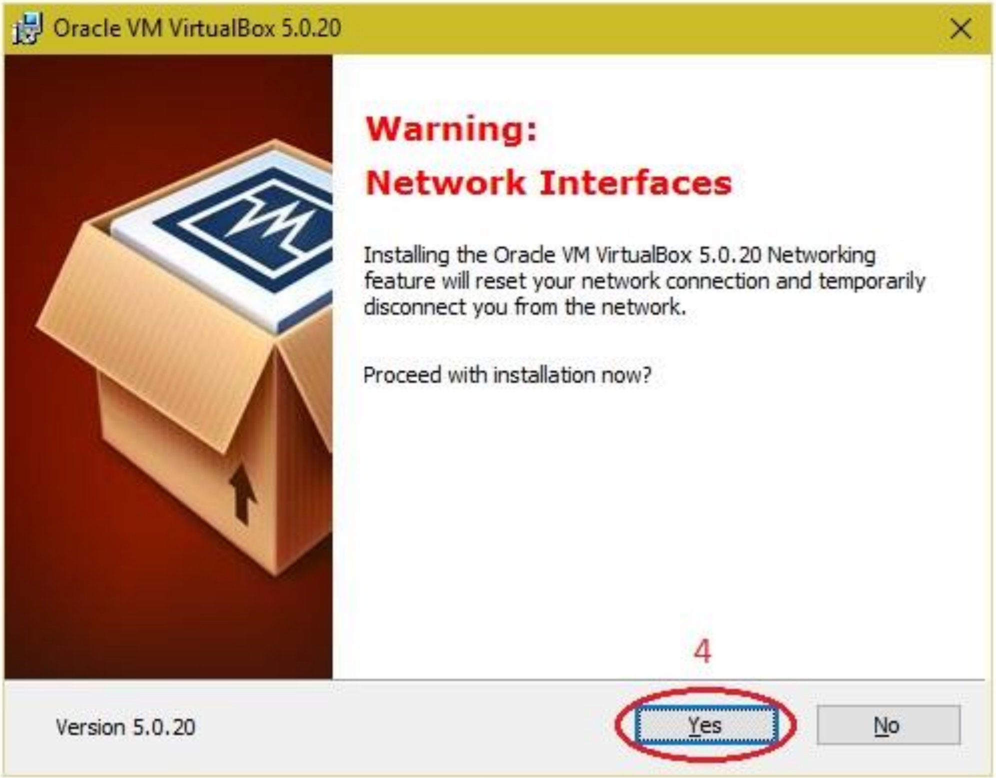
☑ Create a shortcut on the desktop

☑ Create a shortcut in the Quick Launch Bar

☑ Register file associations

Version 5.0.20

4. Click the YES button..



Oracle VM VirtualBox 5.0.20

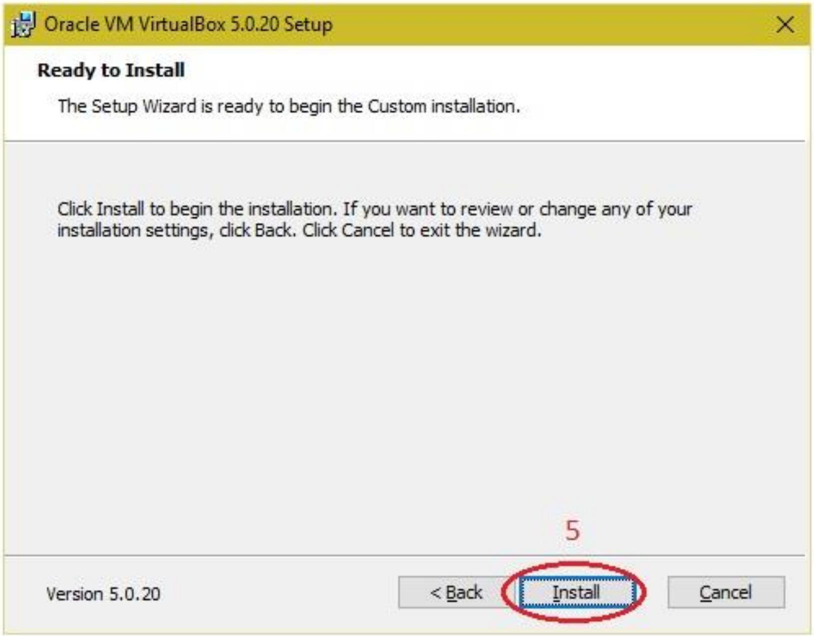
Warning: Network Interfaces

Installing the Oracle VM VirtualBox 5.0.20 Networking feature will reset your network connection and temporarily disconnect you from the network.

Proceed with installation now?

Version 5.0.20

5. Click the install button…



Oracle VM VirtualBox 5.0.20 Setup

Ready to Install

The Setup Wizard is ready to begin the Custom installation.

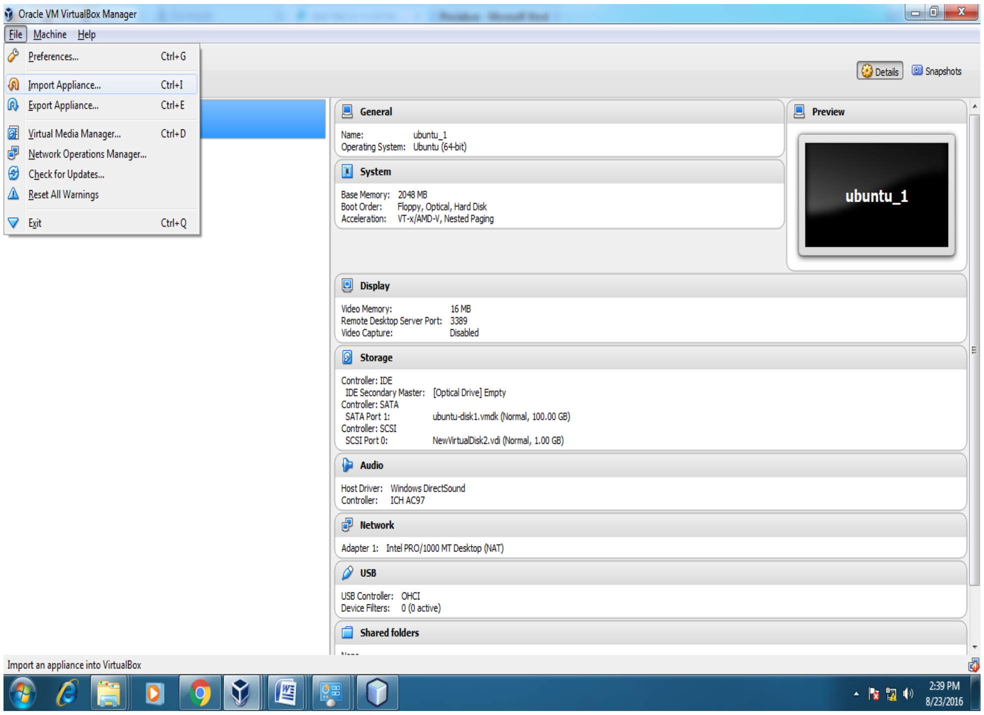
Click Install to begin the installation. If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

Version 5.0.20

6. Then installation was completed..the show virtual box icon on desktop screen….

Steps to import Open nebula sandbox:

1. Open Virtual box
2. File →import Appliance
3. Browse OpenNebula-Sandbox-5.0.ova file
4. Then go to setting, select Usb and choose USB 1.1
5. Then Start the Open Nebula
6. Login using username: root, password:opennebula


Oracle VM VirtualBox Manager
File
Machine
Help
Preferences... Ctrl+G
Import Appliance... Ctrl+I
Export Appliance... Ctrl+E
Virtual Media Manager... Ctrl+D
Network Operations Manager...
Check for Updates...
Reset All Warnings
Exit Ctrl+Q
Details
Snapshots
Preview
ubuntu_1
2:39 PM
8/23/2016

General

Name: ubuntu_1
Operating System: Ubuntu (64-bit)

System

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging

Display

Video Memory: 16 MB
Remote Desktop Server Port: 3389
Video Capture: Disabled

Storage

Controller: IDE
IDE Secondary Master: [Optical Drive] Empty
Controller: SATA
SATA Port 1: ubuntu-disk1.vmdk (Normal, 100.00 GB)
Controller: SCSI
SCSI Port 0: NewVirtualDisk2.vdi (Normal, 1.00 GB)

Audio

Host Driver: Windows DirectSound
Controller: ICH AC97

Network

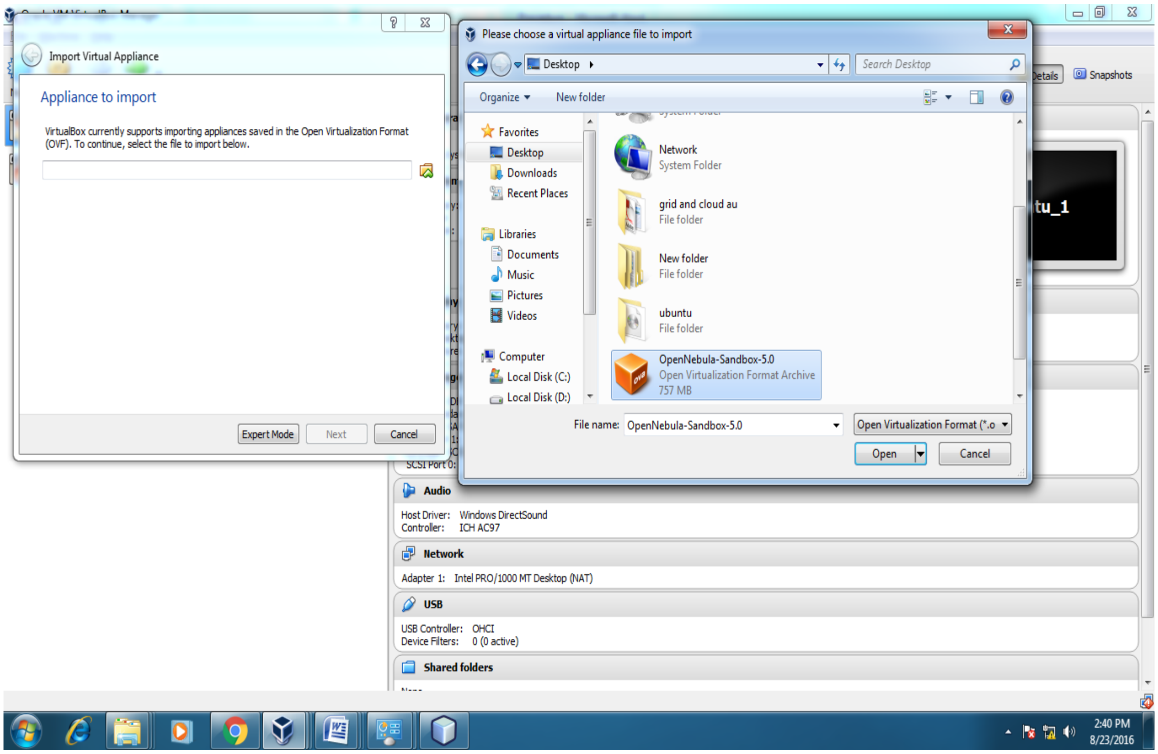
Adapter 1: Intel PRO/1000 MT Desktop (NAT)

USB

USB Controller: OHCI
Device Filters: 0 (0 active)

Shared folders

Import an appliance into VirtualBox



Import Virtual Appliance

Appliance to import

VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

Please choose a virtual appliance file to import

File name: OpenNebula-Sandbox-5.0

Open Virtualization Format (*.o

OpenNebula-Sandbox-5.0
Open Virtualization Format Archive
757 MB

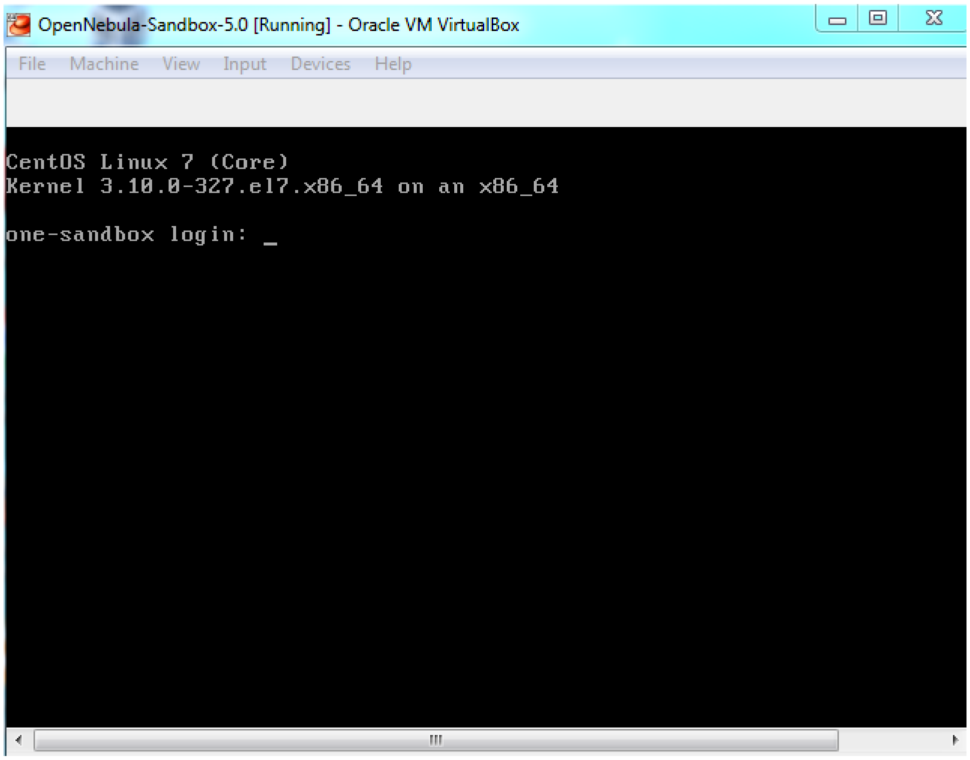
Host Driver: Windows DirectSound
Controller: ICH AC97

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

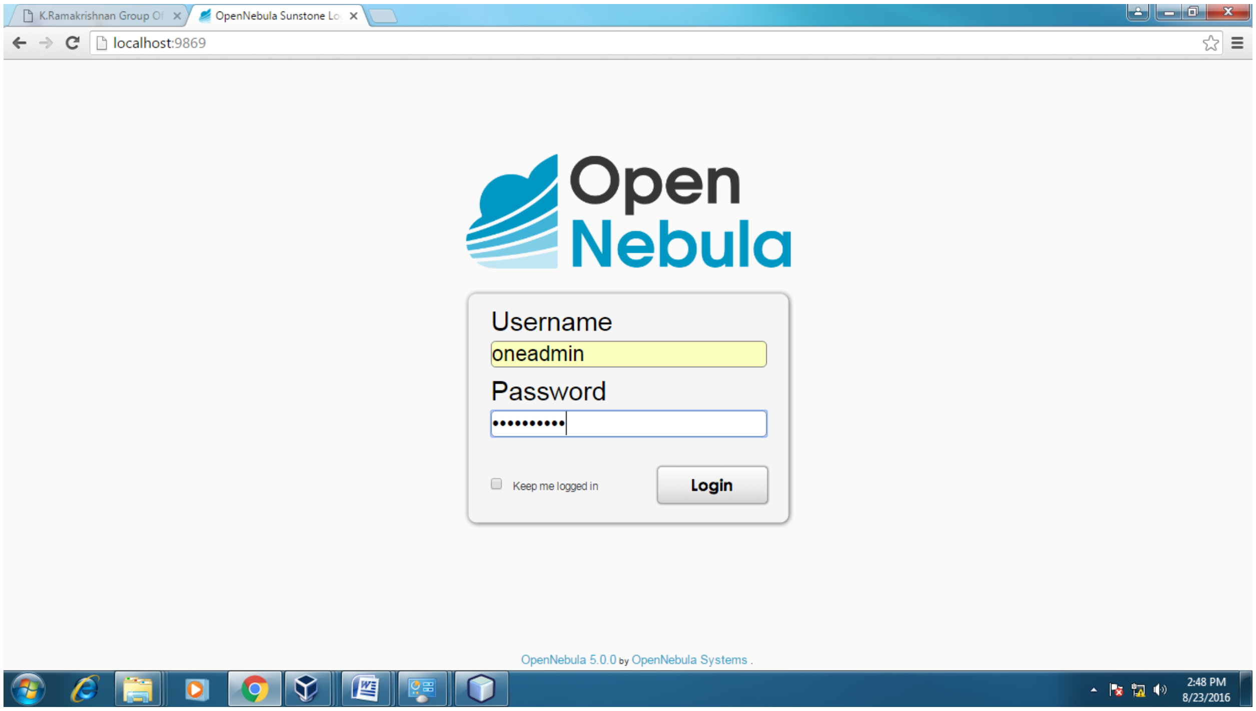
USB Controller: OHCI
Device Filters: 0 (0 active)

Steps to create Virtual Machine through opennebula

1. Open Browser, type localhost:9869
2. Login using username: oneadmin, password: opennebula
3. Click on instances, select VMs then follow the steps to create Virtaul machine
   a. Expand the + symbol
   b. Select user oneadmin
   c. Then enter the VM name,no.of instance, cpu.
   d. Then click on create button.
   e. Repeat the steps the C,D for creating more than one VMs.



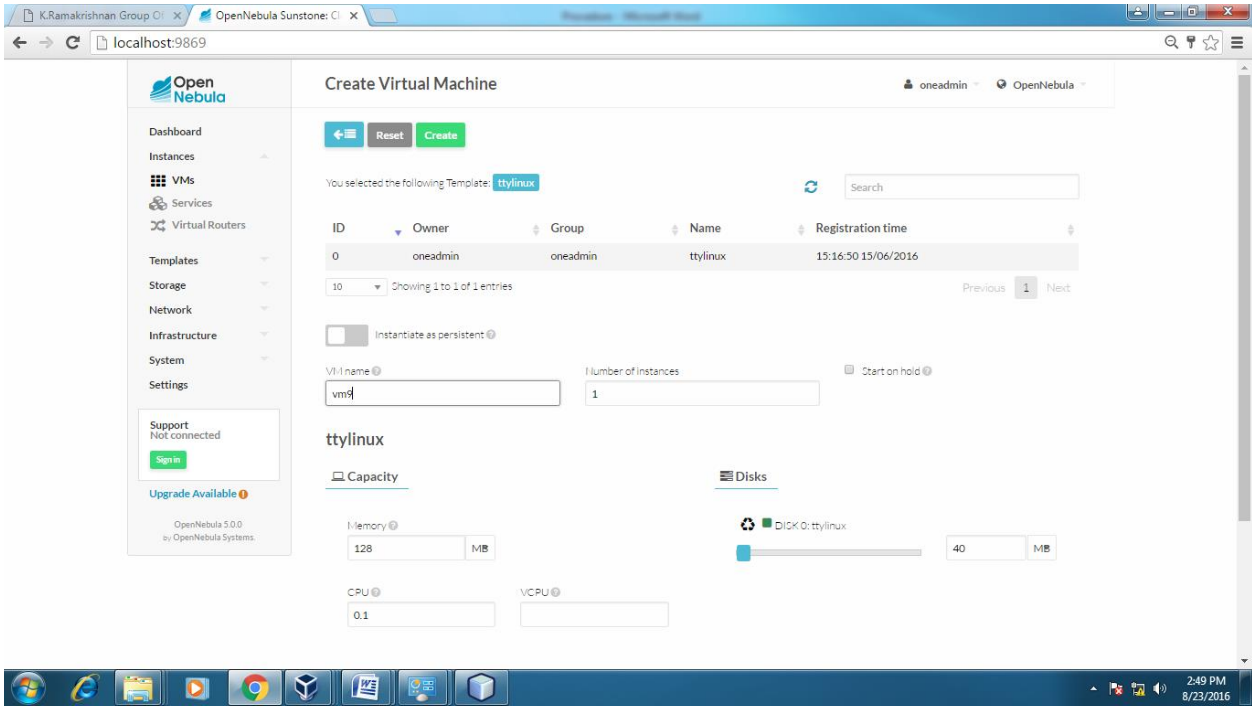
```
CentOS Linux 7 (Core)
Kernel 3.10.0-327.el7.x86_64 on an x86_64

one-sandbox login: _
```



Username: oneadmin

Password: ••••••••••

OpenNebula 5.0.0 by OpenNebula Systems.



