

## INDEX

Exp. No.	DATE	Title	Page No.	STAFF SIGNATURE
1		Installation of Virtual Box / VMware	1	
2		Installation of C Compiler in the VM	5	
3		Installation OF Google App Engine	7	
4		Use of GAE Launcher	9	
5		Simulation of Cloud Scenario using CloudSim	11	
6		Transfer of Files from one VM to Another VM	13	
7		Virtual Machine using TRY STACK	15	
8		Installation of HADOOP	17	

**EXPN No : 1**

**DATE :**

## **INSTALLATION OF VIRTUALBOX/VMWARE WORKSTATION**

### **Aim:**

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

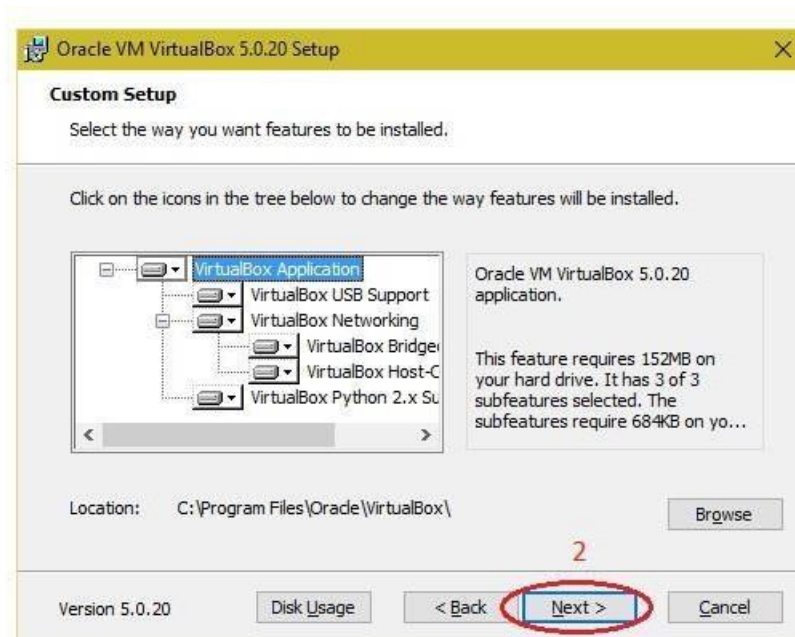
### **Procedure:**

#### **Steps to install Virtual Box:**

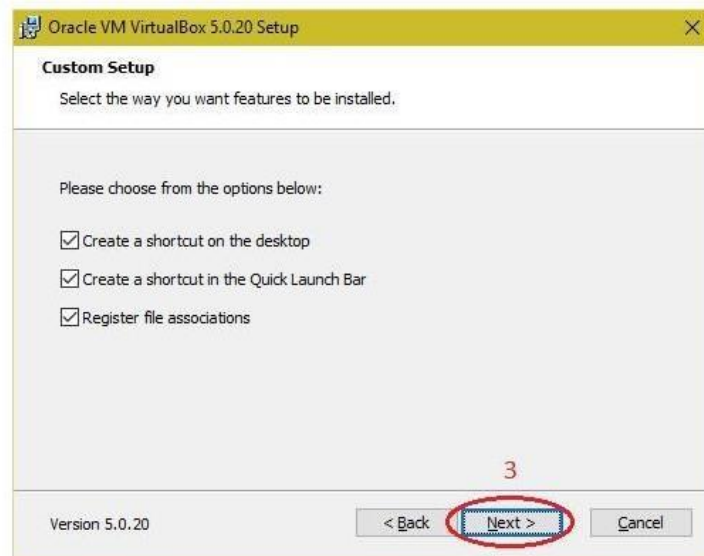
1. Download the Virtual box exe and click the .exe file and select next button.



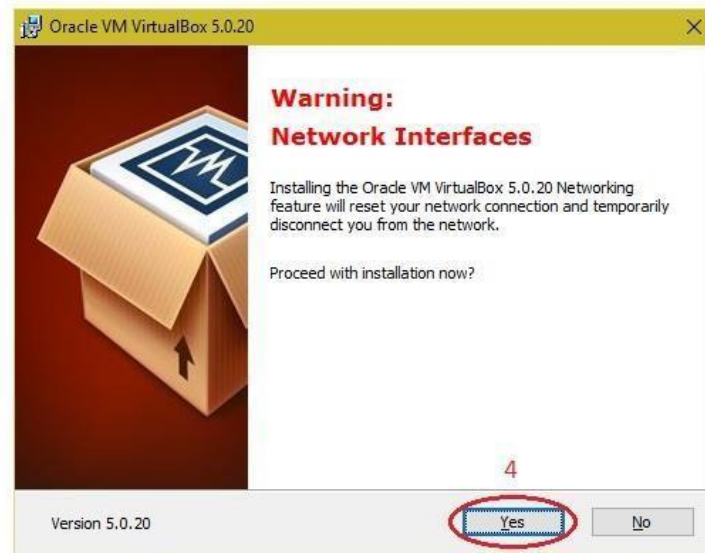
2. Click the next button.