Create Virtual Machine

You selected the following Template: ttylinux

ID	Owner	Group	Name	Registration time
0	oneadmin	oneadmin	ttylinux	15:16:50 15/06/2016

10 Showing 1 to 1 of 1 entries

Instantiate as persistent

VM name: vm9

Number of instances: 1

□ Start on hold

ttylinux

Memory: 128 MB

DISK 0: ttylinux 40 MB

CPU: 0.1

VCPU:

APPLICATIONS:

There are various applications of cloud computing in today’s network world. Many search engines and social websites are using the concept of cloud computing like www.amazon.com, hotmail.com, facebook.com, linkedln.com etc. the advantages of cloud computing in context to scalability is like reduced risk , low cost testing ,ability to segment the customer base and auto-scaling based on application load.

RESULT:

Thus the procedure to run the virtual machine of different configuration.

EX.NO.:2

DATE:

Install a C compiler in the virtual machine created using virtual box and execute Simple Programs

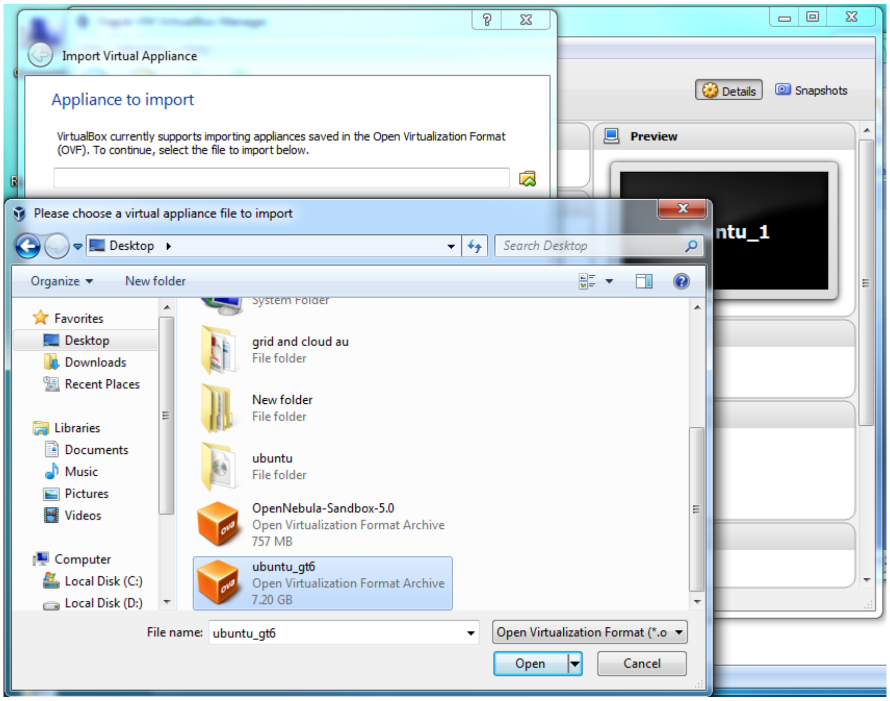
Aim:

To Install a C compiler in the virtual machine created using virtual box and execute Simple Programs`

PROCEDURE:

Steps to import .ova file:

1. Open Virtual box
2. File →import Appliance
3. Browse ubuntu_gt6.ova file
4. Then go to setting, select Usb and choose USB 1.1
5. Then Start the ubuntu_gt6
6. Login using username: dinesh, password:99425.



Import Virtual Appliance

Appliance to import

VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

Please choose a virtual appliance file to import

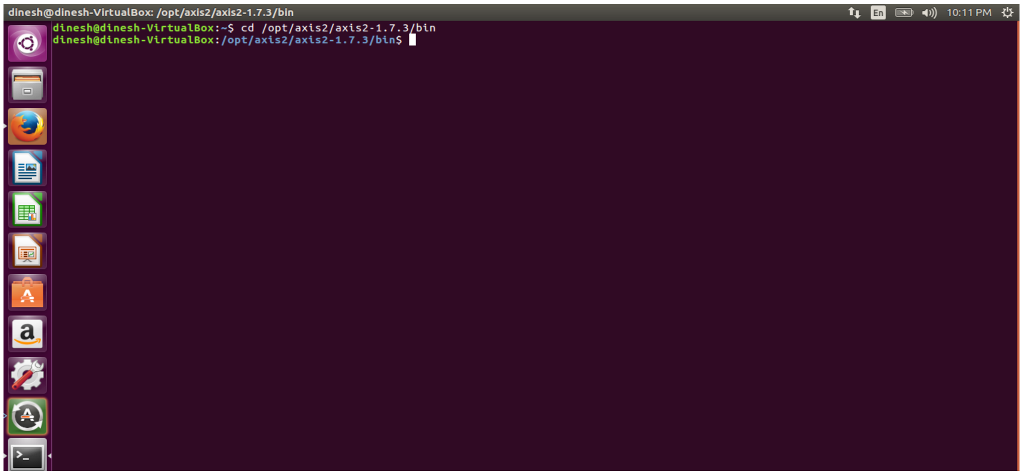
File name: ubuntu_gt6

Open Virtualization Format (*.o

Steps to run c program:

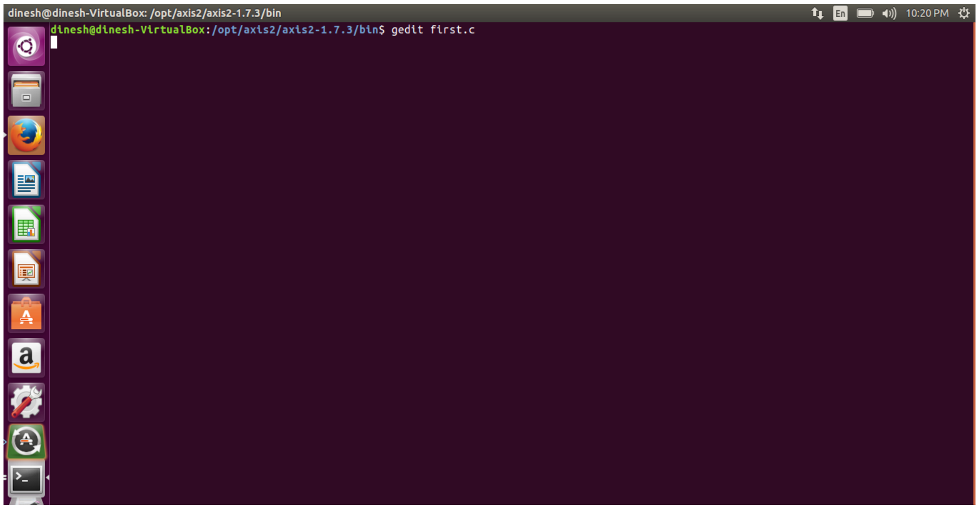
1. Open the terminal
2. Type cd /opt/axis2/axis2-1.7.3/bin then press enter
3. gedit hello.c
4. gcc hello.c
5. ./a.out

1. Type cd /opt/axis2/axis2-1.7.3/bin then press enter



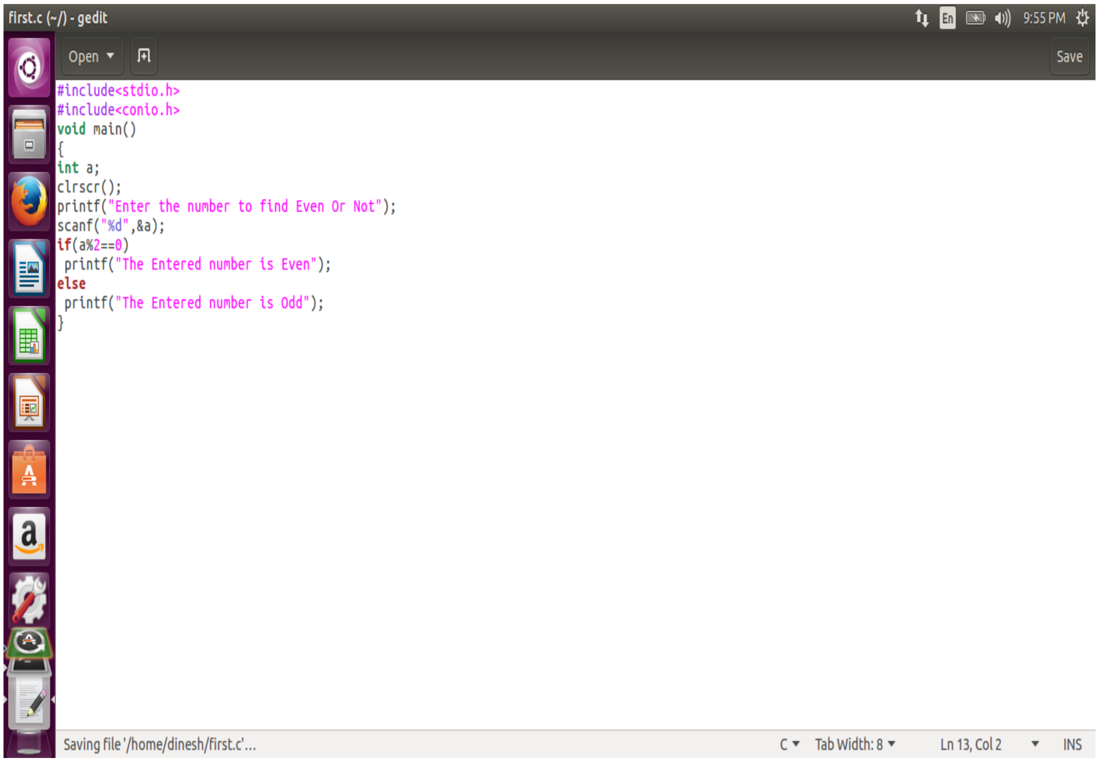
```
dinesh@dinesh-VirtualBox:~$ cd /opt/axis2/axis2-1.7.3/bin
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ 
```

2. Type gedit first.c



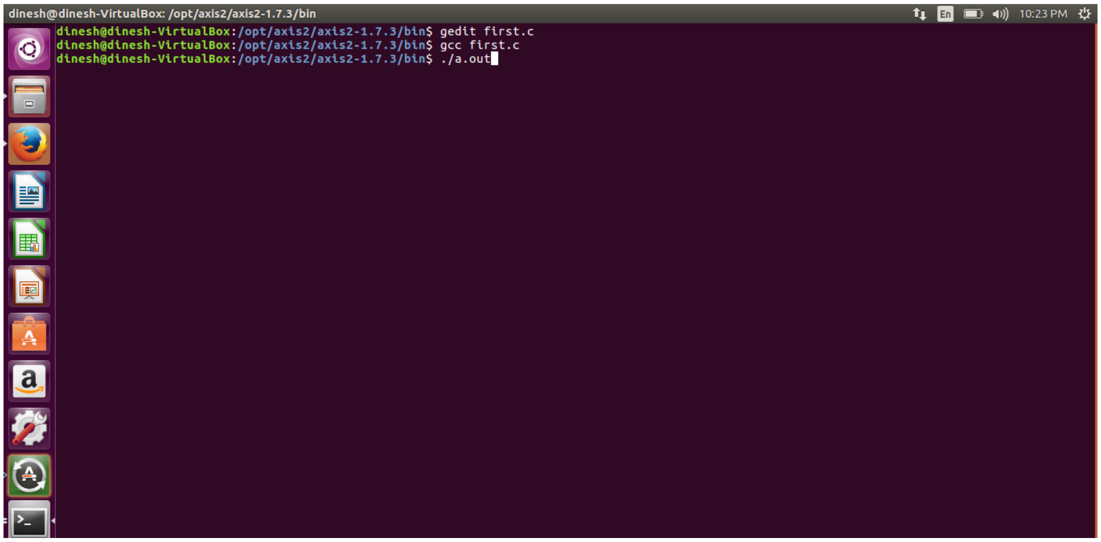
```
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ gedit first.c
```

3. Type the c program



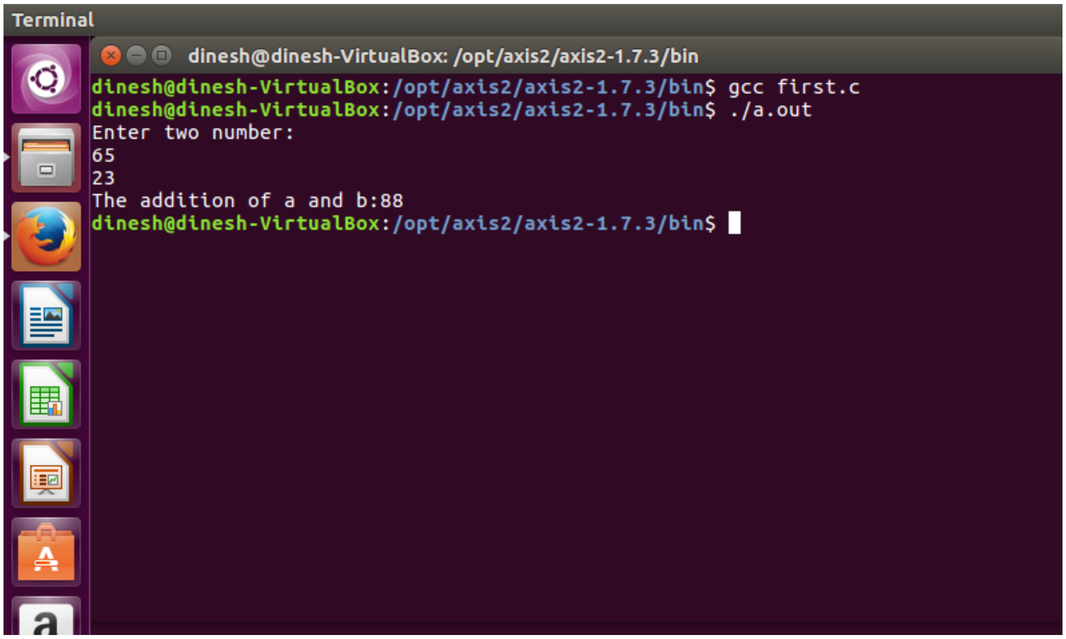
```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
clrscr();
printf("Enter the number to find Even Or Not");
scanf("%d",&a);
if(a%2==0)
 printf("The Entered number is Even");
else
 printf("The Entered number is Odd");
}
```

4. Running the C program



```
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ gedit first.c
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ gcc first.c
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ ./a.out
```

5. **Display the output:**



```
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ gcc first.c
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ ./a.out
Enter two number:
65
23
The addition of a and b:88
dinesh@dinesh-VirtualBox:/opt/axis2/axis2-1.7.3/bin$ 
```

APPLICATIONS:

Simply running all programs in grid environment.

RESULT:

Thus the simple C programs executed successfully.

EX NO.:3
DATE:

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

Aim:

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

Procedure:

1. Install Google Plugin for Eclipse

Read this guide – how to install Google Plugin for Eclipse. If you install the Google App Engine Java SDK together with "**Google Plugin for Eclipse**", then go to step 2, Otherwise, get the Google App Engine Java SDK and extract it.

2. Create New Web Application Project

In Eclipse toolbar, click on the Google icon, and select "**New Web Application Project…**"

Figure – New Web Application Project

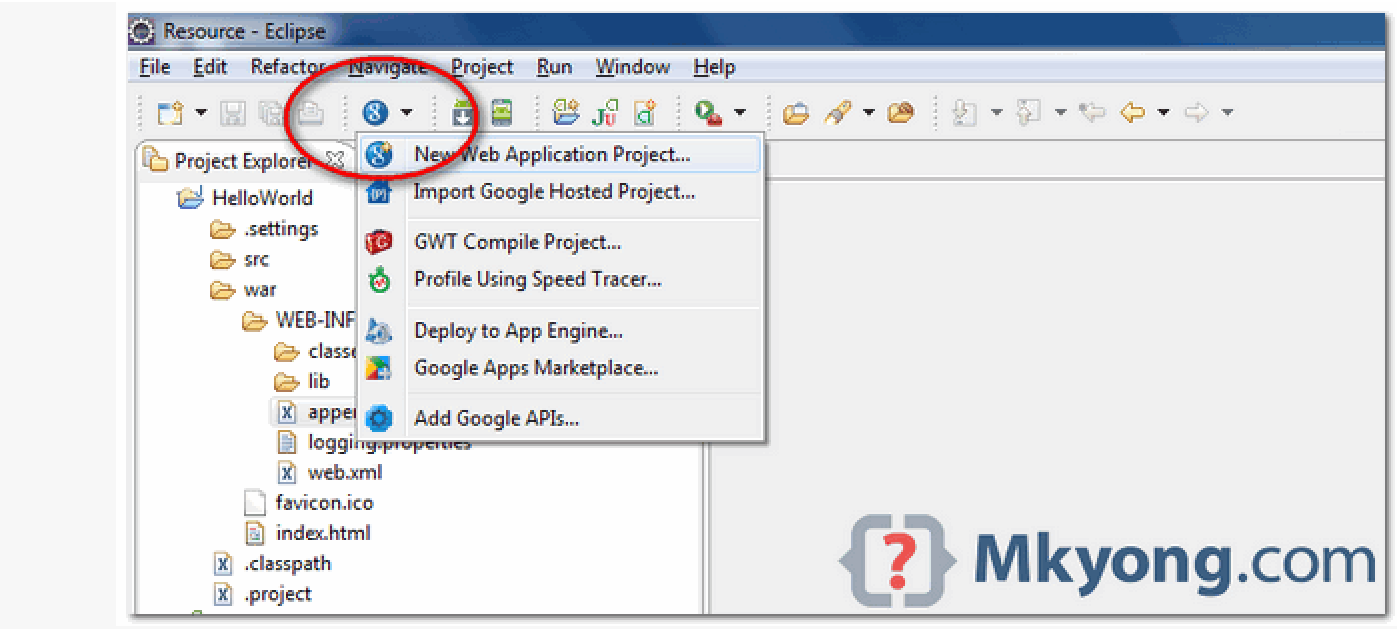
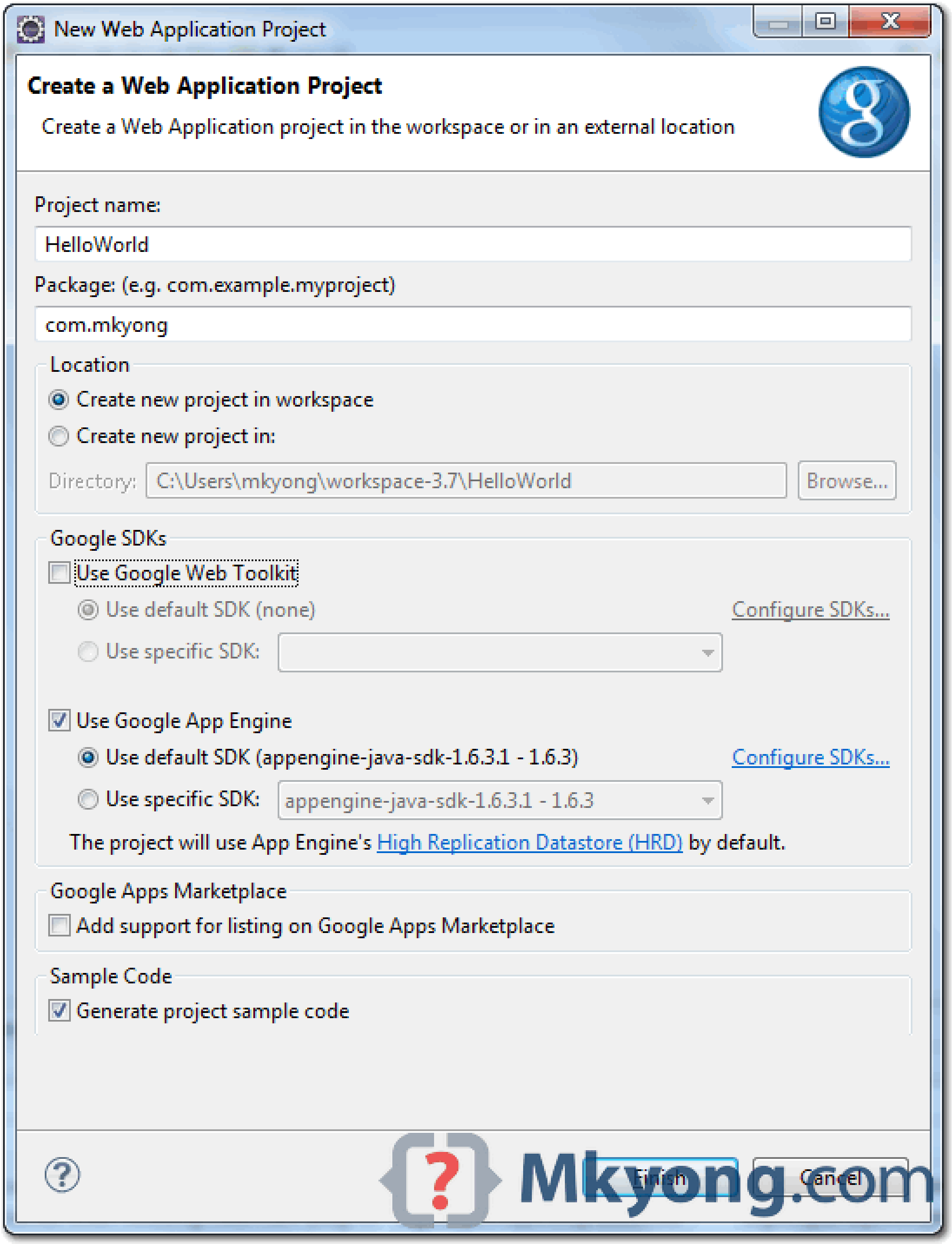


*Figure – Deselect the "**Google Web ToolKit**", and link your GAE Java SDK via the "**configure SDK**" link.*



New Web Application Project

Create a Web Application Project

Create a Web Application project in the workspace or in an external location

Project name:

HelloWorld

Package: (e.g. com.example.myproject)

com.mkyong

Location

● Create new project in workspace

○ Create new project in:

Directory: C:\Users\mkyong\workspace-3.7\HelloWorld

Google SDKs

□ Use Google Web Toolkit

● Use default SDK (none)

○ Use specific SDK:

☑ Use Google App Engine

● Use default SDK (appengine-java-sdk-1.6.3.1 - 1.6.3)

○ Use specific SDK: appengine-java-sdk-1.6.3.1 - 1.6.3

The project will use App Engine's High Replication Datastore (HRD) by default.