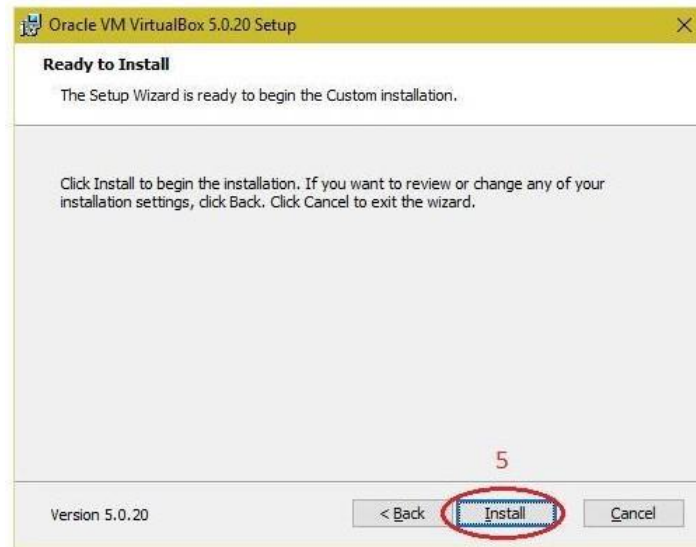
### 3. Click the next button



### 4. Click the YES button.



### 5. Click the install button



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



# VirtualBox

### 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

### **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 Virtual 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.

### **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.

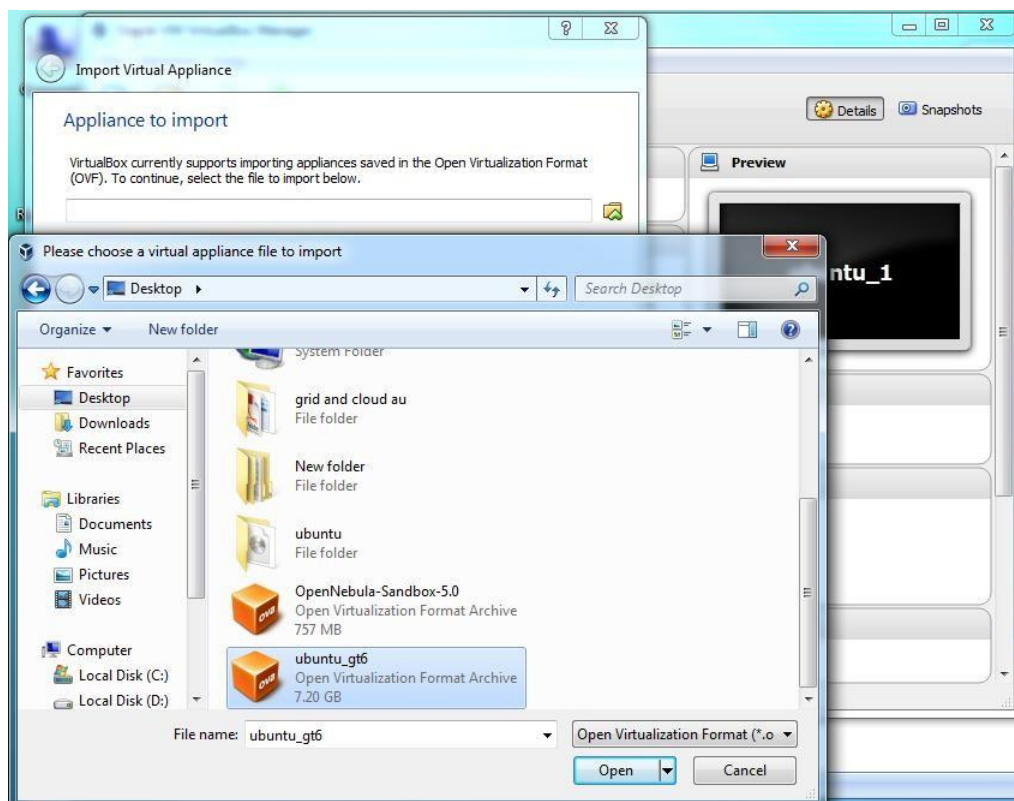
## **INSTALL A C COMPILER IN THE VIRTUAL MACHINE CREATED USING VIRTUAL BOX AND EXECUTE SIMPLE PROGRAMS**

**Aim:**

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

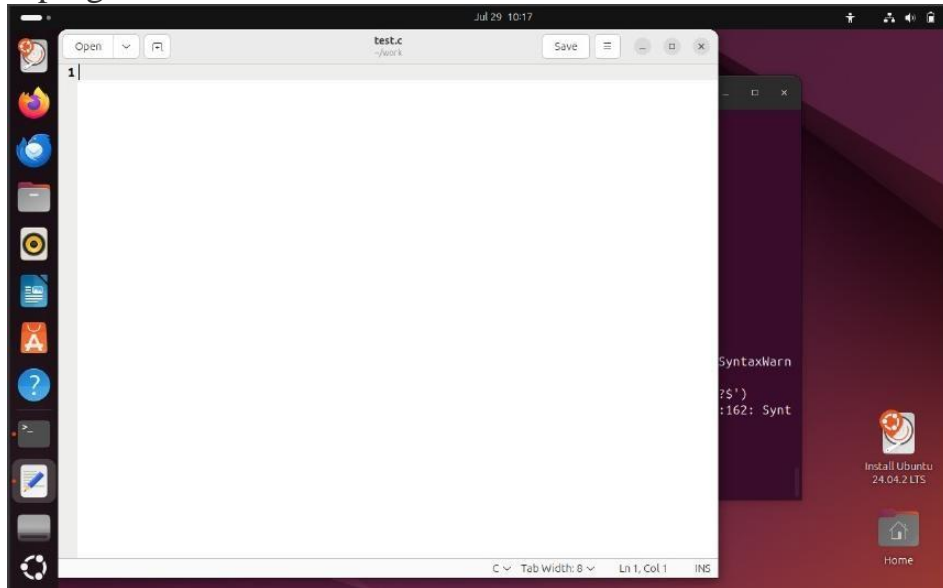
**Procedure:****Steps to import .ova file:**

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.

**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
2. Type `gedit first.c`
3. Type the c program



4. Running the C program
5. Display the output:

```
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
ubuntu@ubuntu:~/work$ gcc -o test test.c
ubuntu@ubuntu:~/work$ ls
test  test.c
ubuntu@ubuntu:~/work$ ./test
hello world!ubuntu@ubuntu:~/work$ gedit test.c
ubuntu@ubuntu:~/work$
```

## APPLICATIONS:

Simply running all programs in grid environment.

## RESULT:

Thus the simple C programs executed successfully.

**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
2. Create New Web Application Project.
3. In Eclipse toolbar, click on the Google icon, and select “**New Web Application Project...**”.
4. Write a Python Code to execute.
5. Run it Locally with Command “**python filename.py**”
6. Print “Hello World ! Welcome to Google App Engine”

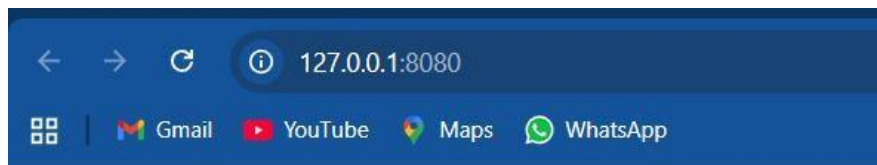
**Code:**

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def hello():
    return "Hello,World!Welcome to Google App Engine"
If __name__ == '__main__':
    app.run(host='127.0.0.1',port=8080,debug=True)
```



## Output:

```
D:\bharathcode\cloud computing>python expn3.py
* Serving Flask app 'expn3'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production de
* Running on http://127.0.0.1:8080
Press CTRL+C to quit
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 320-082-897
127.0.0.1 - - [06/Oct/2025 22:26:37] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [06/Oct/2025 22:26:38] "GET /favicon.ico HTTP/1.1" 404 -
```



## Result:

Thus the simple application was created successfully.

**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.

**Procedure:****1. Download & Setup:**

- Download CloudSim from <https://code.google.com/p/cloudsim/downloads/list>
- Unzip and open Eclipse → File → New → Java Project
- Import the unpacked CloudSim folder into your project.

**2. Initialize CloudSim:**

- `CloudSim.init(num_users, calendar, trace_flag);`

**3. Create Datacenter:**

- Define DatacenterCharacteristics (architecture, OS, host list, allocation policy, timezone, price).

**4. Create Broker:**

- `DatacenterBroker broker = createBroker();`

**5. Create VM:**

- `Vm vm = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared());`
- `broker.submitVmList(vmlist);`

**6. Create Cloudlet:**

- `Cloudlet cloudlet = new Cloudlet(id, length, pesNumber, fileSize, outputSize, utilizationModel, utilizationModel);`
- `broker.submitCloudletList(cloudletList);`

**7. Run Simulation:**