Google Apps Marketplace

□ Add support for listing on Google Apps Marketplace

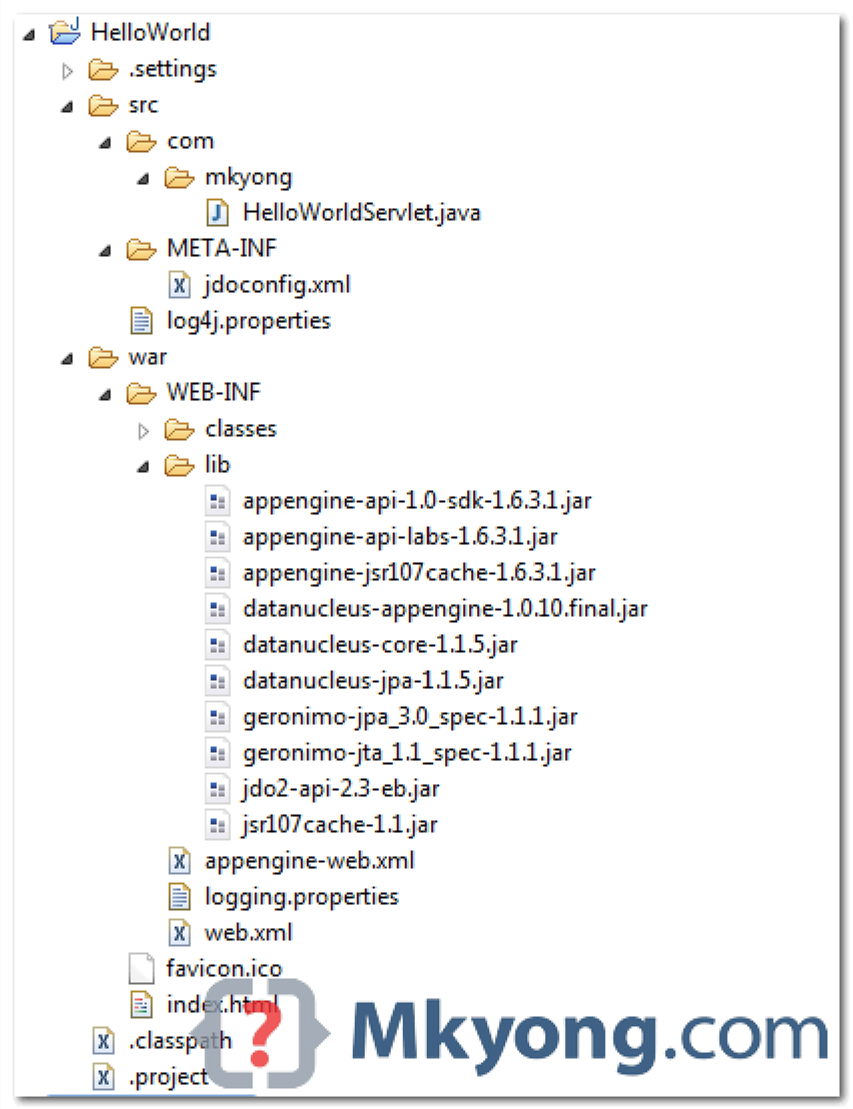
Sample Code

☑ Generate project sample code

Click finished, Google Plugin for Eclipse will generate a sample project automatically.

3. Hello World

Review the generated project directory.



```
HelloWorld
  .settings
  src
    com
      mkyong
        HelloWorldServlet.java
    META-INF
      jdoconfig.xml
    log4j.properties
  war
    WEB-INF
      classes
      lib
        appengine-api-1.0-sdk-1.6.3.1.jar
        appengine-api-labs-1.6.3.1.jar
        appengine-jsr107cache-1.6.3.1.jar
        datanucleus-appengine-1.0.10.final.jar
        datanucleus-core-1.1.5.jar
        datanucleus-jpa-1.1.5.jar
        geronimo-jpa_3.0_spec-1.1.1.jar
        geronimo-jta_1.1_spec-1.1.1.jar
        jdo2-api-2.3-eb.jar
        jsr107cache-1.1.jar
      appengine-web.xml
      logging.properties
      web.xml
    favicon.ico
    index.html
  .classpath
  .project
```

Nothing special, a standard Java web project structure.

```
HelloWorld/
  src/
    ...Java source code...
    META-INF/
      ...other configuration...
  war/
    ...JSPs, images, data files...
    WEB-INF/
      ...app configuration...
      lib/
        ...JARs for libraries...
      classes/
        ...compiled classes...
```

Copy

The extra is this file “appengine-web.xml“, Google App Engine need this to run and deploy the application.

File : appengine-web.xml

```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application></application>
  <version>1</version>

  <!-- Configure java.util.logging -->
  <system-properties>
    <property name="java.util.logging.config.file" value="WEB-INF/logging.properties"/>
  </system-properties>

</appengine-web-app>
```

Copy

4. Run it local

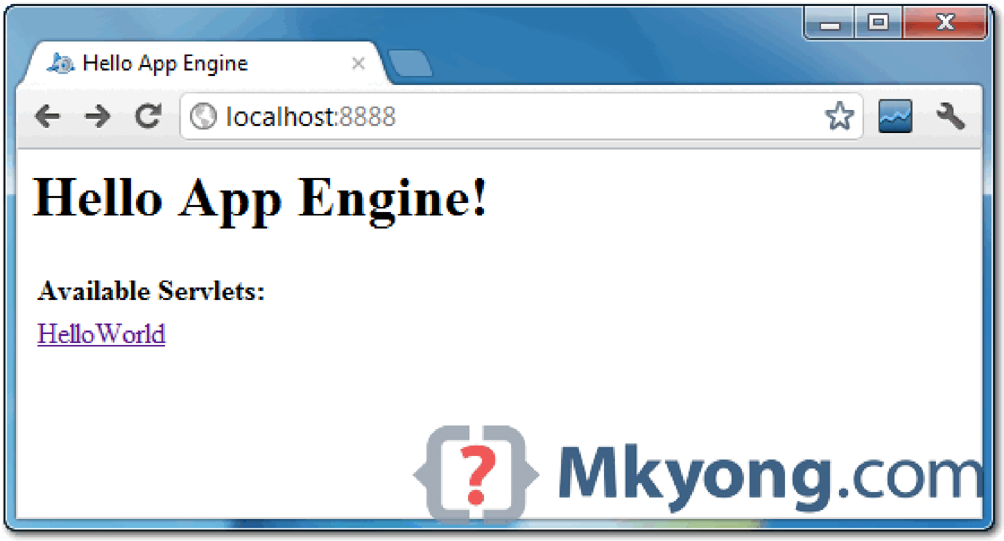
Right click on the project and run as “**Web Application**“.

Eclipse console :

```
//...
INFO: The server is running at http://localhost:8888/
30 Mac 2012 11:13:01 PM com.google.appengine.tools.development.DevAppServerImpl start
INFO: The admin console is running at http://localhost:8888/_ah/admin
```

Copy

Access URL http://localhost:8888/, see output

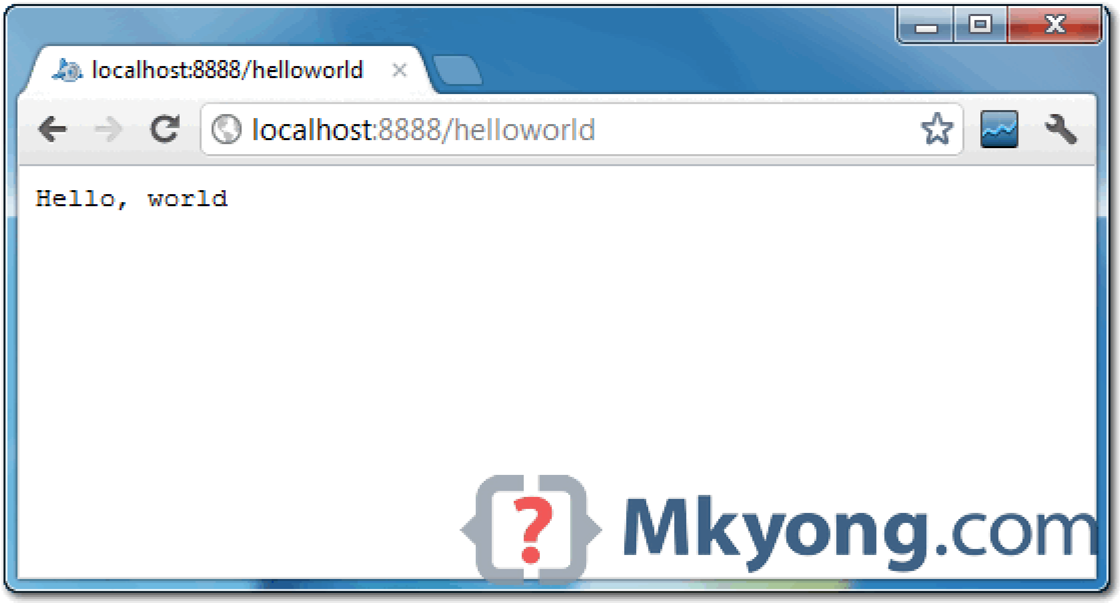


Hello App Engine!

Available Servlets:

HelloWorld

and also the hello world servlet – http://localhost:8888/helloworld



Hello, world

5. Deploy to Google App Engine

Register an account on https://appengine.google.com/, and create an application ID for your web application.

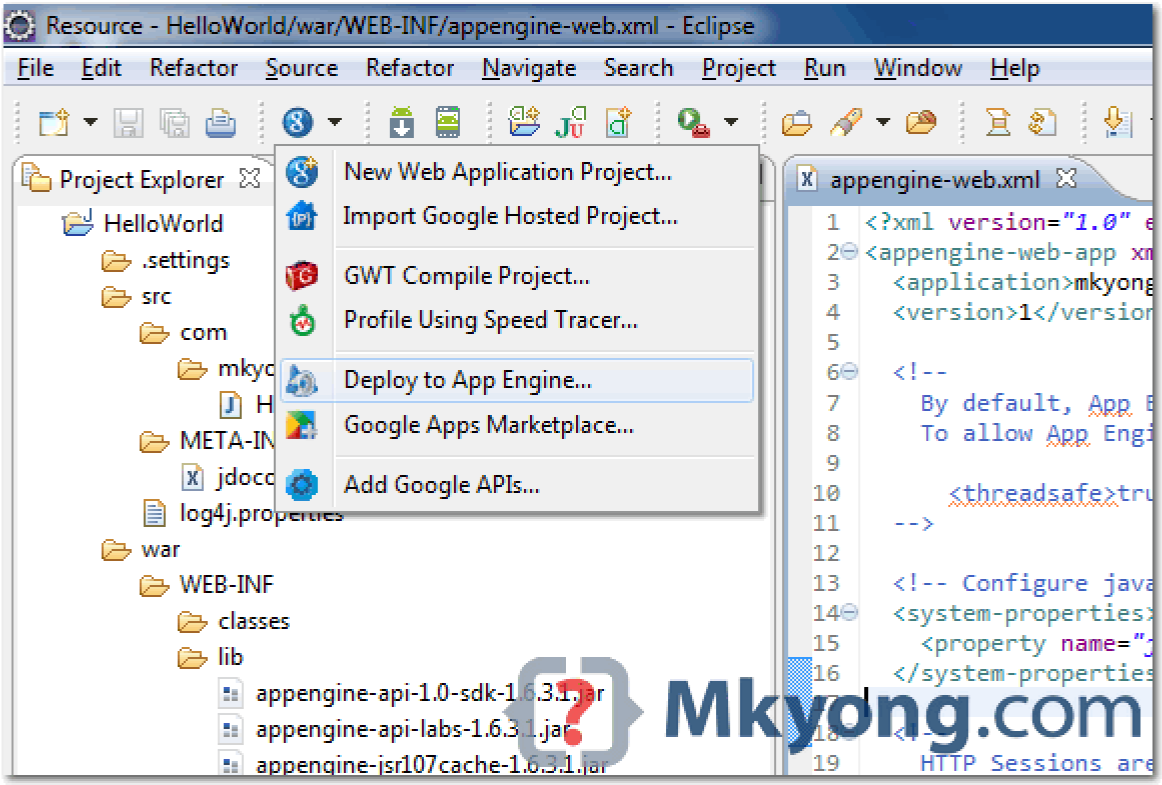
In this demonstration, I created an application ID, named “mkyong123”, and put it in appengine-web.xml.

File : appengine-web.xml

```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application>mkyong123</application>
  <version>1</version>

  <!-- Configure java.util.logging -->
  <system-properties>
    <property name="java.util.logging.config.file" value="WEB-INF/logging.properties"/>
  </system-properties>

</appengine-web-app>
```
Copy

To deploy, see following steps:

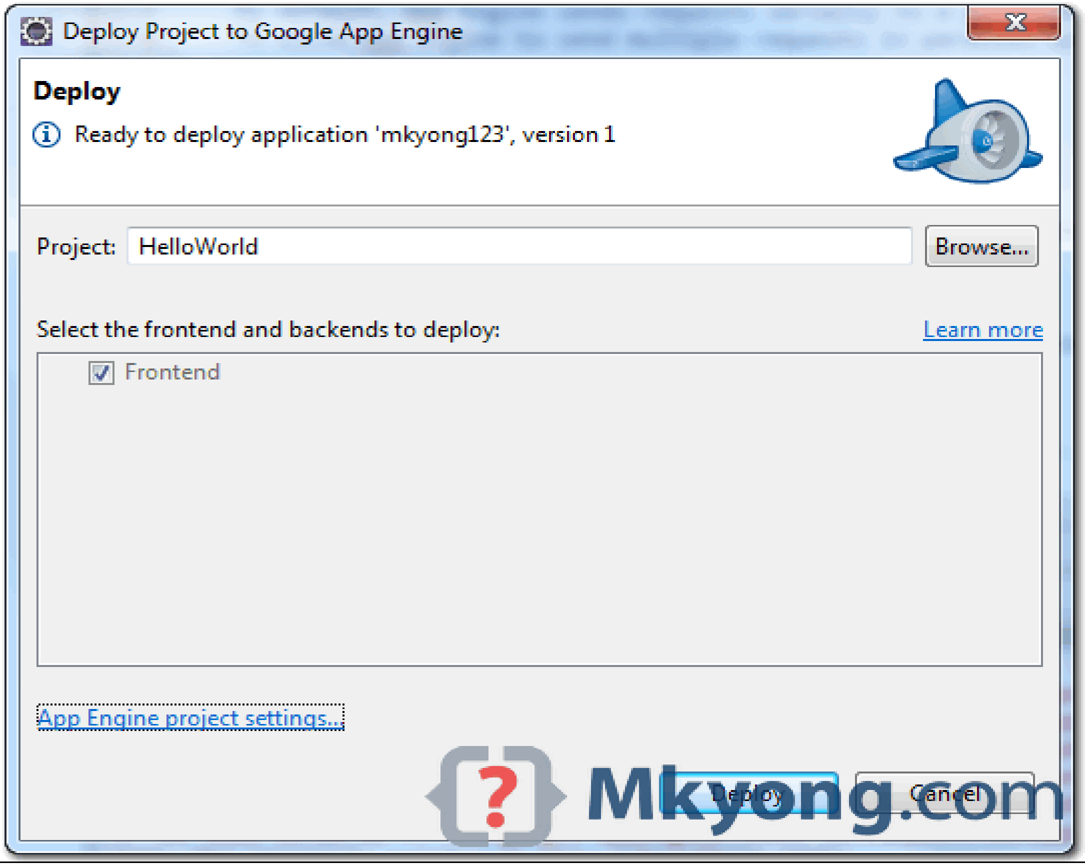
Figure 1.1 – Click on GAE deploy button on the toolbar.



Resource - HelloWorld/war/WEB-INF/appengine-web.xml - Eclipse

```
<?xml version="1.0" e
<appengine-web-app xm
  <application>mkyong
  <version>1</version

  <!--
    By default, App E
    To allow App Engi

      <threadsafe>tru
  -->

  <!-- Configure java
  <system-properties>
    <property name="j
  </system-properties

  <!--
    HTTP Sessions are
```

Figure 1.2 – Sign in with your Google account and click on the Deploy button.



Deploy Project to Google App Engine

Deploy

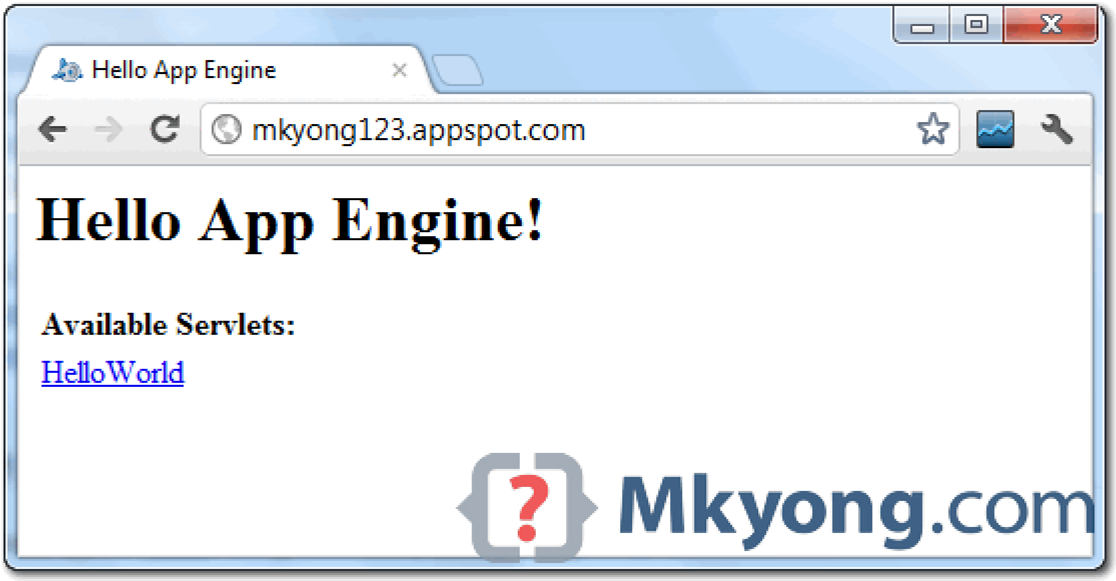
ⓘ Ready to deploy application 'mkyong123', version 1

Project: HelloWorld

Select the frontend and backends to deploy:

☑ Frontend

Figure 1.3 – If everything is fine, the hello world web application will be deployed to this URL – http://mkyong123.appspot.com/



Hello App Engine!

Available Servlets:

HelloWorld

Result:

Thus the simple application was created successfully.

EX. NO.:4
DATE:

Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

Aim:

To Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

Steps:

How to use CloudSim in Eclipse

CloudSim is written in Java. The knowledge you need to use CloudSim is basic Java programming and some basics about cloud computing. Knowledge of programming IDEs such as Eclipse or NetBeans is also helpful. It is a library and, hence, CloudSim does not have to be installed. Normally, you can unpack the downloaded package in any directory, add it to the Java classpath and it is ready to be used. Please verify whether Java is available on your system.

To use CloudSim in Eclipse:

1. Download CloudSim installable files from *https://code.google.com/p/cloudsim/downloads/list and unzip*
2. Open Eclipse
3. Create a new Java Project: File -> New
4. Import an unpacked CloudSim project into the new Java Project

The first step is to initialise the CloudSim package by initialising the CloudSim library, as follows

CloudSim.init(num_user, calendar, trace_flag)

5. 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 its price:

Datacenter datacenter9883 = new Datacenter(name, characteristics, new VmAllocationPolicySimple(hostList), s

6. The third step is to create a broker:

DatacenterBroker broker = createBroker();

7. 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 for cloudlets:

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

8. Submit the VM list to the broker:

```
broker.submitVmList(vmlist)
```

9. Create a cloudlet with length, file size, output size, and utilisation model:

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

10. Submit the cloudlet list to the broker:

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

Sample Output from the Existing Example:

```
Starting
CloudSimExample1...
Initialising...
Starting CloudSim
version 3.0 Datacenter_0
is starting...
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>null
Broker is
starting...
Entities started.
   : Broker: Cloud Resource List received with 1
   resource(s) 0.0: Broker: Trying to Create VM #0
   in Datacenter_0
    : 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
   : 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 ==========
Cloudlet ID  STATUS  Data center ID   VM ID   Time   Start Time
     Finish Time 0         SUCCESS   2                    0              400
                 0.1            400.1
*****Datacenter:
Datacenter_0***** User id
              Debt
    3         35.6

CloudSimExample1 finished!
```

RESULT:

The simulation was successfully executed.

EX.NO.:5
DATE:

Use GAE launcher to launch the web applications.

Aim:

To Use GAE launcher to launch the web applications.

Steps:

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:

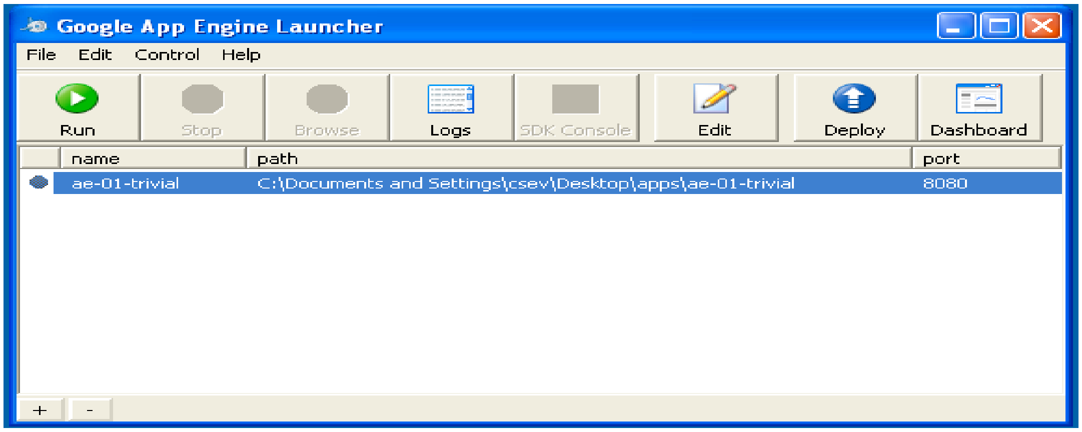
```
application: ae-01-trivial
    version: 1
runtime: python api_version: 1
handlers:- url: /.*
script: index.py
```

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'
```

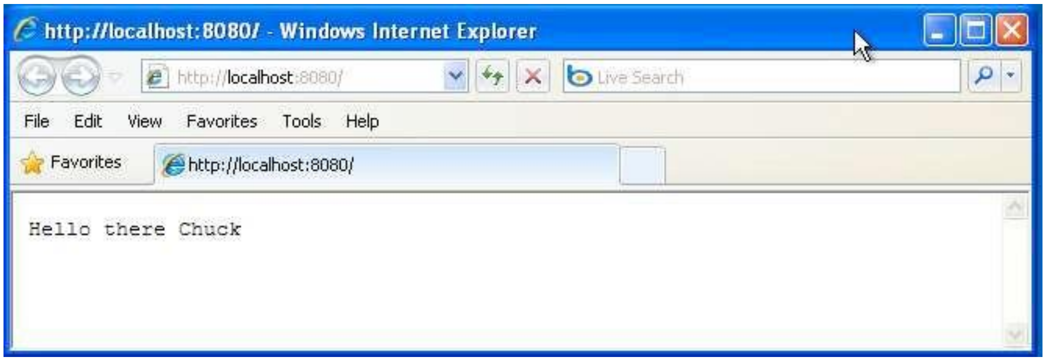
Then start the GoogleAppEngineLauncher program that can be found under Applications. Use the File --> Add Existing Application command and navigate into the apps directory and select the ae--01--trivial folder. Once you have added the application, select it so that you can control the application using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



name	path	port
ae-01-trivial	C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial	8080

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:


http://localhost:8080/ - Windows Internet Explorer
http://localhost:8080/
Live Search
File Edit View Favorites Tools Help
Favorites
http://localhost:8080/

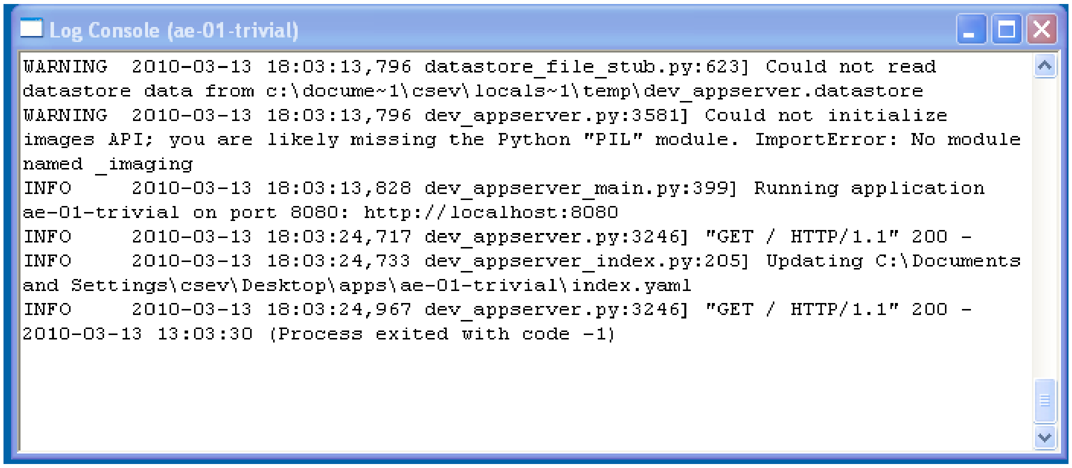
```
Hello there Chuck
```

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:

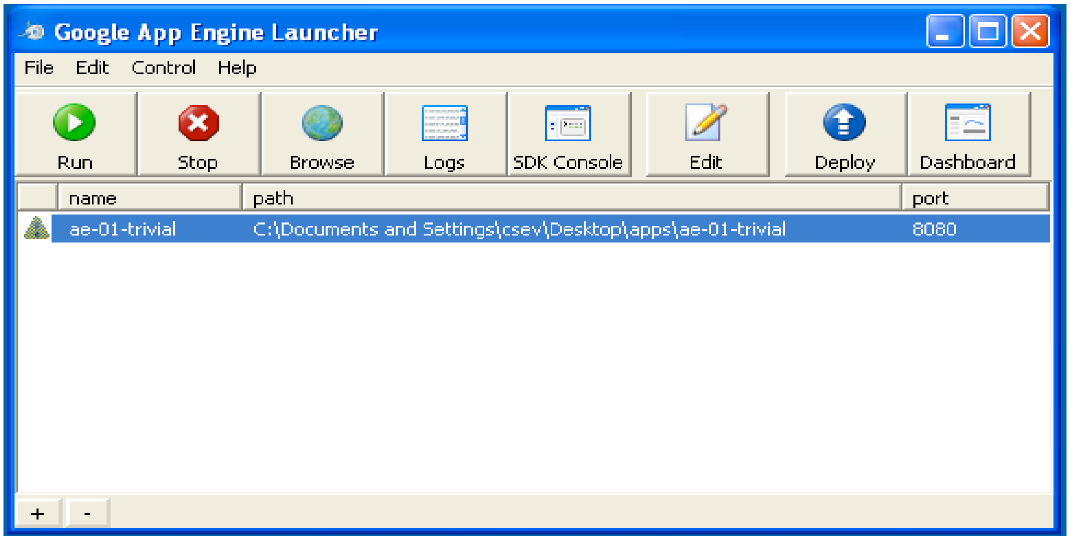
Each time you press Refresh in your browser – you can see it retrieving the output with a GET request.



```
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)
```

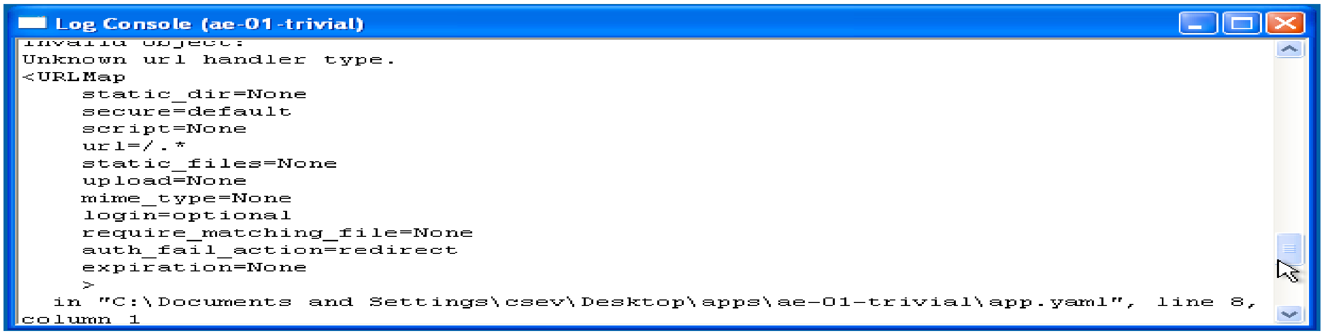
Dealing With Errors

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:



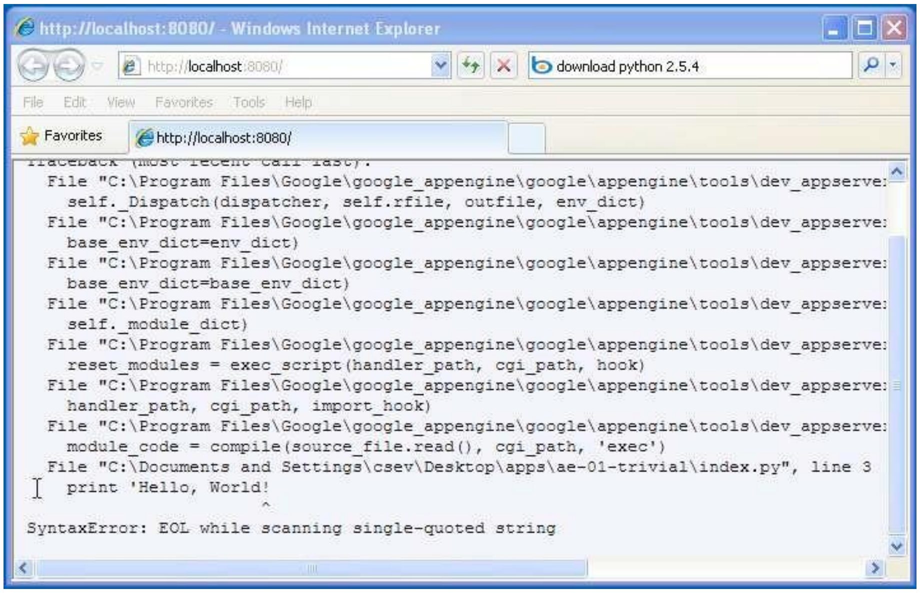
	name	path	port
⚠	ae-01-trivial	C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial	8080

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



```
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.