```
Starting CloudSimExample...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 starting...
Broker starting...
VM #0 created, Cloudlet executed.
Simulation completed successfully.
```

## Output :

```
===== 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: 3  
Debt: 35.6  
  
CloudSimExample1 finished!
```

## RESULT:

The simulation was successfully executed.

**USE GAE LAUNCHER TO LAUNCH THE WEB APPLICATIONS.****Aim:**

To Use GAE launcher to launch the web applications.

**Procedure:****1. Install Google App Engine (GAE) SDK:**

- Download and install the Google App Engine SDK for Java/Python from the official Google Cloud website.

**2. Open GAE Launcher:**

- Go to Start Menu → Google App Engine → GAE Launcher (for Windows) or open it from Applications (for macOS/Linux).

**3. Create a New Application:**

- Click File → Add Existing Application (if you already have a project) or File → New Application to create a new one.
- Select your application folder that contains app.yaml (or appengine-web.xml for Java).

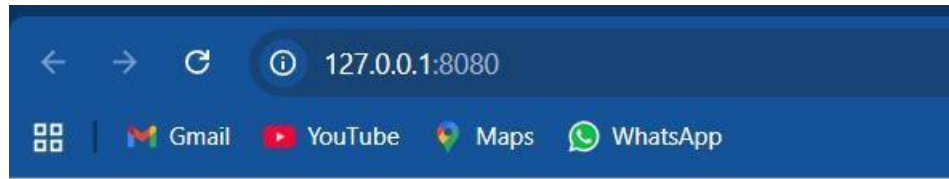
**4. Set Application Configuration:**

- Ensure your application has the correct configuration file (app.yaml or appengine-web.xml) with application ID, runtime, and version.

**5. Run the Application Locally:**

- Click Run in the GAE Launcher.
- The local development server will start (usually at <http://localhost:8080>).
- Open the browser and go to that URL to test your app locally.

## Output :



## Result:

Thus the GAE web applications was created.

**FIND A PROCEDURE TO TRANSFER THE FILES FROM ONE VIRTUAL MACHINE TO ANOTHER VIRTUAL MACHINE.**

**Aim:**

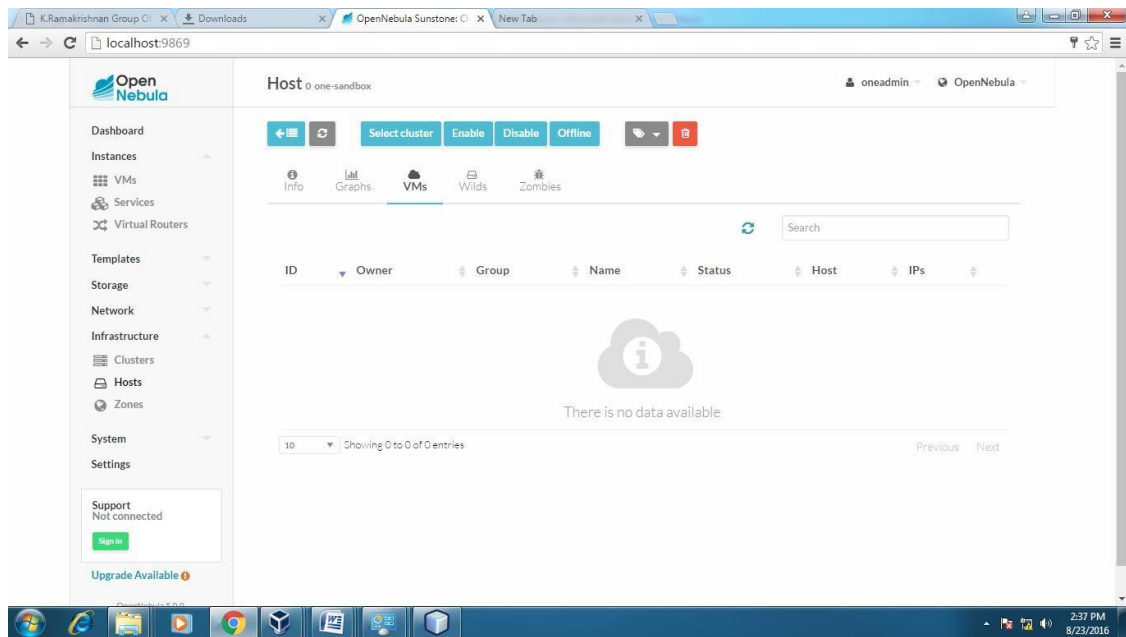
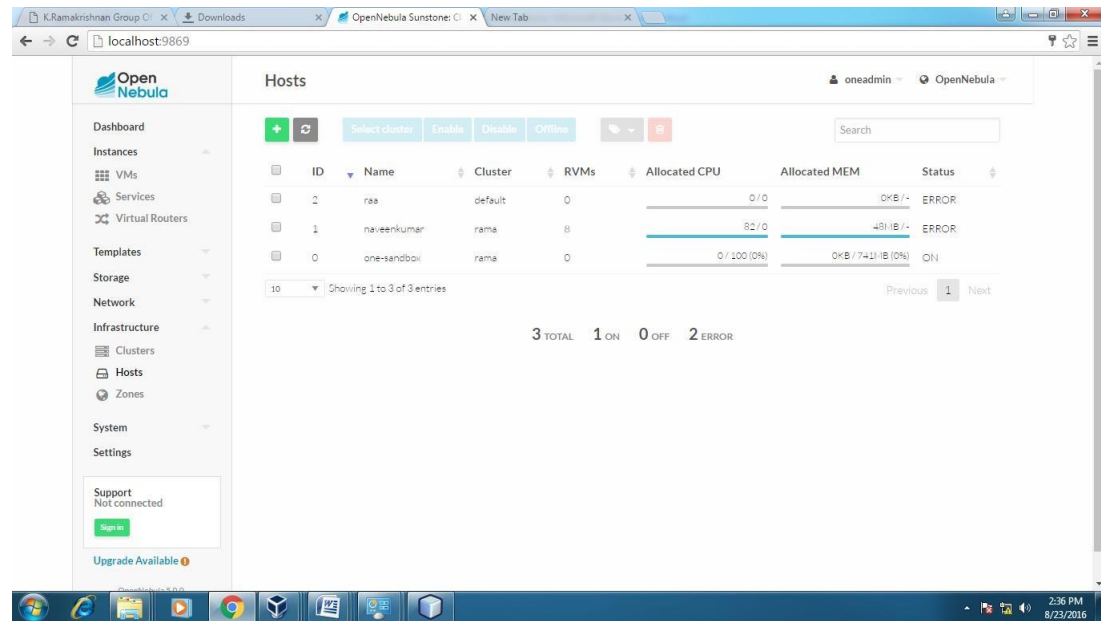
To Find a procedure to transfer the files from one virtual machine to another virtual machine.

**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.

## Output :



## APPLICATIONS:

Easily migrate your virtual machine from one pc to another.