http://localhost:8080/ - Windows Internet Explorer
http://localhost:8080/
download python 2.5.4
File Edit View Favorites Tools Help
Favorites
http://localhost:8080/

```
Traceback (most recent call last):
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    self._Dispatch(dispatcher, self.rfile, outfile, env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    base_env_dict=env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    base_env_dict=base_env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    self._module_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    reset_modules = exec_script(handler_path, cgi_path, hook)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    handler_path, cgi_path, import_hook)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
    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 the 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 web applications was created.

EX.NO:6
DATE:

Find a procedure to transfer the files from one virtual machine to another virtual machine.

Aim:

To Find a procedure to transfer the files from one virtual machine to another virtual machine.

Steps:

1. You can copy few (or more) lines with *copy & paste* mechanism.
   For this you need to share clipboard between host OS and guest OS, installing Guest Addition on both the virtual machines (probably setting *bidirectional* and restarting them). You *copy* from *guest OS* in the clipboard that is shared with the *host OS*.
   Then you *paste* from the *host OS* to the second *guest OS*.
2. You can enable drag and drop too with the same method (Click on the machine, settings, general, advanced, drag and drop: set to *bidirectional*)
3. You can have common *Shared Folders* on both virtual machines and use one of the directory shared as buffer to copy.
   Installing Guest Additions you have the possibility to set Shared Folders too. As you put a file in a shared folder from *host OS* or from *guest OS*, is immediately visible to the other. (Keep in mind that can arise some problems for date/time of the files when there are different clock settings on the different virtual machines).
   If you use the same folder shared on more machines you can exchange files directly copying them in this folder.
4. You can use usual method to copy files between 2 different computer with client-server application. (e.g. scp with sshd active for linux, winscp... you can get some info about SSH servers e.g. here)
   You need an active server (sshd) on the receiving machine and a client on the sending machine. Of course you need to have the authorization setted (via password or, better, via an automatic authentication method).
   Note: many Linux/Ubuntu distribution install sshd by default: you can see if it is running with pgrep sshd from a shell. You can install with sudo apt-get install openssh-server.
5. You can mount part of the file system of a virtual machine via NFS or SSHFS on the other, or you can share file and directory with Samba.
   You may find interesting the article Sharing files between guest and host without VirtualBox shared folders with detailed step by step instructions.

You should remember that you are dialling with a little network of machines with different operative systems, and in particular:

- Each virtual machine has its own operative system running on and acts as a physical machine.
- Each virtual machine is an instance of a program *owned* by an *user* in the hosting operative system and should undergo the restrictions of the *user* in the *hosting OS*.
  E.g Let we say that Hastur and Meow are users of the hosting machine, but they did not allow each other to see their directories (no read/write/execute authorization). When each of them run a virtual machine, for the hosting OS those virtual machine are two normal programs owned by Hastur and Meow and cannot see the private directory of the other user. This is a restriction due to the *hosting OS*. It's easy to overcame it: it's enough to give authorization to read/write/execute to a directory or to chose a different directory in which both users can read/write/execute.
- Windows likes mouse and Linux fingers. :-)
  I mean I suggest you to enable *Drag & drop* to be cosy with the Windows machines and the *Shared folders* or to be cosy with Linux.
  When you will need to be fast with Linux you will feel the need of ssh-keygen and to Generate once SSH Keys to copy files on/from a remote machine without writing password anymore. In this way it functions bash auto-completion remotely too!

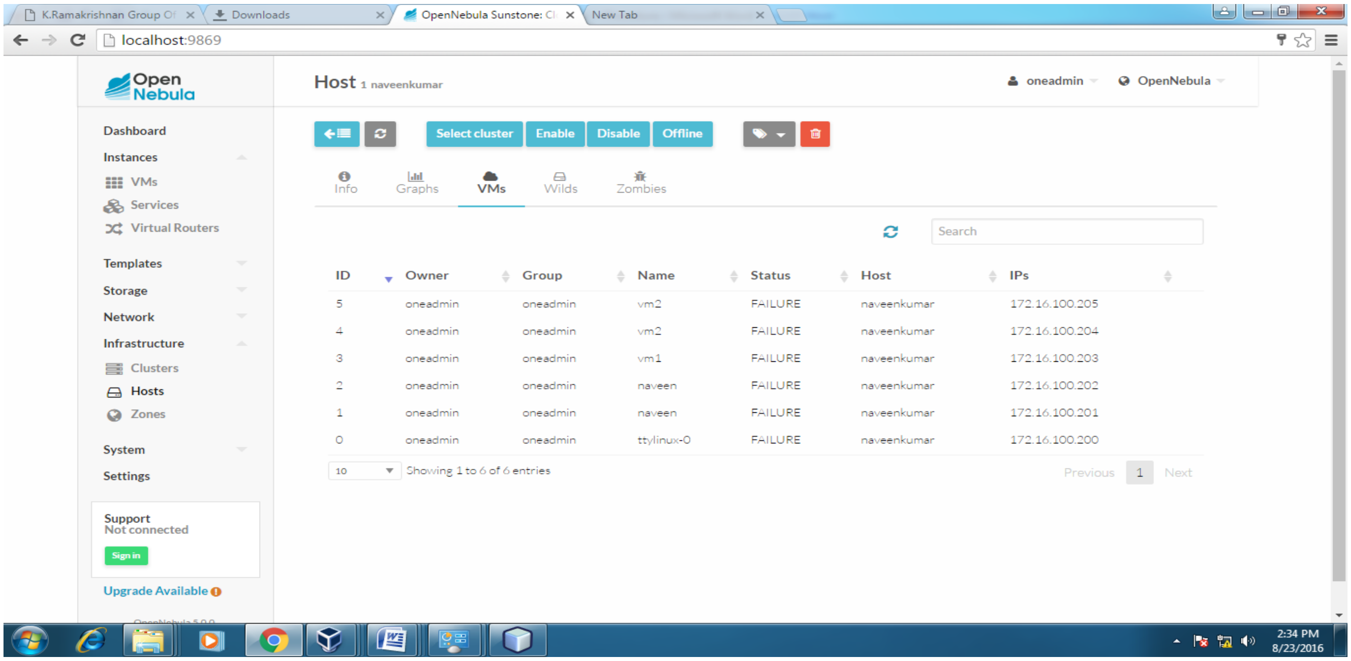
PROCEDURE:

Steps:

1. Open Browser, type localhost:9869
2. Login using username: oneadmin, password: opennebula
3. Then follow the steps to migrate VMs
   a. Click on infrastructure
   b. Select clusters and enter the cluster name
   c. Then select host tab, and select all host
   d. Then select Vnets tab, and select all vnet
   e. Then select datastores tab, and select all datastores
   f. And then choose host under infrastructure tab
   g. Click on + symbol to add new host, name the host then click on create.
4. on instances, select VMs to migrate then follow the stpes
   a. Click on 8th icon ,the drop down list display
   b. Select migrate on that ,the popup window display
   c. On that select the target host to migrate then click on migrate.

Before migration

Host:SACET

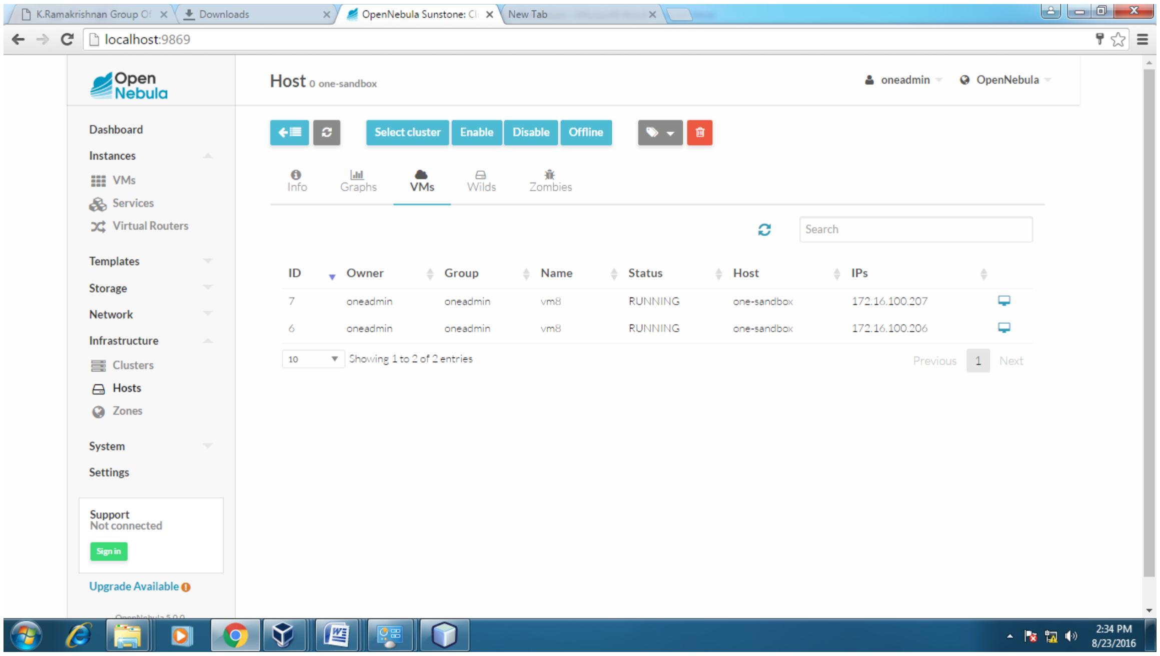


Host 1 naveenkumar

ID	Owner	Group	Name	Status	Host	IPs
5	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.205
4	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.204
3	oneadmin	oneadmin	vm1	FAILURE	naveenkumar	172.16.100.203
2	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.202
1	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.201
0	oneadmin	oneadmin	ttylinux-0	FAILURE	naveenkumar	172.16.100.200

10 Showing 1 to 6 of 6 entries

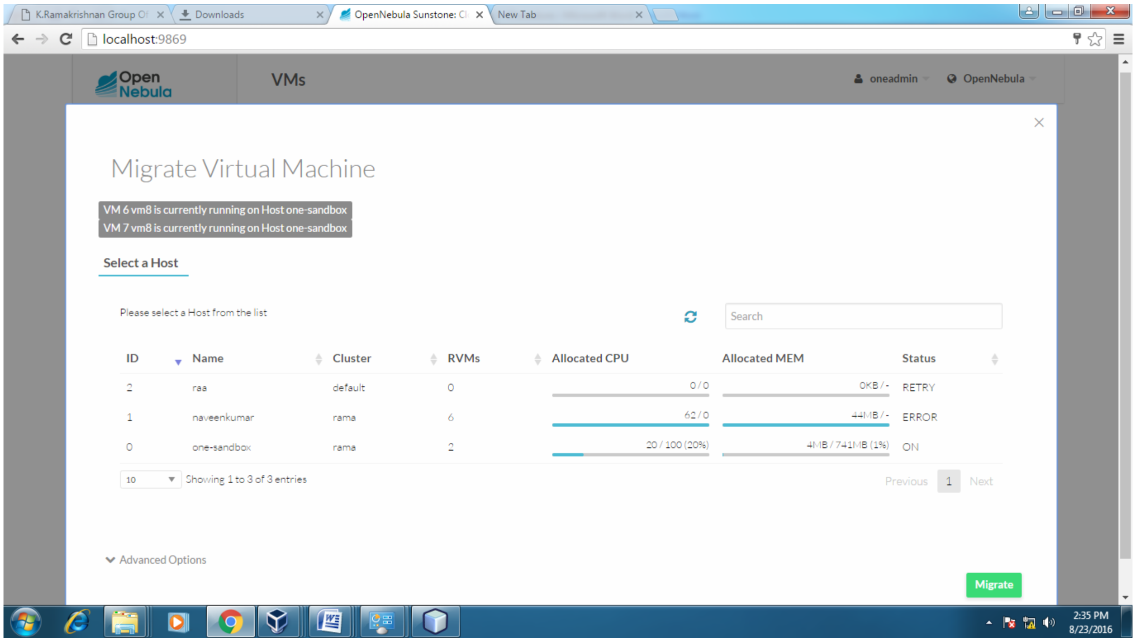
Host:one-sandbox



Host 0 one-sandbox

ID	Owner	Group	Name	Status	Host	IPs
7	oneadmin	oneadmin	vm8	RUNNING	one-sandbox	172.16.100.207
6	oneadmin	oneadmin	vm8	RUNNING	one-sandbox	172.16.100.206

10 Showing 1 to 2 of 2 entries



Migrate Virtual Machine

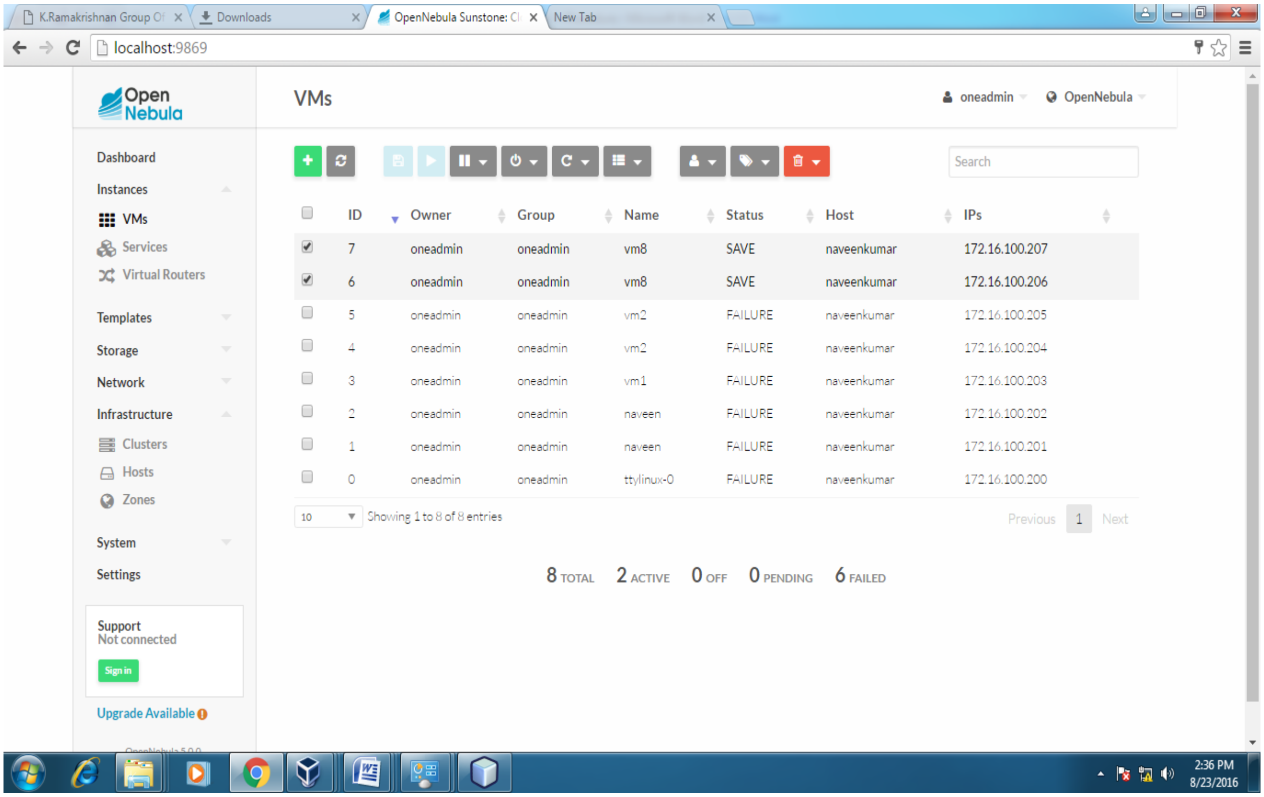
VM 6 vm8 is currently running on Host one-sandbox
VM 7 vm8 is currently running on Host one-sandbox

Select a Host

Please select a Host from the list

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
2	raa	default	0	0 / 0	0KB / -	RETRY
1	naveenkumar	rama	6	62 / 0	44MB / -	ERROR
0	one-sandbox	rama	2	20 / 100 (20%)	4MB / 741MB (1%)	ON

10 Showing 1 to 3 of 3 entries

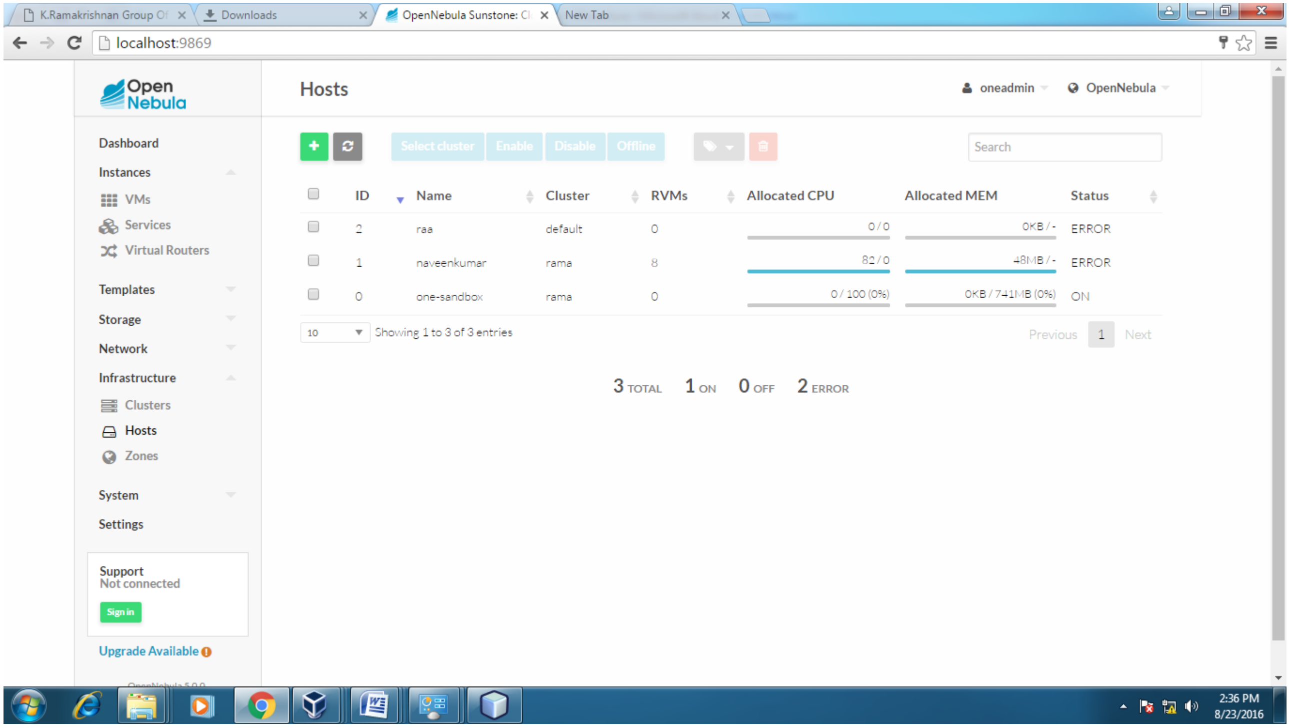


	ID	Owner	Group	Name	Status	Host	IPs
☑	7	oneadmin	oneadmin	vm8	SAVE	naveenkumar	172.16.100.207
☑	6	oneadmin	oneadmin	vm8	SAVE	naveenkumar	172.16.100.206
□	5	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.205
□	4	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.204
□	3	oneadmin	oneadmin	vm1	FAILURE	naveenkumar	172.16.100.203
□	2	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.202
□	1	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.201
□	0	oneadmin	oneadmin	ttylinux-0	FAILURE	naveenkumar	172.16.100.200

10 Showing 1 to 8 of 8 entries

8 TOTAL 2 ACTIVE 0 OFF 0 PENDING 6 FAILED

After Migration:



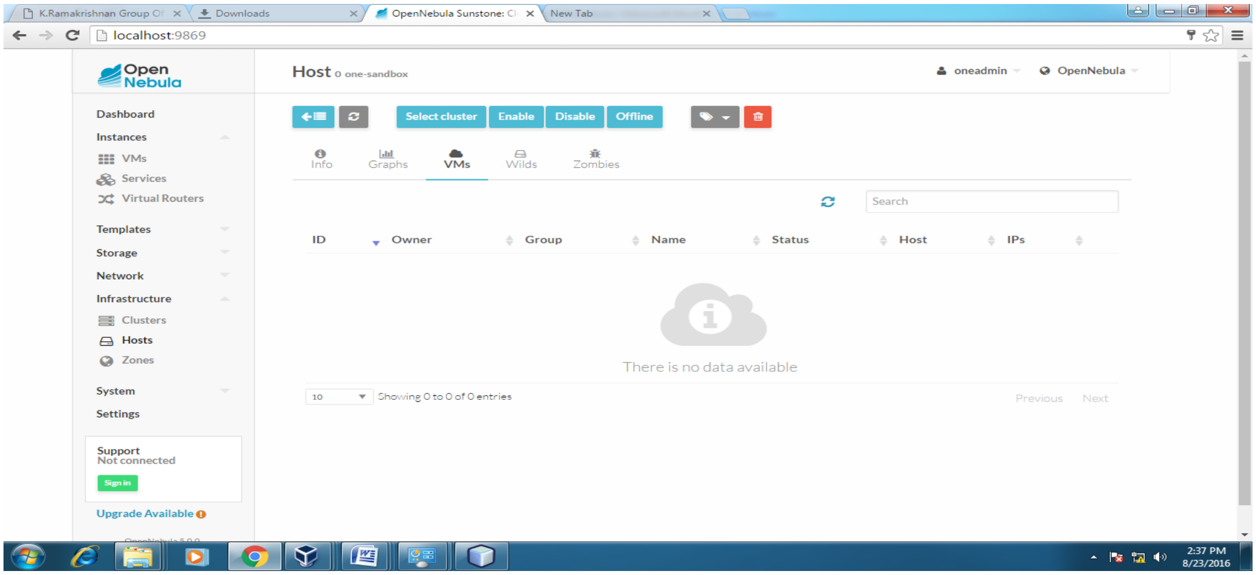
Hosts

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
2	raa	default	0	0 / 0	0KB / -	ERROR
1	naveenkumar	rama	8	82 / 0	48MB / -	ERROR
0	one-sandbox	rama	0	0 / 100 (0%)	0KB / 741MB (0%)	ON

Showing 1 to 3 of 3 entries

3 TOTAL 1 ON 0 OFF 2 ERROR

Host:one-sandbox



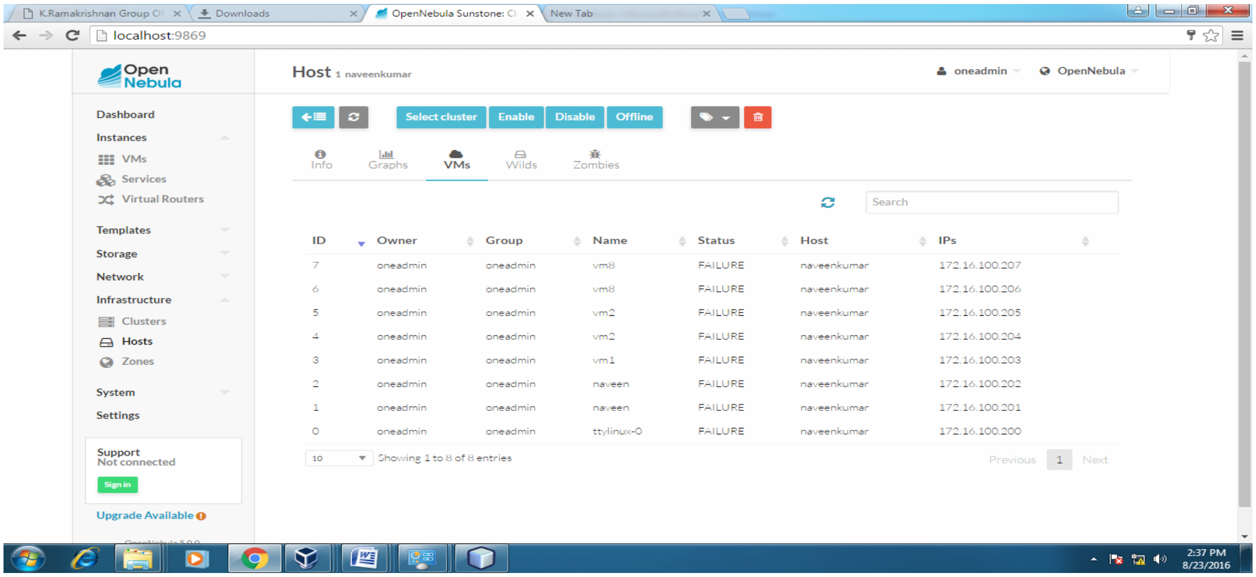
Host 0 one-sandbox

ID	Owner	Group	Name	Status	Host	IPs

There is no data available

Showing 0 to 0 of 0 entries

Host:SACET



Host 1 naveenkumar

ID	Owner	Group	Name	Status	Host	IPs
7	oneadmin	oneadmin	vm8	FAILURE	naveenkumar	172.16.100.207
6	oneadmin	oneadmin	vm8	FAILURE	naveenkumar	172.16.100.206
5	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.205
4	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.204
3	oneadmin	oneadmin	vm1	FAILURE	naveenkumar	172.16.100.203
2	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.202
1	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.201
0	oneadmin	oneadmin	ttylinux-0	FAILURE	naveenkumar	172.16.100.200

10 Showing 1 to 8 of 8 entries

APPLICATIONS:

Easily migrate your virtual machine from one pc to another.

Result:

Thus the file transfer between VM was successfully completed…..

EX.NO:7
DATE:

Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)

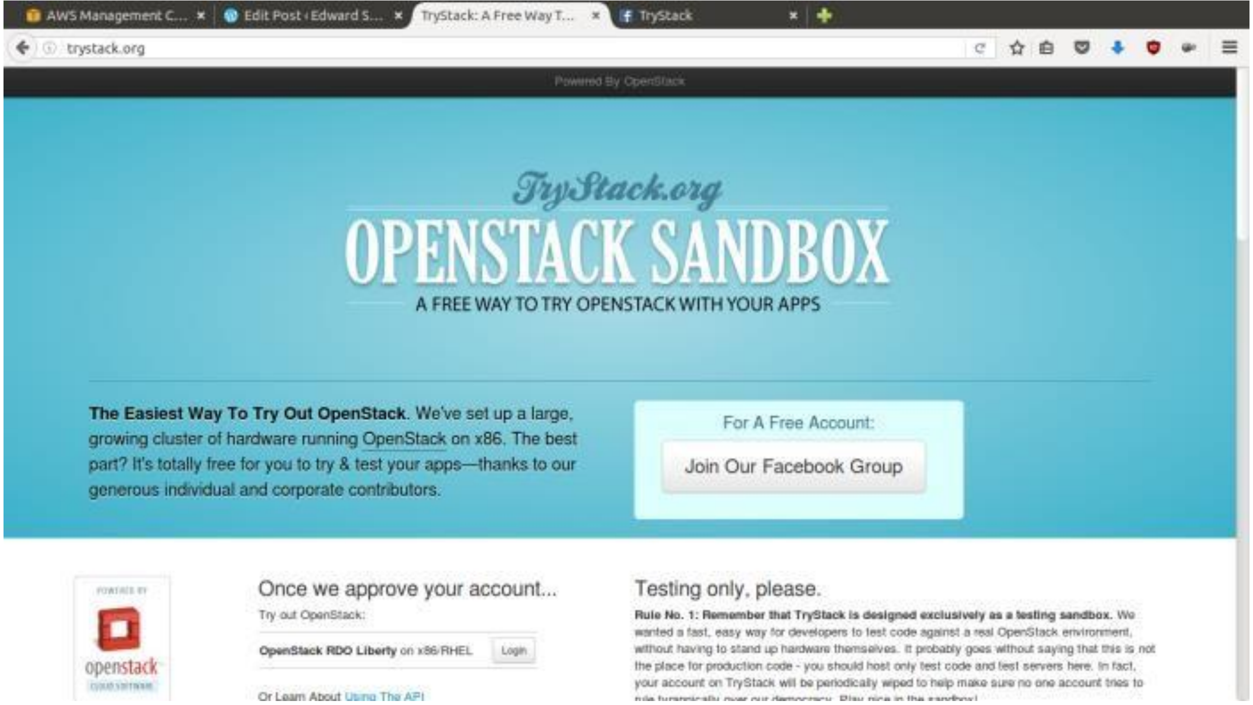
Aim:

To Find a procedure to launch virtual machine using trystack.

Steps:

OpenStack is an open-source software cloud computing platform. OpenStack is primarily used for deploying an infrastructure as a service (IaaS) solution like Amazon Web Service (AWS). In other words, you can *make your own AWS* by using OpenStack. If you want to try out OpenStack, **TryStack** is the easiest and free way to do it.

In order to try OpenStack in TryStack, you must register yourself by joining TryStack Facebook Group. The acceptance of group needs a couple days because it's approved manually. After you have been accepted in the TryStack Group, you can log in TryStack.



The Easiest Way To Try Out OpenStack. We've set up a large, growing cluster of hardware running OpenStack on x86. The best part? It's totally free for you to try & test your apps—thanks to our generous individual and corporate contributors.

Once we approve your account...

Try out OpenStack:

OpenStack RDO Liberty on x86/RHEL

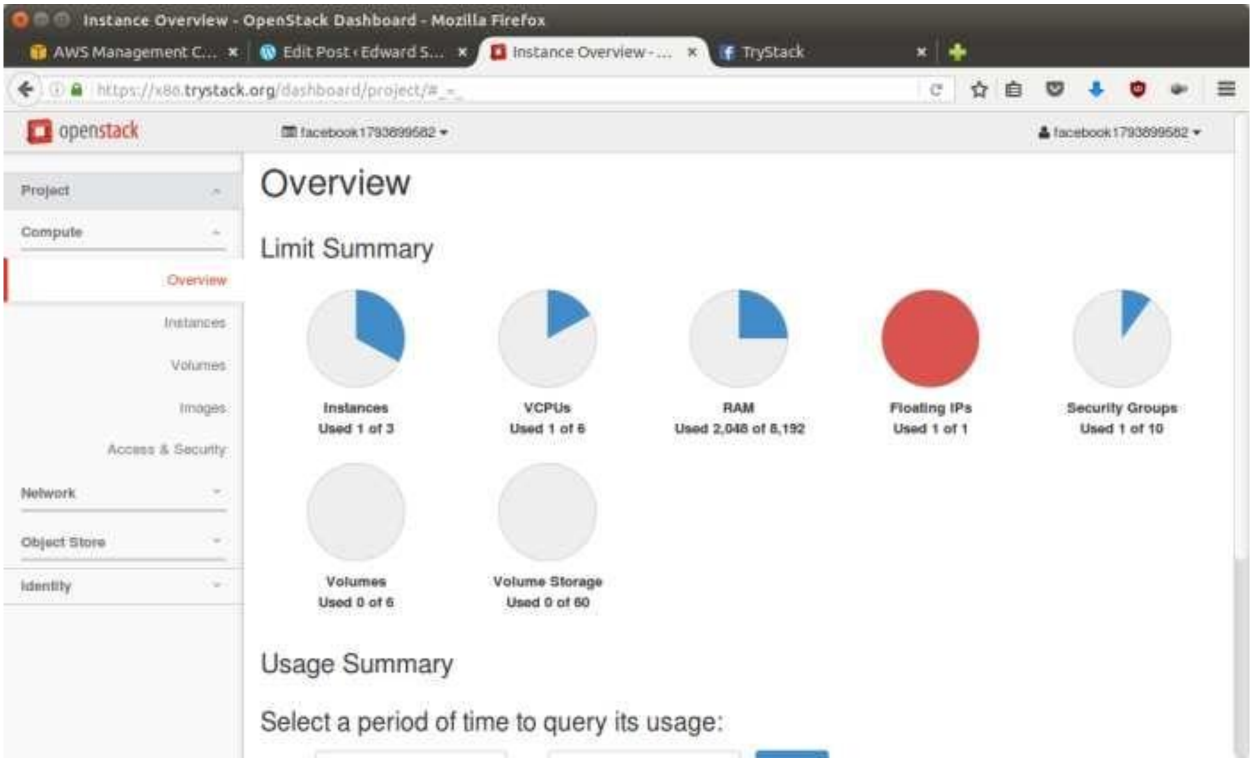
Or Learn About Using The API

Testing only, please.

Rule No. 1: Remember that TryStack is designed exclusively as a testing sandbox. We wanted a fast, easy way for developers to test code against a real OpenStack environment, without having to stand up hardware themselves. It probably goes without saying that this is not the place for production code - you should host only test code and test servers here. In fact, your account on TryStack will be periodically wiped to help make sure no one account tries to rule tyrannically over our democracy. Play nice in the sandbox!

TryStack.org Homepage

I assume that you already join to the Facebook Group and login to the dashboard. After you log in to the TryStack, you will see the Compute Dashboard like:



Overview

Limit Summary

Instances
Used 1 of 3

VCPUs
Used 1 of 6

RAM
Used 2,048 of 8,192

Floating IPs
Used 1 of 1

Security Groups
Used 1 of 10

Volumes
Used 0 of 6

Volume Storage
Used 0 of 60

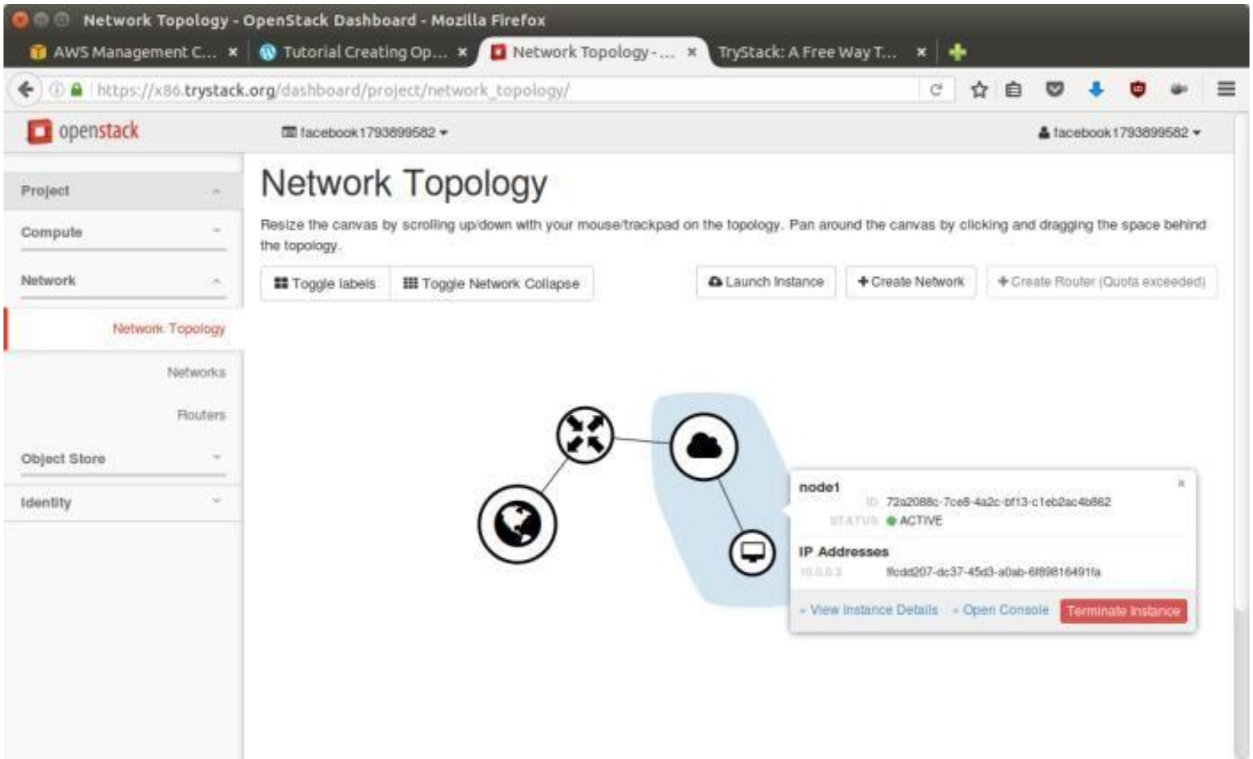
Usage Summary

Select a period of time to query its usage:

OpenStack Compute Dashboard

Overview: What we will do?

In this post, I will show you how to run an OpenStack instance. The instance will be accessible through the internet (have a public IP address). The final topology will like:



Network Topology

Resize the canvas by scrolling up/down with your mouse/trackpad on the topology. Pan around the canvas by clicking and dragging the space behind the topology.

node1
ID 72a2088c-7ce8-4a2c-bf13-c1eb2ac4b862
STATUS ACTIVE
IP Addresses
10.0.0.3 ffcdd207-dc37-45d3-a0ab-6f898164911a

Network topology
As you see from the image above, the instance will be connected to a local network and the local network will be connected to internet.

Step 1: Create Network

Network? Yes, the network in here is our own local network. So, your instances will be not mixed up with the others. You can imagine this as your own LAN (Local Area Network) in the cloud.

1. Go to **Network > Networks** and then click **Create Network**.
2. In **Network** tab, fill **Network Name** for example internal and then click **Next**.
3. In **Subnet** tab,
    1. Fill **Network Address** with appropriate CIDR, for example 192.168.1.0/24. Use private network CIDR block as the best practice.
    2. Select **IP Version** with appropriate IP version, in this case IPv4.
    3. Click **Next**.
4. In **Subnet Details** tab, fill **DNS Name Servers** with 8.8.8.8 (Google DNS) and then click **Create**.

Step 2: Create Instance

Now, we will create an instance. The instance is a virtual machine in the cloud, like AWS EC2. You need the instance to connect to the network that we just created in the previous step.

1. Go to **Compute > Instances** and then click **Launch Instance**.
2. In **Details** tab,
    1. Fill **Instance Name**, for example Ubuntu 1.
    2. Select **Flavor**, for example m1.medium.
    3. Fill **Instance Count** with **1**.
    4. Select **Instance Boot Source** with **Boot from Image**.
    5. Select **Image Name** with **Ubuntu 14.04 amd64 (243.7 MB)** if you want install Ubuntu 14.04 in your virtual machine.
3. In **Access & Security** tab,
    1. Click [+] button of **Key Pair** to import key pair. This key pair is a public and private key that we will use to connect to the instance from our machine.
    2. In **Import Key Pair** dialog,
        1. Fill **Key Pair Name** with your machine name (for example Edward-Key).
        2. Fill **Public Key** with your **SSH public key** (usually is in ~/.ssh/id_rsa.pub). See description in Import Key Pair dialog box for more information. If you are using Windows, you can use **Puttygen** to generate key pair.
        3. Click **Import key pair**.
    3. In **Security Groups**, mark/check **default**.

4. In **Networking** tab,
    1. In **Selected Networks**, select network that have been created in Step 1, for example internal.
5. Click **Launch**.
6. If you want to create multiple instances, you can repeat step 1-5. I created one more instance with instance name Ubuntu 2.

Step 3: Create Router

I guess you already know what router is. In the step 1, we created our network, but it is isolated. It doesn't connect to the internet. To make our network has an internet connection, we need a router that running as the gateway to the internet.

1. Go to **Network > Routers** and then click **Create Router**.
2. Fill **Router Name** for example router1 and then click **Create router**.
3. Click on your **router name link**, for example router1, **Router Details** page.
4. Click **Set Gateway** button in upper right:
    1. Select **External networks** with **external**.
    2. Then **OK**.
5. Click **Add Interface** button.
    1. Select **Subnet** with the network that you have been created in Step 1.
    2. Click **Add interface**.
6. Go to **Network > Network Topology**. You will see the network topology. In the example, there are two network, i.e. external and internal, those are bridged by a router. There are
instances those are joined to internal network.

Step 4: Configure Floating IP Address

Floating IP address is public IP address. It makes your instance is accessible from the internet. When you launch your instance, the instance will have a private network IP, but no public IP. In OpenStack, the public Ips is collected in a pool and managed by admin (in our case is TryStack). You need to request a public (floating) IP address to be assigned to your instance.

1. Go to **Compute > Instance**.
2. In one of your instances, click **More > Associate Floating IP**.
3. In **IP Address**, click Plus [+].
4. Select **Pool** to **external** and then click **Allocate IP**.
5. Click **Associate**.
6. Now you will get a public IP, e.g. 8.21.28.120, for your instance.

Step 5: Configure Access & Security

OpenStack has a feature like a firewall. It can whitelist/blacklist your in/out connection. It is called *Security Group*.

1. Go to **Compute > Access & Security** and then open **Security Groups** tab.
2. In **default** row, click **Manage Rules**.
3. Click **Add Rule**, choose **ALL ICMP** rule to enable ping into your instance, and then click **Add**.
4. Click **Add Rule**, choose **HTTP** rule to open HTTP port (port 80), and then click **Add**.
5. Click **Add Rule**, choose **SSH** rule to open SSH port (port 22), and then click **Add**.
6. You can open other ports by creating new rules.

Step 6: SSH to Your Instance

Now, you can SSH your instances to the floating IP address that you got in the step 4. If you are using Ubuntu image, the SSH user will be ubuntu.

Result:

Thus the openstack demo worked successfully.

EX NO.:8
DATE :

Install Hadoop single node cluster and run simple applications like wordcount.

Aim:

To Install Hadoop single node cluster and run simple applications like wordcount.

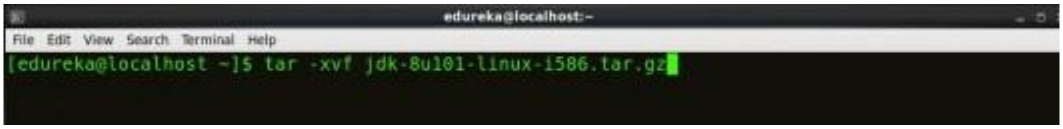
Steps:

Install Hadoop

Step 1: Click here to download the Java 8 Package. Save this file in your home directory.

Step 2: Extract the Java Tar File.

***Command*:** tar -xvf jdk-8u101-linux-i586.tar.gz

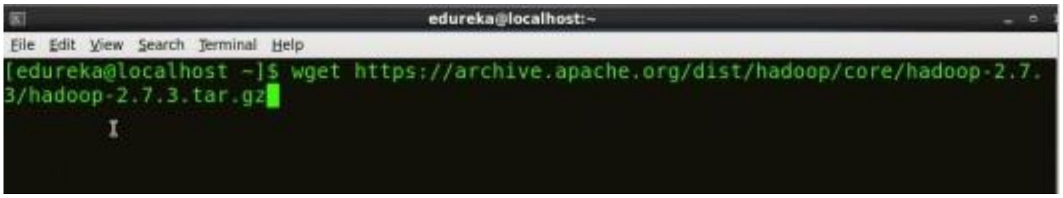