## Result:

Thus the file transfer between VM was successfully completed.

## **FIND A PROCEDURE TO LAUNCH VIRTUAL MACHINE USING TRYSTACK (ONLINE OPENSTACK DEMO VERSION)**

**Aim:**

To Find a procedure to launch virtual machine using trystack.

**Procedure:****Step 1: Create Network**

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**

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

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**.

### Step 4: Configure Floating IP Address .

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

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.

## INSTALL HADOOP SINGLE NODE CLUSTER AND RUN SIMPLE APPLICATIONS LIKE WORDCOUNT.

### Aim :



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

### ALGORITHM:

**STEP 1:** To install Hadoop the primary task is to setup and install java environment

**STEP 2:** The java version that needed to be installed depends on the Hadoop's version. Here we are installing the latest version of Hadoop which is 3.3.0 which supports java version varying from 8-11(runtime only).

**STEP 3:** Use the following link to install java  
<https://www.oracle.com/java/technologies/downloads/#java8-windows>

Linux	macOS	Solaris	Windows
Product/file description		File size	Download
x86 Installer		136.83 MB	 <a href="#">jdk-8u381-windows-i586.exe</a>
x64 Installer		145.55 MB	 <a href="#">jdk-8u381-windows-x64.exe</a>

**STEP 4:** After installing java setup, the java environment in environmental variables directing the bin folder inside the java folder (C:\java\jdk-11.0.2\bin) copy the path till bin folder and paste it in the environmental variable define the new path and add the bin folder location as **JAVA\_HOME=" C:\java\jdk11.0.2\bin"** and apply the changes



**STEP 5:** Now after setting up the java environment check the setup has been successfully set by using **java -version** command in your command prompt and it should display the version of java you have installed.

```
C:\Windows\System32>java -version
java version "1.8.0_381"
Java(TM) SE Runtime Environment (build 1.8.0_381-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.381-b09, mixed mode)
```

**STEP 7:** Hadoop is Unix distribution-based file with tar.gz extension we have to extract the file using the 7-zip manager which supports multiple formats follow this link to install 7-zip

<https://7-zip.org/>

**STEP 8:** Now install the Notepad++ text editor which is further used to modify or edit the configuration file within Hadoop as per our requirement

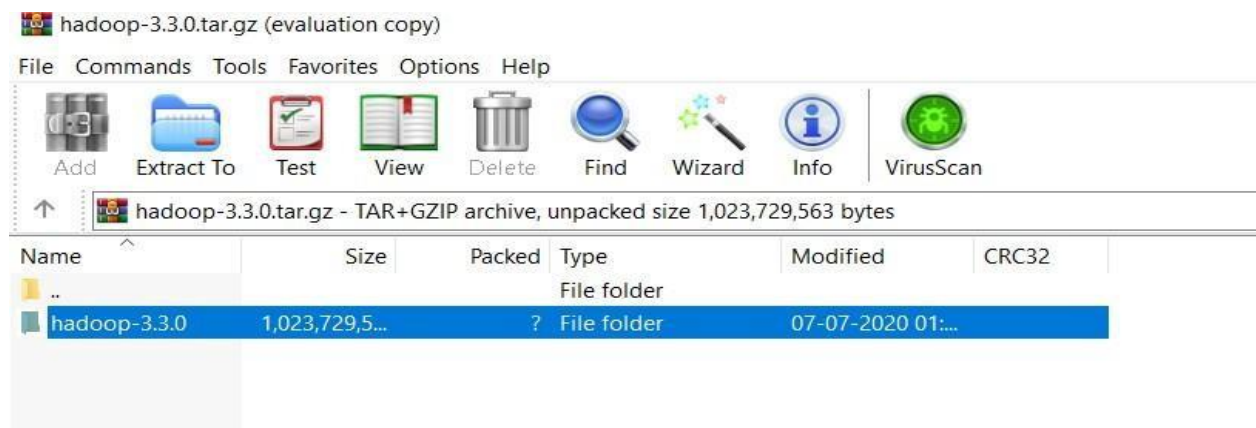
**STEP 9:** After installing and setting up all the required application install Hadoop from the official Apache Hadoop website

<https://hadoop.apache.org/releases.html>

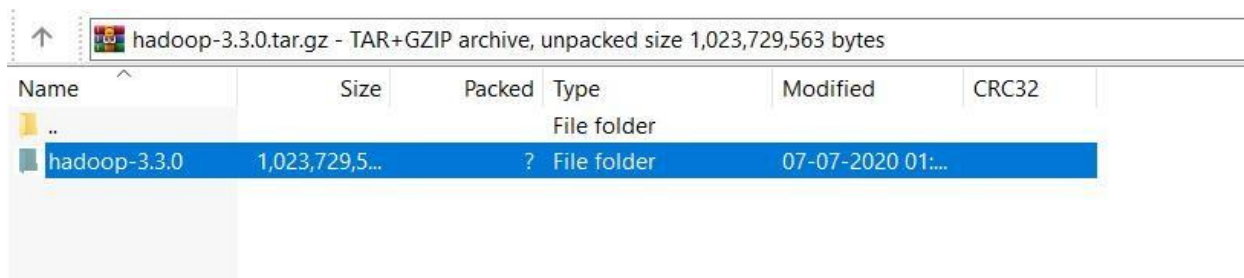
download the binary download which can run directly without any need for compilation.

Version	Release date	Source download	Binary download	Release notes
3.3.6	2023 Jun 23	<a href="#">source (checksum signature)</a>	<a href="#">binary (checksum signature)</a> <a href="#">binary-aarch64 (checksum signature)</a>	<a href="#">Announcement</a>
3.2.4	2022 Jul 22	<a href="#">source (checksum signature)</a>	<a href="#">binary (checksum signature)</a>	<a href="#">Announcement</a>
2.10.2	2022 May 31	<a href="#">source (checksum signature)</a>	<a href="#">binary (checksum signature)</a>	<a href="#">Announcement</a>

**STEP 10:** Run 7-zip manager as administrator and navigate to the path where Hadoop is located for extract the compiled binary download of Hadoop



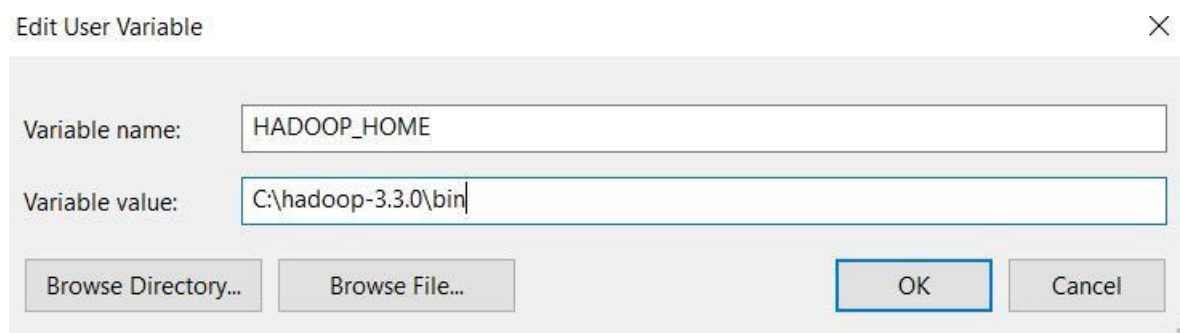
**STEP 11:** After doing the extraction process there is another compressed file with in the extractedfile extract that as well.



**STEP 12:** From the extracted folder replace the bin file with the reliable windows supportedconfigured file here is the drive link to download the bin file

<https://drive.google.com/file/d/1kVhX9snOZ3oLUxDjh3AVI8fcRnEWAAE4/view>

**STEP 13:** Setup the Hadoop environment in environment variable and set path location as **HADOOP\_HOME= "C:\hadoop-3.3.0\bin"**



**STEP 14:** Add the Hadoop bin and sbin path location by editing the path. And add the bin, sbinlocation there



**STEP 15:** Now open etc folder inside the Hadoop folder and locate the file Hadoop-env.cmd and set the java home location