```
[edureka@localhost ~]$ tar -xvf jdk-8u101-linux-i586.tar.gz
```

Fig: Hadoop Installation – Extracting Java Files

Step 3: **Download the Hadoop 2.7.3 Package.**

***Command*:** wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.3/hadoop-2.7.3.tar.gz

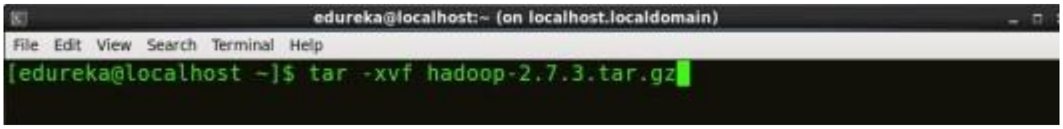


```
[edureka@localhost ~]$ wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.
3/hadoop-2.7.3.tar.gz
```

Fig: Hadoop Installation – Downloading Hadoop

Step 4: Extract the Hadoop tar File.

Command: tar -xvf hadoop-2.7.3.tar.gz


edureka@localhost:~ (on localhost.localdomain)
File Edit View Search Terminal Help

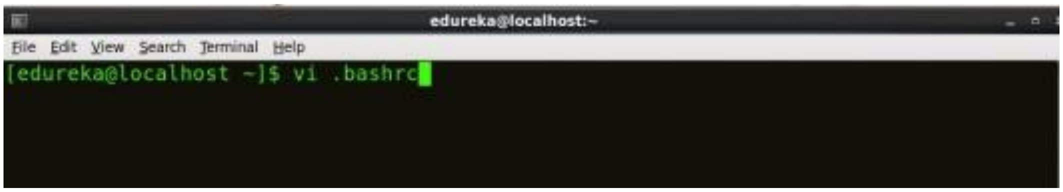
```
[edureka@localhost ~]$ tar -xvf hadoop-2.7.3.tar.gz
```

Fig: Hadoop Installation – Extracting Hadoop Files **Step**

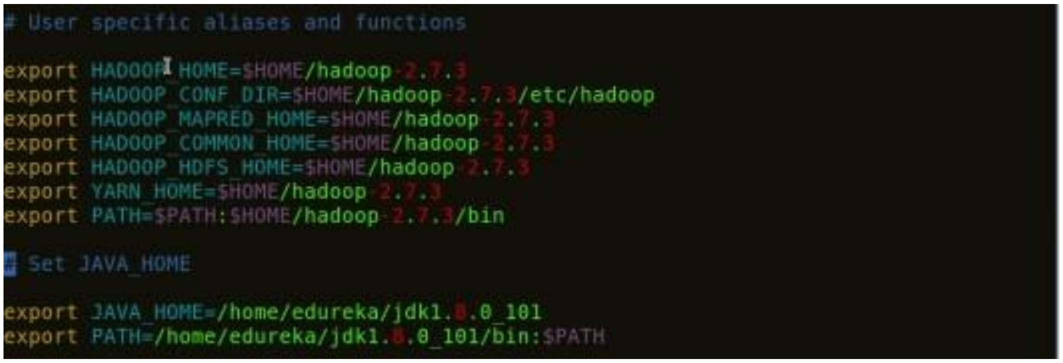
5: Add the Hadoop and Java paths in the bash file (.bashrc). Open**. bashrc**

file. Now, add Hadoop and Java Path as shown below.

Command**:** vi .bashrc



```
[edureka@localhost ~]$ vi .bashrc
```



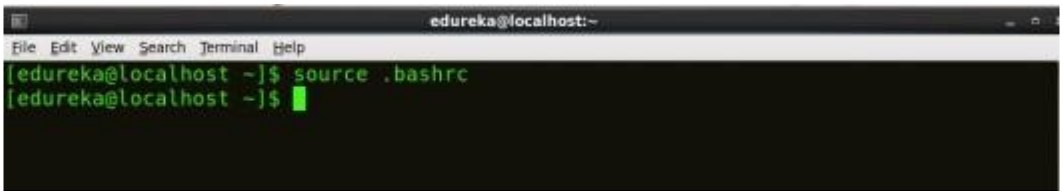
```
# User specific aliases and functions

export HADOOP_HOME=$HOME/hadoop-2.7.3
export HADOOP_CONF_DIR=$HOME/hadoop-2.7.3/etc/hadoop
export HADOOP_MAPRED_HOME=$HOME/hadoop-2.7.3
export HADOOP_COMMON_HOME=$HOME/hadoop-2.7.3
export HADOOP_HDFS_HOME=$HOME/hadoop-2.7.3
export YARN_HOME=$HOME/hadoop-2.7.3
export PATH=$PATH:$HOME/hadoop-2.7.3/bin

# Set JAVA_HOME

export JAVA_HOME=/home/edureka/jdk1.8.0_101
export PATH=/home/edureka/jdk1.8.0_101/bin:$PATH
```

Fig: Hadoop Installation – Setting Environment Variable

Then, save the bash file and close it.

For applying all these changes to the current Terminal, execute the source command.

Command**:** source .bashrc



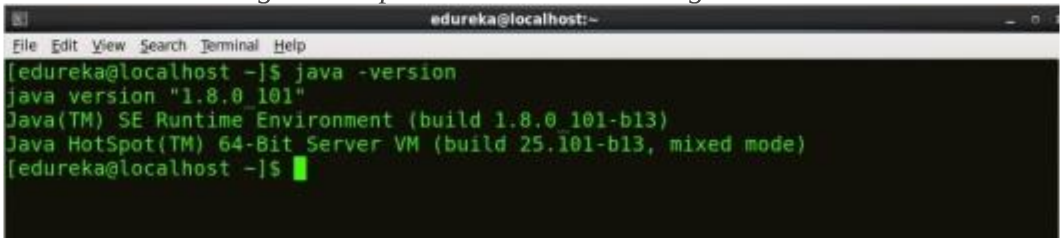
```
[edureka@localhost ~]$ source .bashrc
[edureka@localhost ~]$ 
```

Fig: Hadoop Installation – Refreshing environment variables

To make sure that Java and Hadoop have been properly installed on your system and can be accessed through the Terminal, execute the java -version and hadoop version commands.

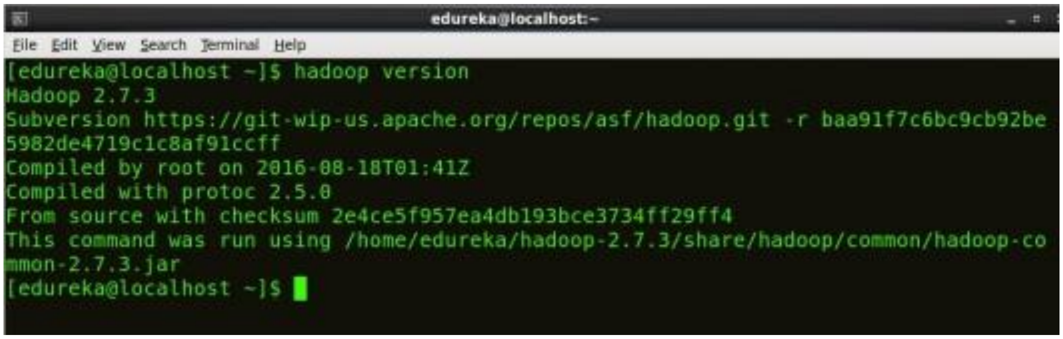
Command**:** java -version

Fig: Hadoop Installation – Checking Java Version



```
[edureka@localhost ~]$ java -version
java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)
[edureka@localhost ~]$ 
```

***Command*:** hadoop version