```
22 @rem remote nodes.
23
24 @rem The java implementation to use. Required.
25 set JAVA_HOME=%JAVA_HOME%
26 set JAVA_HOME=C:\java\java8
27
28 @rem The jsvc implementation to use. Jsvc is required to run secure datanodes.
29 @rem set JSVC_HOME=%JSVC_HOME%
30
31 @rem set HADOOP_CONF_DIR=
```

**STEP 16:** Edit the following configuration XML files core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml are used to configure the behaviour of your Hadoop Cluster and save them.

**STEP 17:** Starting from core-site.xml edit it using notepad++ and following the program to configure.

#### **PROGRAM (CORE-SITE.XML)**

```
<configuration>

<property>

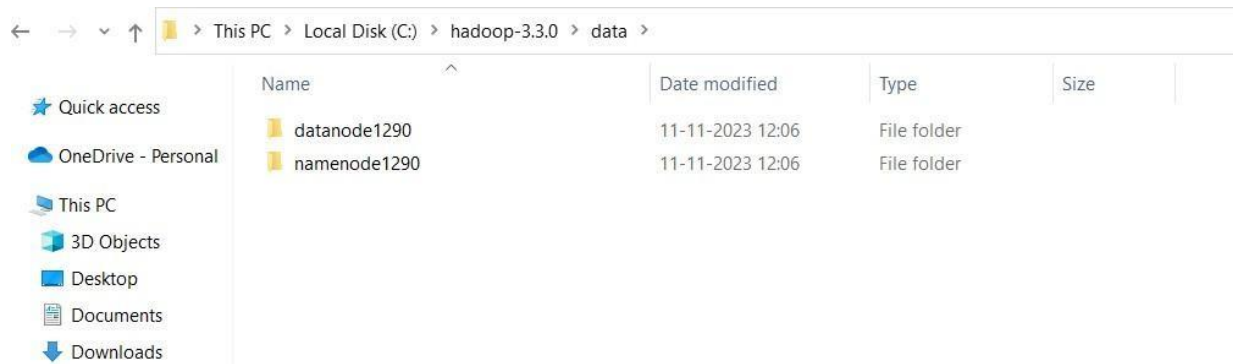
<name>fs.default.name</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>
```

**STEP 18:** To edit the hdfs-site.xml create Data folder and then within the Data folder create Namenode, Datenode which are used to manage and cluster flow date and log files.



**STEP 19:** Now open the hdfs-site.xml in notepad++ and add the following program

```
<configuration
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value> C:/hadoop-3.3.0/Data/datanode1290 </value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>> C:/hadoop-3.3.0/Data/namenode1290 </value>
</property>
</configuration>
```

**STEP 20:** Edit mapred-site.xml file using notepad++ add this following program

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



**STEP 21:** Edit the yarn-site.xml with following program and save it.

```
<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>
```

**STEP 22:** Now save them and open command prompt as administrator and run the following command to **hdfs namenode -format** to format the contents of namenode

```
\current\fsimage.ckpt_000000000000000000 using no compression
2023-11-06 11:45:46,760 INFO namenode.FSImageFormatProtobuf: Image file C:\hadoop-3.3.0\data\namenode1290\current\fsimage.ckpt_000000000000000000 of size 400 bytes saved in 0 seconds .
2023-11-06 11:45:46,783 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2023-11-06 11:45:46,794 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2023-11-06 11:45:46,795 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at Azhar/127.0.0.1
*****/
C:\Windows\system32>
```

**STEP 23:** To check the daemons configured correctly open command prompt as administrator and run the following command“s

**hdfs namenode, hdfs Datanode, yarn nodemanager, yarn resourcemanager**  
**hdfs namenode**

```
C:\hadoop-3.3.0\sbin>hdfs namenode
2023-11-06 11:47:10,614 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG:  host = Azhar/127.0.0.1
STARTUP_MSG:  args = []
STARTUP_MSG:  version = 3.3.0
STARTUP_MSG:  classpath = C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\share\hadoop\common;C:\hadoop-3.3.0\share\hadoop\common\lib\accessors-smart-1.2.jar;C:\hadoop-3.3.0\share\hadoop\common\lib\animal-sniffer-annotations-1.1
```

## hdfs datanode

```
C:\Windows\system32>hdfs datanode
2023-11-06 11:51:47,958 INFO datanode.DataNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting DataNode
STARTUP_MSG:  host = Azhar/127.0.0.1
STARTUP_MSG:  args = []
STARTUP_MSG:  version = 3.3.0
STARTUP_MSG:  classpath = C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\share\hadoop\common;C:\hadoop-3.3.0\share\hadoop\common\lib\accessors-smart-1.2.jar;C:\hadoop-3.3.0\share\hadoop\common\lib\animal-sniffer-annotations-1.17.jar;C:\hadoop-3.3.0\share\hadoop\common\lib\asm-5.0.4.jar;C:\hadoop-3.3.0\share\
```

## yarn nodemanager

```
C:\Windows\system32>yarn nodemanager
2023-11-06 11:52:28,180 INFO nodemanager.NodeManager: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NodeManager
STARTUP_MSG:  host = Azhar/127.0.0.1
STARTUP_MSG:  args = []
STARTUP_MSG:  version = 3.3.0
STARTUP_MSG:  classpath = C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\share\hadoop\common;C:\hadoop-3.3.0\share\hadoop\common\lib\accessors-smart-1.2.jar;C:\hadoop-3.3.0\share\hadoop\common\lib\animal-sniffer-annotations-1.17.jar;C:\hadoop-3.3.0\share\
```

## yarn resourcemanager

```
C:\Windows\system32>yarn resourcemanager
2023-11-06 11:53:32,487 INFO resourcemanager.ResourceManager: STARTUP_MSG:
/*****
STARTUP_MSG: Starting ResourceManager
STARTUP_MSG:  host = Azhar/127.0.0.1
STARTUP_MSG:  args = []
STARTUP_MSG:  version = 3.3.0
STARTUP_MSG:  classpath = C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\etc\hadoop;C:\hadoop-3.3.0\share\hadoop\common;C:\hadoop-3.3.0\share\hadoop\common\lib\accessors-smart-1.2.jar;C:\hadoop-3.3.0\share\hadoop\common\lib\animal-sniffer-annotations-1.17.jar;C:\hadoop-3.3.0\share\
```

**STEP 24:** To check the daemons that are running in background we can use Java Virtual Machine Process Status which is used to list the java virtual machines that are currently running on a system it is used to display the process ID(PID) of each JVM

```
C:\Windows\System32>jps
9712 NodeManager
8212 Jps
16056 NameNode
1800 ResourceManager
18200 DataNode
```



**STEP 25:** Now we can access the Namenode and Datanode as web user interface (web-UI) by using the following localhost address

**localhost:9870**



**STEP 26:** To access the Datanode use the following localhost address

**localhost:8088**



## Result:

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