```
[edureka@localhost ~]$ hadoop version
Hadoop 2.7.3
Subversion https://git-wip-us.apache.org/repos/asf/hadoop.git -r baa91f7c6bc9cb92be
5982de4719c1c8af91ccff
Compiled by root on 2016-08-18T01:41Z
Compiled with protoc 2.5.0
From source with checksum 2e4ce5f957ea4db193bce3734ff29ff4
This command was run using /home/edureka/hadoop-2.7.3/share/hadoop/common/hadoop-co
mmon-2.7.3.jar
[edureka@localhost ~]$
```

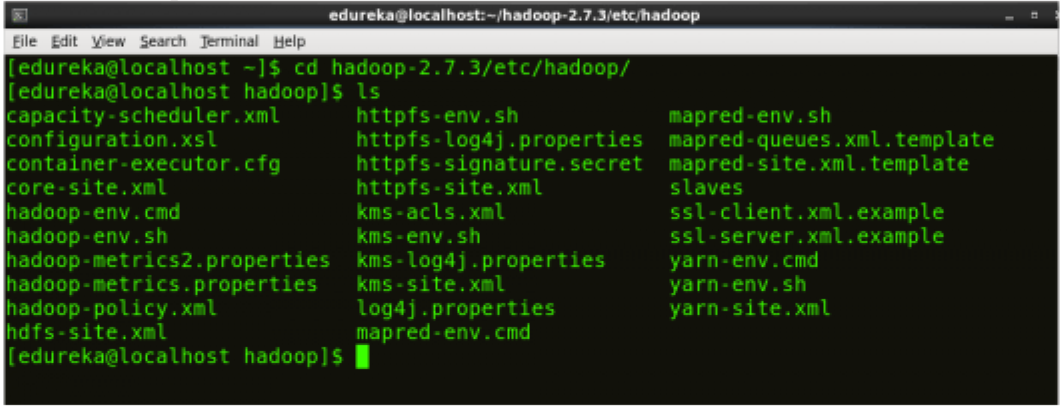
Fig: Hadoop Installation – Checking Hadoop Version

Step 6: Edit the **Hadoop Configuration files**.

Command: cd hadoop-2.7.3/etc/hadoop/

Command: ls

All the Hadoop configuration files are located in **hadoop-2.7.3/etc/hadoop** directory as you can see in the snapshot below:



edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

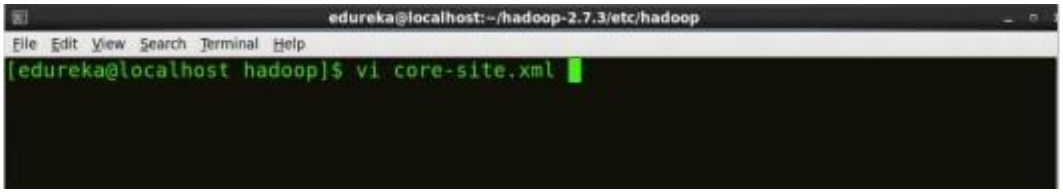
```
[edureka@localhost ~]$ cd hadoop-2.7.3/etc/hadoop/
[edureka@localhost hadoop]$ ls
capacity-scheduler.xml      httpfs-env.sh             mapred-env.sh
configuration.xsl           httpfs-log4j.properties   mapred-queues.xml.template
container-executor.cfg      httpfs-signature.secret   mapred-site.xml.template
core-site.xml               httpfs-site.xml           slaves
hadoop-env.cmd              kms-acls.xml              ssl-client.xml.example
hadoop-env.sh               kms-env.sh                ssl-server.xml.example
hadoop-metrics2.properties  kms-log4j.properties      yarn-env.cmd
hadoop-metrics.properties   kms-site.xml              yarn-env.sh
hadoop-policy.xml           log4j.properties          yarn-site.xml
hdfs-site.xml               mapred-env.cmd
[edureka@localhost hadoop]$
```

Fig: Hadoop Installation – Hadoop Configuration Files

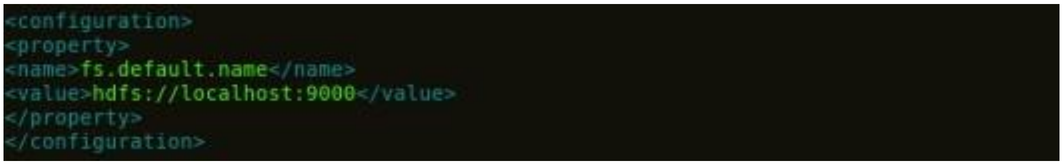
Step 7: Open *core-site.xml* and edit the property mentioned below inside configuration tag:

core-site.xml informs Hadoop daemon where NameNode runs in the cluster. It contains configuration settings of Hadoop core such as I/O settings that are common to HDFS & MapReduce.

Command**:** vi core-site.xml

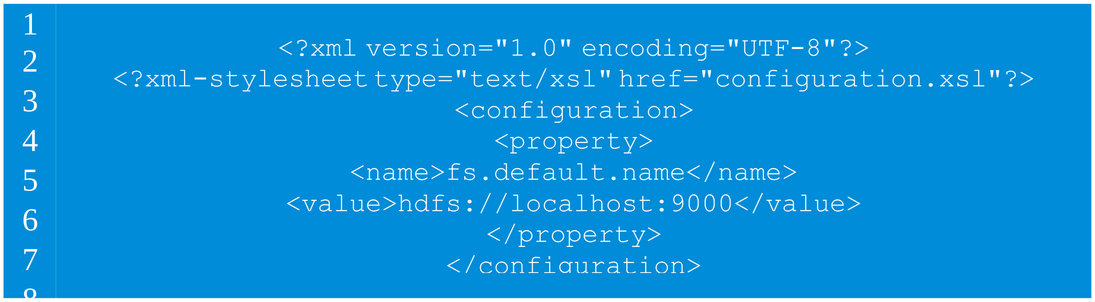

edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ vi core-site.xml
```



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

Fig: Hadoop Installation – Configuring core-site.xml

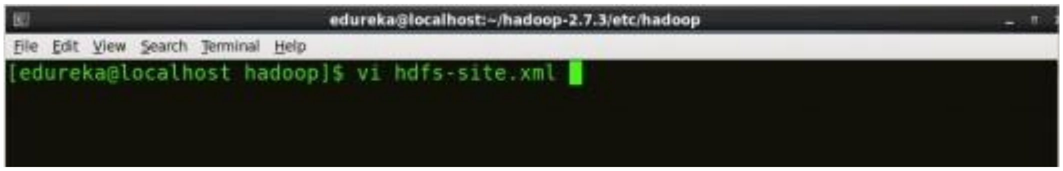


```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9000</value>
</property>
</configuration>
```

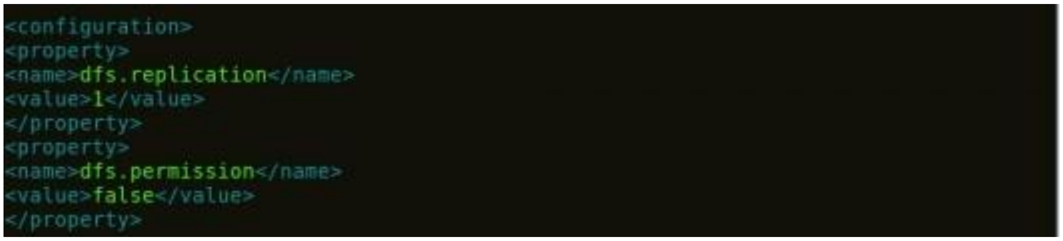
Step 8: Edit *hdfs-site.xml* and edit the property mentioned below inside **configuration tag:**

hdfs-site.xml contains configuration settings of HDFS daemons (i.e. NameNode, DataNode, Secondary NameNode). It also includes the replication factor and block size of HDFS.

Command**:** vi hdfs-site.xml

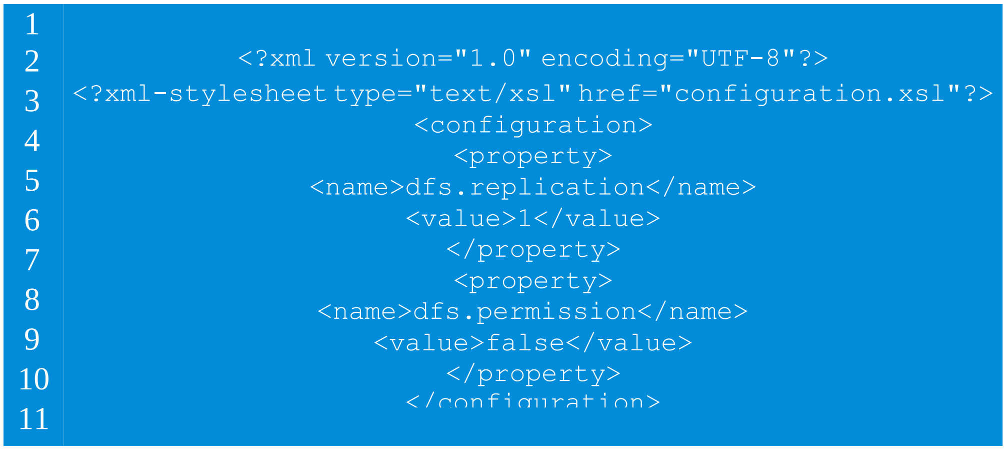

edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ vi hdfs-site.xml
```



```
<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.permission</name>
<value>false</value>
</property>
```

Fig: Hadoop Installation – Configuring hdfs-site.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.permission</name>
<value>false</value>
</property>
</configuration>
```

Step 9: Edit the *mapred-site.xml* file and edit the property mentioned below

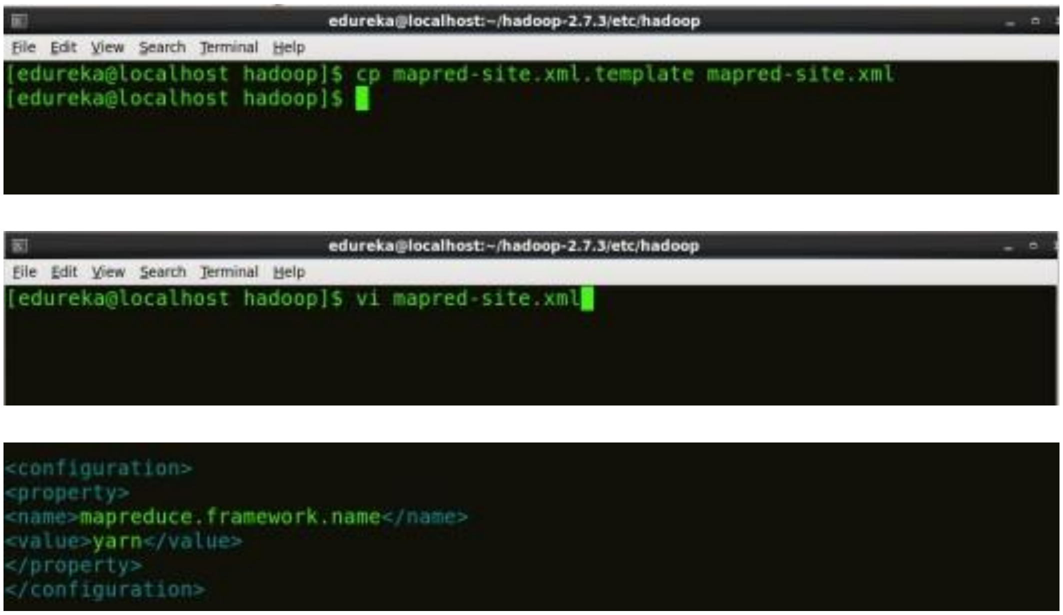
inside configuration tag:

mapred-site.xml contains configuration settings of MapReduce application like number of JVM that can run in parallel, the size of the mapper and the reducer process, CPU cores available for a process, etc.

In some cases, mapred-site.xml file is not available. So, we have to create the mapred- site.xml file using mapred-site.xml template.

***Command*:** cp mapred-site.xml.template mapred-site.xml

***Command*:** vi mapred-site.xml.



edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ cp mapred-site.xml.template mapred-site.xml
[edureka@localhost hadoop]$
```

edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ vi mapred-site.xml
```

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

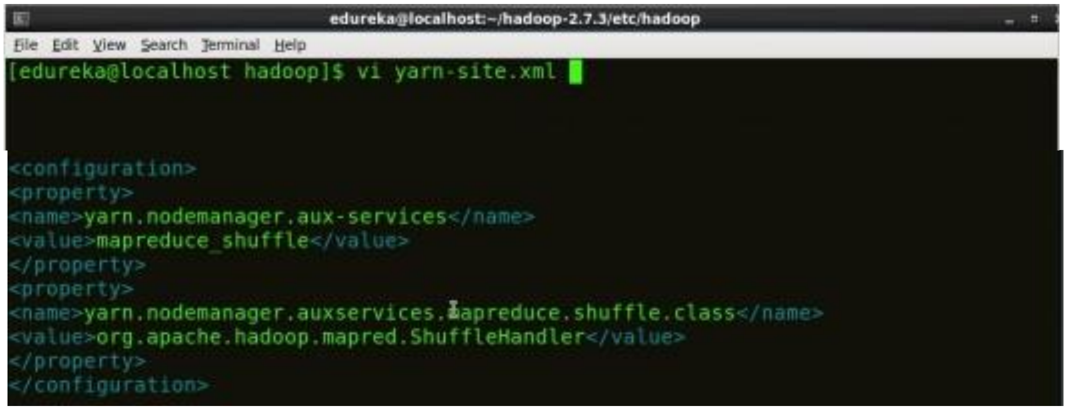
Fig: Hadoop Installation – Configuring mapred-site.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>
```

Step 10: Edit *yarn-site.xml* and edit the property mentioned below inside configuration tag:

yarn-site.xml contains configuration settings of ResourceManager and NodeManager like application memory management size, the operation needed on program & algorithm, etc.

***Command*:** vi yarn-site.xml


edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ vi yarn-site.xml
```

```
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<property>
<name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
</configuration>
```

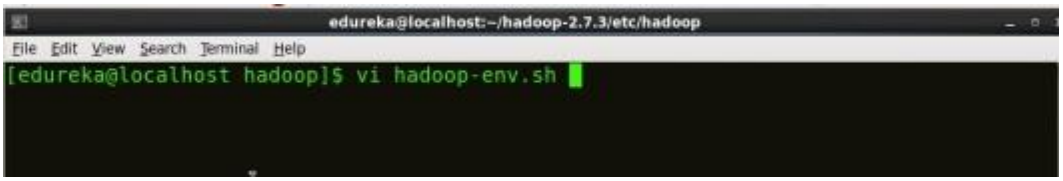
Fig: Hadoop Installation – Configuring yarn-site.xml

Step 11: Edit *hadoop-env.sh* and add the Java Path as mentioned below:

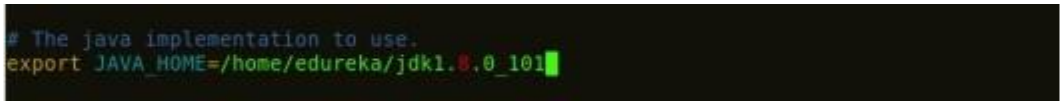
```
<?xml version="1.0">
<configuration>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<property>
<name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
</configuration>
```

hadoop-env.sh contains the environment variables that are used in the script to run Hadoop like Java home path, etc.

Command**:** vi hadoop–env.sh


edureka@localhost:~/hadoop-2.7.3/etc/hadoop
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ vi hadoop-env.sh
```



```
# The java implementation to use.
export JAVA_HOME=/home/edureka/jdk1.8.0_101
```

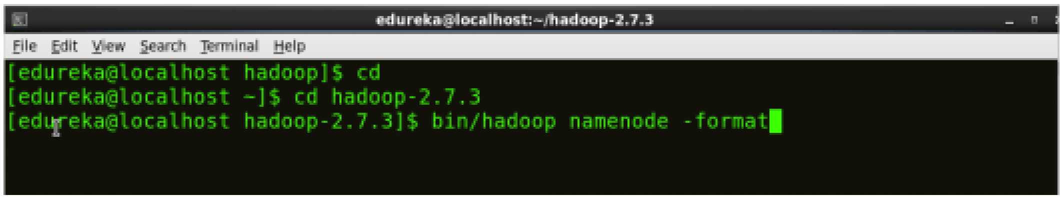
Fig: Hadoop Installation – Configuring hadoop-env.sh **Step**

12: Go to Hadoop home directory and format the NameNode.

Command**:** cd

Command**:** cd hadoop-2.7.3

Command**:** bin/hadoop namenode -format


edureka@localhost:~/hadoop-2.7.3
File Edit View Search Terminal Help

```
[edureka@localhost hadoop]$ cd
[edureka@localhost ~]$ cd hadoop-2.7.3
[edureka@localhost hadoop-2.7.3]$ bin/hadoop namenode -format
```

Fig: Hadoop Installation – Formatting NameNode

This formats the HDFS via NameNode. This command is only executed for the first time. Formatting the file system means initializing the directory specified by the dfs.name.dir variable.

Never format, up and running Hadoop filesystem. You will lose all your data stored in the HDFS.

Step 13: **Once the NameNode is formatted, go to hadoop-2.7.3/sbin directory and start all the daemons.**

Command: cd hadoop-2.7.3/sbin

Either you can start all daemons with a single command or do it individually.

Command: ./start-all.sh

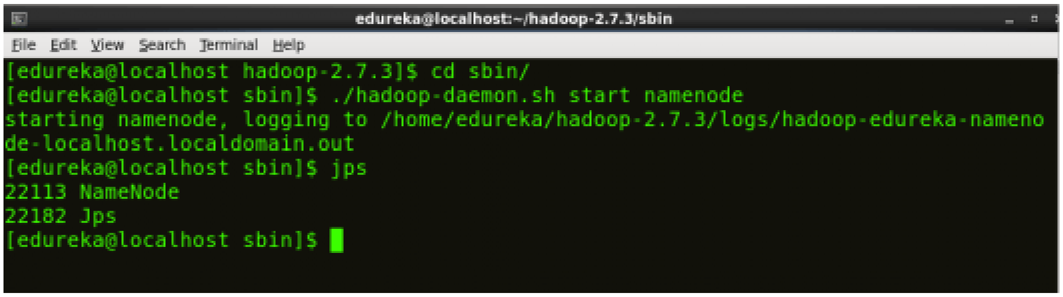
The above command is a combination of ***start-dfs.sh, start-yarn.sh*** & ***mr-jobhistory-daemon.sh***

Or you can run all the services individually as below:

Start NameNode:

The NameNode is the centerpiece of an HDFS file system. It keeps the directory tree of all files stored in the HDFS and tracks all the file stored across the cluster.

Command: ./hadoop-daemon.sh start namenode


edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

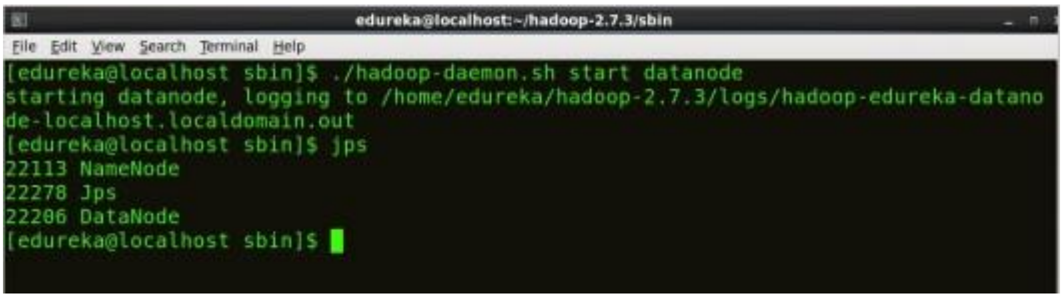
```
[edureka@localhost hadoop-2.7.3]$ cd sbin/
[edureka@localhost sbin]$ ./hadoop-daemon.sh start namenode
starting namenode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-nameno
de-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22182 Jps
[edureka@localhost sbin]$
```

Fig: Hadoop Installation – Starting NameNode

Start DataNode:

On startup, a DataNode connects to the Namenode and it responds to the requests from the Namenode for different operations.

Command: ./hadoop-daemon.sh start datanode



edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

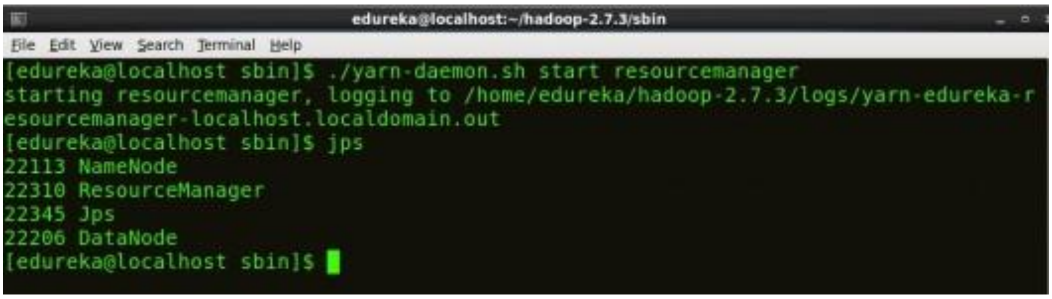
```
[edureka@localhost sbin]$ ./hadoop-daemon.sh start datanode
starting datanode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-datano
de-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22278 Jps
22206 DataNode
[edureka@localhost sbin]$ 
```

Fig: Hadoop Installation – Starting DataNode

Start ResourceManager:

ResourceManager is the master that arbitrates all the available cluster resources and thus helps in managing the distributed applications running on the YARN system. Its work is to manage each NodeManagers and the each application's ApplicationMaster.

Command: ./yarn-daemon.sh start resourcemanager



edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

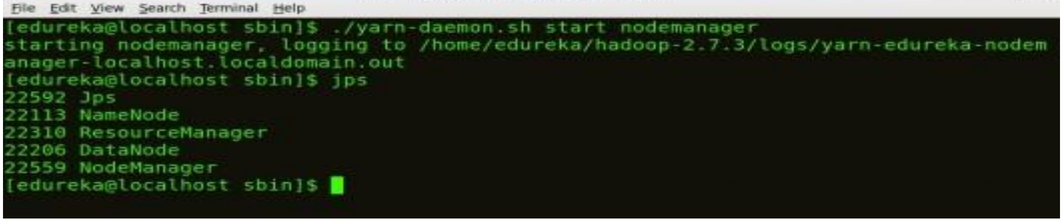
```
[edureka@localhost sbin]$ ./yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /home/edureka/hadoop-2.7.3/logs/yarn-edureka-r
esourcemanager-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22310 ResourceManager
22345 Jps
22206 DataNode
[edureka@localhost sbin]$ 
```

Fig: Hadoop Installation – Starting ResourceManager

Start NodeManager:

The NodeManager in each machine framework is the agent which is responsible for managing containers, monitoring their resource usage and reporting the same to the ResourceManager.

Command: ./yarn-daemon.sh start nodemanager



edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

```
[edureka@localhost sbin]$ ./yarn-daemon.sh start nodemanager
starting nodemanager, logging to /home/edureka/hadoop-2.7.3/logs/yarn-edureka-nodem
anager-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22592 Jps
22113 NameNode
22310 ResourceManager
22206 DataNode
22559 NodeManager
[edureka@localhost sbin]$ 
```

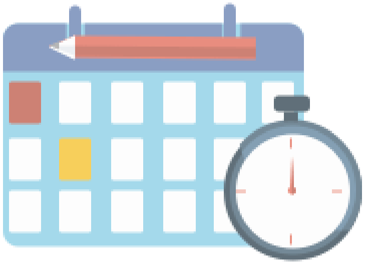


Fig: Hadoop Installation – Starting NodeManager

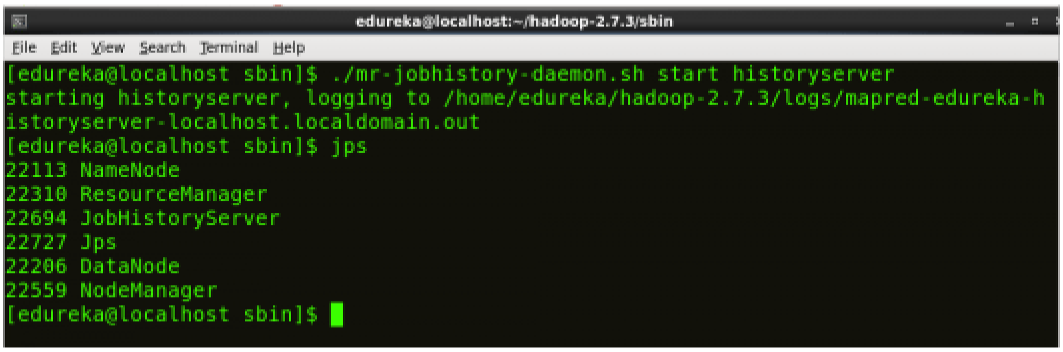
Start JobHistoryServer:

JobHistoryServer is responsible for servicing all job history related requests from client.

***Command*:** ./mr-jobhistory-daemon.sh start historyserver

Step 14: **To check that all the Hadoop services are up and running, run the below command.**

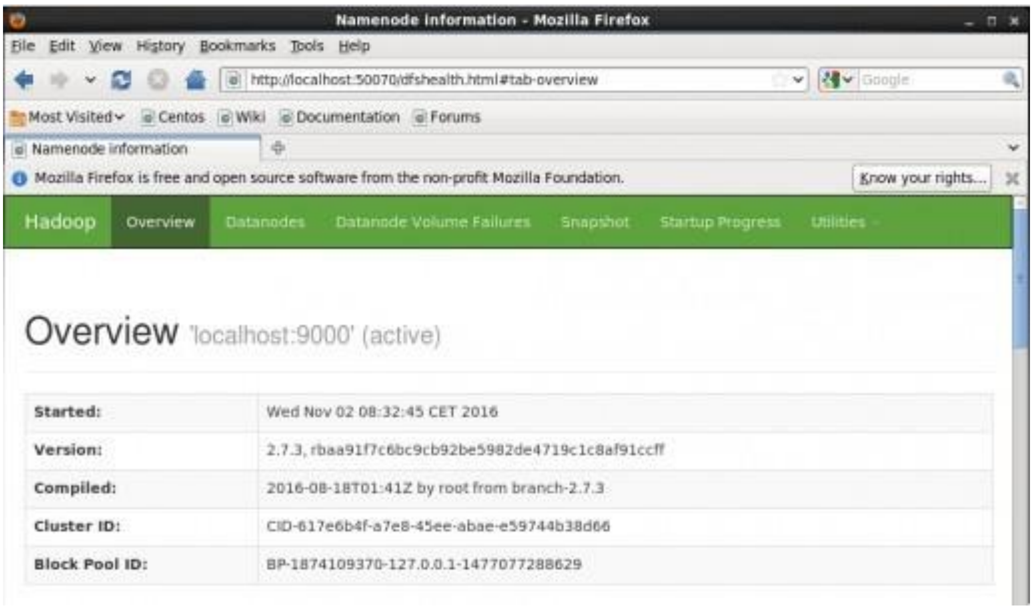
Command: jps


edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help

```
[edureka@localhost sbin]$ ./mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /home/edureka/hadoop-2.7.3/logs/mapred-edureka-h
istoryserver-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22310 ResourceManager
22694 JobHistoryServer
22727 Jps
22206 DataNode
22559 NodeManager
[edureka@localhost sbin]$
```

Fig: Hadoop Installation – Checking Daemons

Step 15: Now open the Mozilla browser and go to **localhost**:**50070/dfshealth.html** to check the NameNode interface.



Namenode information - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:50070/dfshealth.html#tab-overview
Google
Most Visited Centos Wiki Documentation Forums
Namenode information

Mozilla Firefox is free and open source software from the non-profit Mozilla Foundation.

Know your rights...
Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities

Overview 'localhost:9000' (active)

Started:	Wed Nov 02 08:32:45 CET 2016
Version:	2.7.3, rbaa91f7c6bc9cb92be5982de4719c1c8af91ccff
Compiled:	2016-08-18T01:41Z by root from branch-2.7.3
Cluster ID:	CID-617e6b4f-a7e8-45ee-abae-e59744b38d66
Block Pool ID:	BP-1874109370-127.0.0.1-1477077288629

Fig: Hadoop Installation – Starting WebUI

Congratulations, you have successfully installed a single node Hadoop cluster

Result:

Thus the Hadoop one cluster was installed and simple applications executed successfully.

VIVA QUESTIONS AND ANSWERS

1.Define Cloud Computing with example.
Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

2.What is the working principle of Cloud Computing?
The cloud is a collection of computers and servers that are publicly accessible via the Internet. This hardware is typically owned and operated by a third party on a consolidated basis in one or more data center locations. The machines can run any combination of operating systems.

3.What are the advantages and disadvantages of Cloud Computing?
Advantages

Lower-Cost Computers for Users
Improved Performance
Lower IT Infrastructure Costs
Fewer Maintenance Issues
Lower Software Costs
Instant Software Updates
Increased Computing Power
Unlimited Storage Capacity
Increased Data Safety
Improved Compatibility Between Operating Systems
Improved Document Format Compatibility
Easier Group Collaboration
Universal Access to Documents
Latest Version Availability
Removes the Tether to Specific Devices

Disadvantages

Requires a Constant Internet Connection
Doesn't Work Well with Low-Speed Connections
Can Be Slow
Features Might Be Limited
Stored Data Might Not Be Secure
If the Cloud Loses Your Data, You're Screwed

4.What is distributed system?
A *distributed system* is a software system in which components located on networked computers communicate and coordinate their actions by passing messages. The components interact with each other in order to achieve a common goal.

Three significant characteristics of distributed systems are:

- ✓ Concurrency of components
- ✓ Lack of a global clock
- ✓ Independent failure of components
- ✓ What is cluster?
- ✓ Acomputingclusterconsistsofinterconnectedstand-alonecomputerswhichworkcooperativelyasasingleintegratedcomputingresource.Inthepast,clusteredcomputersystemshavedemonstrated

5.What is grid computing?

Grid Computing enables virtuals organizations to share geographically distributed resources as they pursue common goals, assuming the absence of central location, central control, omniscience, and an existing trust relationship.

(or)

- ✓ Gridtechnologydemandsnewdistributedcomputingmodels,software/middlewaresupport,networkprotocols,andhardwareinfrastructures.
- ✓ Nationalgridprojectsarefollowedbyindustrialgridplat-formdevelopmentbyIBM,Microsoft,Sun,HP,Dell,Cisco,EMC,PlatformComputing,andothers. Newgridserviceproviders(GSPs)andnewgridapplicationshaveemergedrapidly,similartothegrowthofInternetandwebservicesinthepasttwodecades.
- ✓ gridsystemsareclassifiedinessentiallytwocategories:computationalordatagridsandP2Pgrids.

6.What are the business areas needs in Grid computing?

- ✓ Life Sciences
- ✓ Financial services
- ✓ Higher Education
- ✓ Engineering Services
- ✓ Government
- ✓ Collaborative games

7.List out the Grid Applications:

- ✓ Application partitioning that involves breaking the problem into discrete pieces
- ✓ Discovery and scheduling of tasks and workflow
- ✓ Data communications distributing the problem data where and when it is required
- ✓ Provisioning and distributing application codes to specific system nodes
- ✓ Autonomic features such as self-configuration, self-optimization, self-recovery and self-management

8.List some grid computing toolkits and frameworks?

- ✓ Globus Toolkit Globus Resource Allocation Manager(GRAM)
- ✓ Grid Security Infrastructure(GSI)
- ✓ Information Services
- ✓ Legion, Condor and Condor-G
- ✓ NIMROD, UNICORE, NMI.

9.What are Desktop Grids?

These are grids that leverage the compute resources of desktop computers.
Because of the true (but unfortunate) ubiquity of Microsoft® Windows® operating
system in corporations, desktop grids are assumed to apply to the Windows environment.
The Mac OS™ environment is supported by a limited number of vendors.

10.What are Server Grids?

- ✓ Some corporations, while adopting Grid Computing , keep it limited to server resources that are within the purview of the IT department.
- ✓ Special servers, in some cases, are bought solely for the purpose of creating an internal “utility grid” with resources made available to various departments.
- ✓ No desktops are included in server grids. These usually run some flavor of the Unix/Linux operating system.

11.Define Opennebula.

OpenNebula is an open source management tool that helps virtualized data centers oversee private clouds, public clouds and hybrid clouds. ... OpenNebula is vendor neutral, as well as platform- and API-agnostic. It can use KVM, Xen or VMware hypervisors.

12.Define Eclipse.

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment.

13.Define Netbeans.

NetBeans is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages. NetBeans is also referred to as a platform of modular components used for developing Java desktop applications.

14.Define Apache Tomcat.

Apache Tomcat (or Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run."

15.What is private cloud?

The *private cloud* is built within the domain of an intranet owned by a single organization. Therefore, they are client owned and managed. Their access is limited to the owning clients and their partners. Their deployment was not meant to sell capacity over the Internet through publicly accessible interfaces. Private clouds give local users a flexible and agile private infrastructure to run service workloads within their administrative domains.

16.What is public cloud?

A *public cloud* is built over the Internet, which can be accessed by any user who has paid for the service. Public clouds are owned by service providers. They are accessed by subscription. Many companies have built public clouds, namely Google App Engine, Amazon AWS, Microsoft Azure, IBM Blue Cloud, and Salesforce Force.com. These are commercial providers that offer a publicly accessible remote interface for creating and managing VM instances within their proprietary infrastructure.

17. What is hybrid cloud?

A *hybrid cloud* is built with both public and private clouds, Private clouds can also support a *hybrid cloud* model by supplementing local infrastructure with computing capacity from an external public cloud. For example, the *research compute cloud* (RC2) is a private cloud built by IBM.

18.What is a Community Cloud ?

A community cloud in computing is a collaborative effort in which infrastructure is shared between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.), whether managed internally or by a third-party and hosted internally or externally. This is controlled and used by a group of organizations that have shared interest. The costs are spread over fewer users than a public cloud (but more than a private cloud

19.Define IaaS?

The IaaS layer offers storage and infrastructure resources that is needed to deliver the Cloud services. It only comprises of the infrastructure or physical resource. Top IaaS Cloud Computing Companies: Amazon (EC2), Rackspace, GoGrid, Microsoft, Terremark and Google.

20.Define PaaS?

PaaS provides the combination of both, infrastructure and application. Hence, organisations using PaaS don’t have to worry for infrastructure nor for services. Top PaaS Cloud Computing Companies: Salesforce.com, Google, Concur Technologies, Ariba, Unisys and Cisco..

21.Define SaaS?

In the SaaS layer, the Cloud service provider hosts the software upon their servers. It can be defined as a in model in which applications and softwares are hosted upon the server and made available to customers over a network. Top SaaS Cloud Computing Companies: Amazon Web Services, AppScale, CA Technologies, Engine Yard, Salesforce and Windows Azure.

22.What is meant by virtualization?

Virtualizationisacomputerarchitecturetechnologybywhichmultiplevirtualmachines (VMs)are multipl exedin the same hardwar emachine.Theideaof VMs canbe dated back to the 1960s.The purpose of a VM is to enhance resource sharing by many users and improve computer performance interms of resource utilization and application flexibility.

23.What are the implementation levels of virtualization?

The virtualization types are following

1. OS-level virtualization
2. ISA level virtualization
3. User-ApplicationLevel virtualization
4. hardware level virtualization
5. library level virtualization

24.List the requirements of VMM?

There are three requirements for a VMM.

First, a VMM should provide an environment for programs which is essentially identical to the original machine.

Second, programs run in this environment should show, at worst, only minor decreases in speed.

Third, a VMM should be in complete control of the system resources.

25.Explain Host OS and Guest OS?

A comparison of the differences between a host system, a guest system, and a virtual machine within a virtual infrastructure.

A host system (host operating system) would be the primary & first installed operating system. If you are using a bare metal Virtualization platform like Hyper-V or ESX, there really isn’t a host operating system besides the Hypervisor. If you are using a Type-2 Hypervisor like VMware Server or Virtual Server, the host operating system is whatever operating system those applications are installed into.

A guest system (guest operating system) is a virtual guest or virtual machine (VM) that is installed under the host operating system. The guests are the VMs that you run in your virtualization platform.

26.Write the steps for live VM migration?

The five steps for live VM migration is

Stage 0: *Pre-Migration*
Active VM on Host A
Alternate physical host may be preselected for migration
Block devices mirrored and free resources maintained

Stage 1: *Reservation*
Initialize a container on the target host

Stage 2: *Iterative pre-copy*
Enable shadow paging
Copy dirty pages in successive rounds.

Stage 3: *Stop and copy*
Suspend VM on host A
Generate ARP to redirect traffic to Host B
Synchronize all remaining VM state to Host B

Stage 4: *Commitment*
VM state on Host A is released

Stage 5: *Activation*
VM starts on Host B
Connects to local devices

Resumes normal operation

27..Define Globus Toolkit: Grid Computing Middleware

- ✓ Globus is open source grid software that addresses the most challenging problmes in distributed resources sharing.
- ✓ The Globus Toolkit includes software services and libraries for distributed security, resource management, monitoring and discovery, and data management.

28.Define Blocks in HDFS

- ✓ A disk has a block size, which is the minimum amount of data that it can read or write. Filesystems for a single disk build on this by dealing with data in blocks, which are an integral multiple of the disk block size. Filesystem blocks are typically a few kilobytes in size, while disk blocks are normally 512 bytes. This is generally transparent to the filesystem user who is simply reading or writing a file—of whatever length.

29.Define Namenodes and Datanodes

- ✓ An HDFS cluster has two types of node operating in a master-worker pattern:
  - ➢ a *namenode* (the master) and
  - ➢ a number of *datanodes*(workers).
- ✓ The namenode manages the filesystem namespace. It maintains the filesystem tree and the metadata for all the files and directories in the tree. This information is stored persistently on the local disk in the form of two files: the namespace image and the edit log.
- ✓ The namenode also knows the datanodes on which all the blocks for a given file are located, however, it does not store block locations persistently, since this information is reconstructed from datanodes when the system starts.

30.Define HADOOP.

Hadoop is an open source, Java-based programming framework that supports the processing and storage of extremely large data sets in a distributed computing environment. It is part of the Apache project sponsored by the Apache Software Foundation.

31.Define HDFS.

Hadoop Distributed File System (HDFS) is a Java-based file system that provides scalable and reliable data storage that is designed to span large clusters of commodity servers. HDFS, MapReduce, and YARN form the core of Apache™ Hadoop®.

32.Write about HADOOP.

Hadoop was created by Doug Cutting and Mike Cafarella in 2005. Cutting, who was working at Yahoo! at the time, named it after his son's toy elephant. It was originally developed to support distribution for the Nutch search engine project.

33.Definition of *Grid Portal*:

A *Grid Portal* provides an efficient infrastructure to put Grid-empowered applications on corporate Intranet/Internet.

34.Define GAE.

Google App Engine (often referred to as GAE or simply App Engine) is a Platform as a Service and cloud computing platform for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand.

35.What is Cloudsim?

CloudSim is a simulation toolkit that supports the modeling and simulation of the core functionality of cloud, like job/task queue, processing of events, creation of cloud entities(datacenter, datacenter brokers, etc), communication between different entities, implementation of broker policies, etc. This toolkit allows to:

- Test application services in a repeatable and controllable environment.
- Tune the system bottlenecks before deploying apps in an actual cloud.
- Experiment with different workload mix and resource performance scenarios on simulated infrastructure for developing and testing adaptive application provisioning techniques

36.Core features of CloudSim are:

- The Support of modeling and simulation of large scale computing environment as federated cloud data centers, virtualized server hosts, with customizable policies for provisioning host resources to virtual machines and energy-aware computational resources
- It is a self-contained platform for modeling cloud's service brokers, provisioning, and allocation policies.
- It supports the simulation of network connections among simulated system elements.
- Support for simulation of federated cloud environment, that inter-networks resources from both private and public domains.
- Availability of a virtualization engine that aids in the creation and management of multiple independent and co-hosted virtual services on a data center node.
- Flexibility to switch between space shared and time shared allocation of processing cores to virtualized services.

37.Uses of Cloudsim.

- Load Balancing of resources and tasks
- Task scheduling and its migrations
- Optimizing the Virtual machine allocation and placement policies
- Energy-aware Consolidations or Migrations of virtual machines
- Optimizing schemes for Network latencies for various cloud scenarios

38.Define OpenStack.

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed and provisioned through APIs with common authentication mechanisms.A dashboard is also available, giving administrators control while empowering their users to provision resources through a web interface.

39.Define Trystack.

TryStack is a great way to take OpenStack for a spin without having to commit to a full deployment.

This free service lets you test what the cloud can do for you, offering networking, storage and compute instances, without having to go all in with your own hardware.

It's a labor of love spearheaded by three Red Hat OpenStack experts Will Foster, Kambiz Aghaiepour and Dan Radez.

TryStack's set-up must bear the load of anyone who wants to use it, but instead of an equally boundless budget and paid staff, it was originally powered by donated equipment and volunteers from Cisco, Dell, Equinix, NetApp, Rackspace and Red Hat who pulled together for this OpenStack Foundation project.

40.Define Hadoop.

Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs.