NAME :

REGISTER NO:

III YEAR / VI SEMESTER

20CSPL602-CLOUDCOMPUTINGLABORATORY

DEPARTMENT OF
M.TECH
COMPUTER SCIENCE AND ENGINEERING
(5 YEAR INTEGRATED)
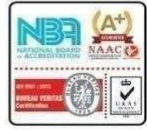
ACADEMIC YEAR: 2024 - 2025

Sri
SAI RAM ENGINEERING COLLEGE
An Autonomous Institution | Affiliated to Anna University & Approved by AICTE, New Delhi
Accredited by **NBA** and **NAAC "A+"** | An **ISO** 9001:2015 Certified and MHRD **NIRF** ranked institution
Sai Leo Nagar, West Tambaram, Chennai - 600 044. www.sairam.edu.in
Founder Chairman : MJF. Ln. Leo Muthu

Bonafide Certificate

Register No.:

This is to certify the Bonafide record of work done by

Mr./ Ms. _____ of **M.Tech Computer**

Science and Engineering (5 Year Integrated) in the **20CSPL602 – Cloud**

Computing Laboratory during the academic year 2024-2025

Station   : Chennai –600 044

Date   :

STAFF IN-CHARGE   **HEAD OF THE DEPARTMENT**

Submitted for End Semester Practical Examination held on_____ _____ at
Sri Sai Ram Engineering College, Chennai – 600 044.

INTERNAL EXAMINER   **EXTERNAL EXAMINER**

<u>INDEX</u>

Ex. No.: 1   **INSTALL VIRTUALBOX WITH DIFFERENT FLAVORS OF LINUX
                        ON TOP OF WINDOWS 7**

Date:

Introduction

Virtualization:

Virtualization is the creation of virtual servers, infrastructures, devices and computing resources.

Virtualization changes the hardware-software relations and is one of the foundational elements of cloud computing technology that helps utilize the capabilities of cloud computing to the full.

Virtualization techniques allow companies to turn virtual their networks, storage, servers, data, desktops and applications.

Hypervisor or Virtual Machine Monitor (VMM)

A hypervisor or virtual machine monitor (VMM) is a piece of computer software, firmware or hardware that creates and runs virtual machines. A computer on which a hypervisor is running one or more virtual machines is defined as a host machine. Each virtual machine is called a guest machine. The hypervisor presents the guest operating systems with a virtual operating platform and manages the execution of the guest operating systems. Multiple instances of a variety of operating systems may share the virtualized hardware resources.

Types of Virtualization

Operating-system-level virtualization - is a server-virtualization method where the kernel of an operating system allows for multiple isolated user- space instances, instead of just one. Such instances (sometimes called containers, software containers,[1] virtualization engines (VE), virtual private servers (VPS), or jails) may look and feel like a realserver from the point of view of its owners and users Platform / Hardware virtualization -Hardware virtualization or platform

virtualization refers to the creation of a virtual machine that acts like a real computer with an operating system. Software executed on these virtual machines is separated from the underlying hardware resources. For example, a computer that is running Microsoft Windows may host a virtual machine that looks like a computer with the Ubuntu Linux operating system; Ubuntu-based software can be run on the virtual machine.

In hardware virtualization, the host machine is the actual machine on which the virtualization takes place, and the guest machine is the virtual machine. The words host and guest are used to distinguish the software that runs on the physical machine from the software that runs on the virtual machine. Different types of hardware virtualization include:

o Full virtualization: Almost complete simulation of the actual hardware to allow software, which typically consists of a guest operating system, to run unmodified.

o Partial virtualization: Some but not all of the target environment is simulated. Some guest programs, therefore, may need modifications to run in this virtual environment.

o Para virtualization: A hardware environment is not simulated; however, the guest programs are executed in their own isolated domains, as if they are running on a separate system.

Application virtualization is software technology that encapsulates computer programs from the underlying operating system on which it is executed. A fully virtualized application is not installed in the traditional sense, although it is still executed as if it were.

Oracle Virtualbox

o VirtualBox is a general-purpose full virtualizer for x86 hardware, targeted at server, desktop and embedded use.Each virtual machine can execute its own operating system, including versions of Microsoft Windows, Linux, BSD, and MS-DOS. VMware Workstation is developed and sold by VMware, Inc., a division ofEMCCorporation

Ubuntu

o Ubuntu is an operating system like any other and it is free & open source. It means that we can

download it freely and install it on as many computers as we like. By the term open source it means that we can actually see its code. To provide a more secure environment, the —SUDO‖ tool is used to assign temporary privileges for performing administrative tasks. Ubuntu comes installed with a wide range of Software that includes Libre Office ,Firefox, Thunderbird

AIM:

To install Virtualbox with different flavors of linux or windows OS on top of windows host operating system.

PROCEDURE:

Steps to install VirtualBox:

1. Download VirtualBox installer for windows.

2. The installer can be downloaded from the link https://www.virtualbox.org/wiki/Downloads

3. Click "Windows host" to download the binary version for windows host.

4. The installer file downloaded will have the file name format like VirtualBox VersionNumber-BuildNumber-Win.exe. Example: VirtualBox-6.1.12-139181- Win.exe.

5. Double click on the installer to launchthesetupWizard. Click onNext to continue.

6. Custom setup dialog box will be opened. Accept the default settings and clicknext.

7. Select the way you want the features to be installed. You can accept the default and clicknext.

8. A dialog box opens with Network Interfaces warning. Click Yes to proceed.

9. Click install to begin the installation process.

10. When prompted with a message to install (Trust) Oracle Universal Serial Bus, click Install to continue.

11. After the installation completes, click finish to exit the setup wizard.12. Launch the Oracle VMVirtualBox.

Steps to create a virtual machine [Ubuntu] in VirtualBox:

1. Open the Oracle VMVirtualBox.

2. Click the Newiconor,,Ctrl+N"to create a new virtual machine.
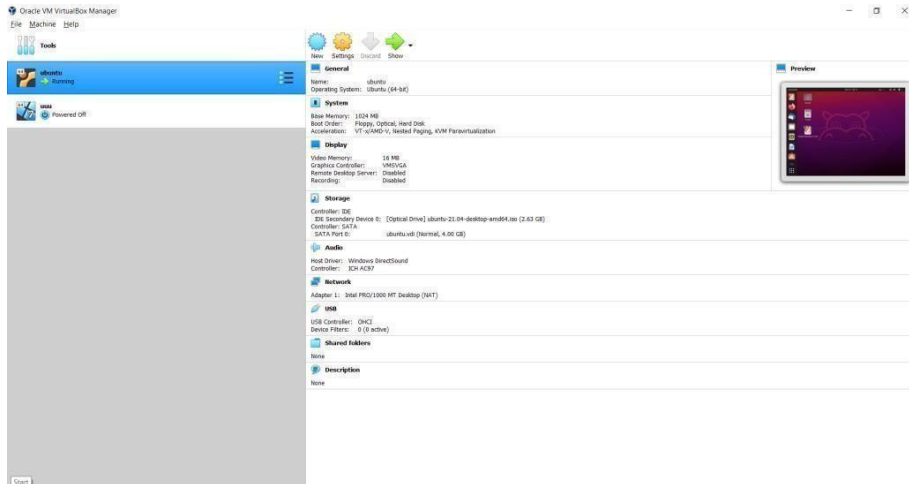
3. Enter a name for the new virtual machine. Choose the Type and Version. Note that VirtualBox automatically changes 'Type' to Linux and 'Version' to 'Ubuntu (64 bit)' if the name is given as „Ubuntu". ClickNext.
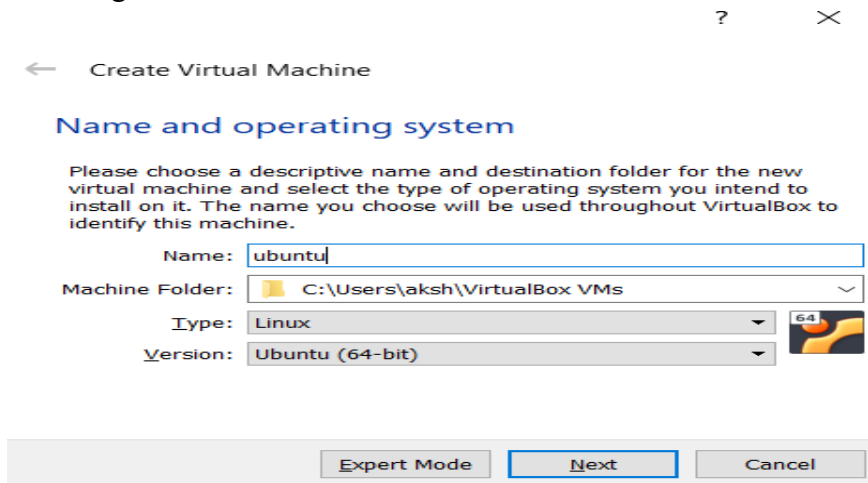
4. Select the amount of RAM to use(eg: 1000 MB). The ideal amount of RAM will automatically be selected. Do not increase the RAM into the red section of the slider; keep the slider in the greensection.

5. Accept the default 'Create a virtual hard drive now' and click the 'Create' button.

6. Choose the hard disk filetypeasVMDK(VirtualBoxDisk).ClickNext.

7. Click Next to accept the default option„ Dynamically allocated" for storage on a physical harddrive.

8. Select the size of the virtual hard disk(eg:8.0 GB) and click create.

9. The newly created virtual machine will be displayed in the dashboard.

10. Download the ISO file [Ubuntu disk image file]. Latest version of Ubuntu iso file can be downloaded from the link https://ubuntu.com/download/desktop.

11. For previous versions, gotohttp://releases.ubuntu.com. Choose the preferred version of Ubuntu and download the isofile.
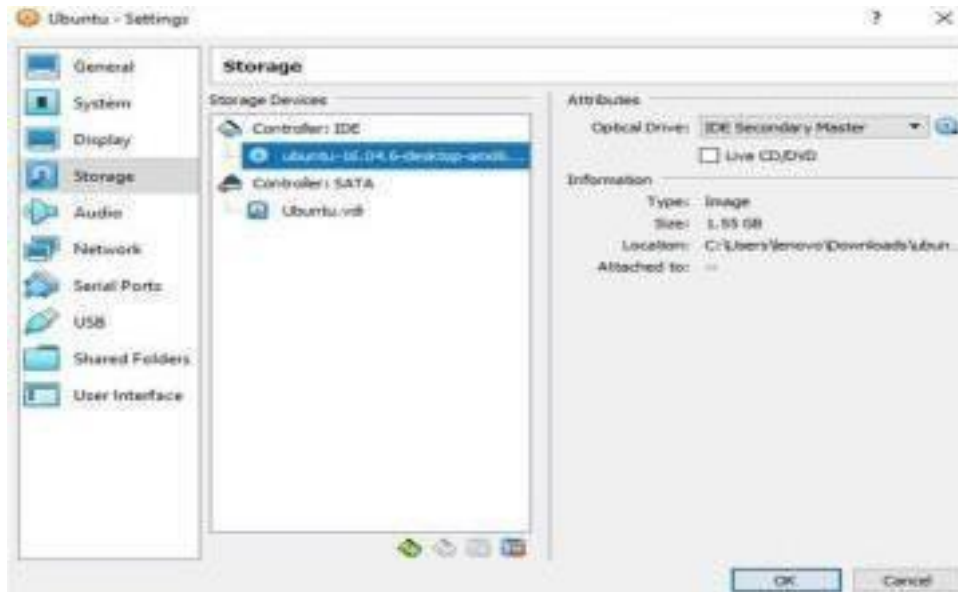

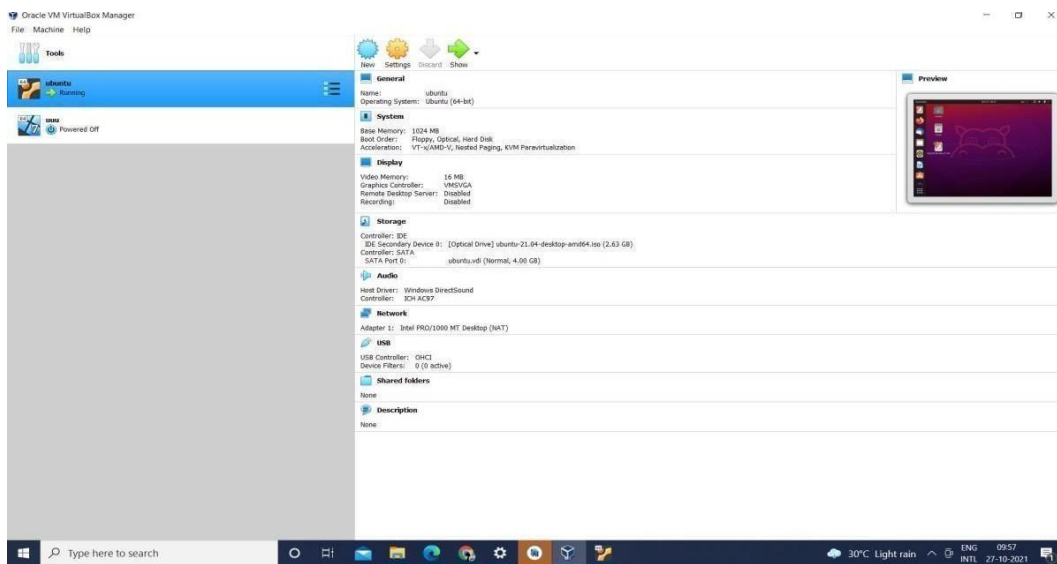12. To setup the Ubuntu disk image file (iso file) goto settings.

13. Click Storage. Under „Storage Devices" section click„Empty".

14. In the Attributes section, click the disk image and then "Choose Virtual Optical Disk File".
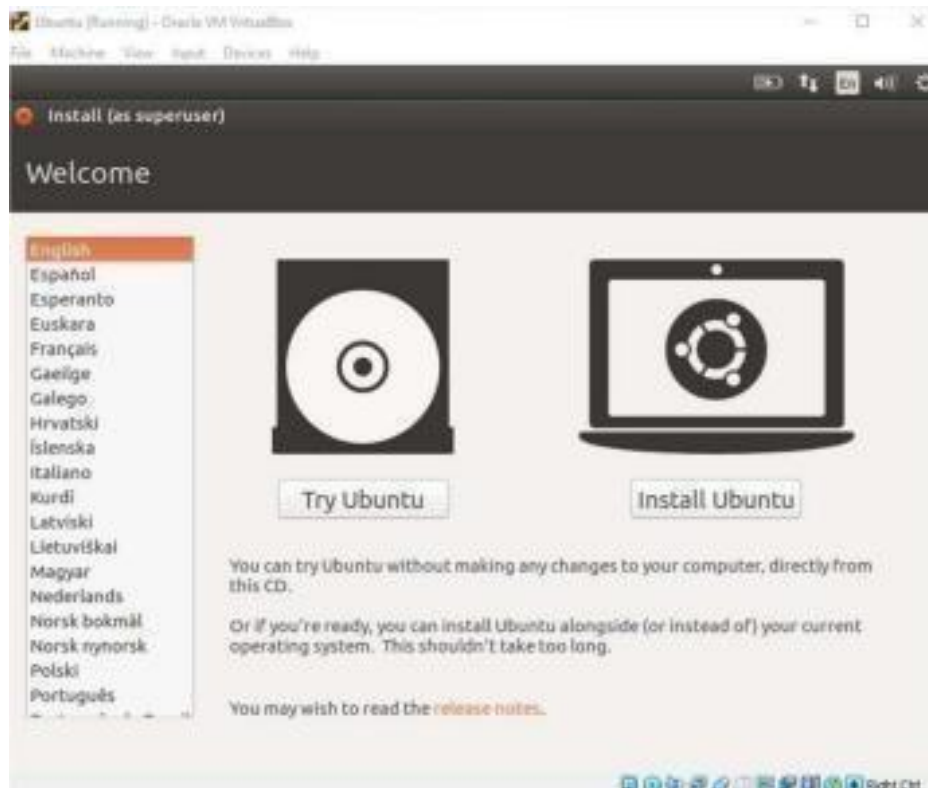
15. Browse and select the downloaded iso file. Clickok.

16. Select the newly created virtual machine in the dashboard and click the start button.

17. In the welcome screen, click the "Install Ubuntu" button.

18. Click 'Continue' button.

19. Make sure 'Erase disk and install Ubuntu' option is selected and click 'Install Now' button.

20. Choose the default and click continue.

21. Setup up your profile by creating username and password.

22. After installation is complete, click 'Restart Now' button and follow the instructions.

23. The Ubuntu OS is ready to use. Login with the username and password.

20. Choose the default and click continue.

21. Setup up your profile by creating username and password.

22. After installation is complete, click 'Restart Now' button and follow the instructions.

23. The Ubuntu OS is ready to use. Login with the username andpassword.

RESULT:

The Virtualbox installation is completed and the Virtual machine is created on top of the Windows host operating system.

Ex. No.: 2

Date:

INSTALL A C COMPILER IN THE VIRTUAL MACHINE CREATED USING VIRTUAL BOX AND EXECUTE SIMPLE MACHINE
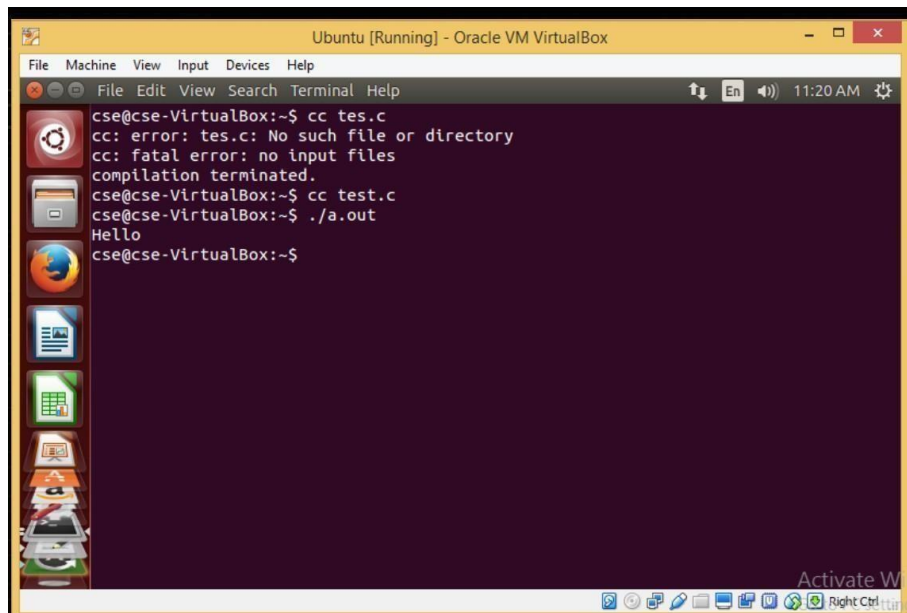
AIM:

To create and run a sample C program in Ubuntu.

PROCEDURE:

1. Open Virtual Box and run the Ubuntu virtual machine.
2. Open a terminal in Ubuntu virtual machine
3. Create a sample C program {
   cat>>myprogram.c to create a new program. Stop the command after typing the program as ctrl+Z
   Compile: gcc myprogram.c (or) cc myprogram
   Run: ./a.out

   [or]

   type vi; window appears;
   Insert the test by typing i
   After typing test type as :wq

4. Compile the C file with cc command
5. Run the C file with ./a.out command

OUTPUT :
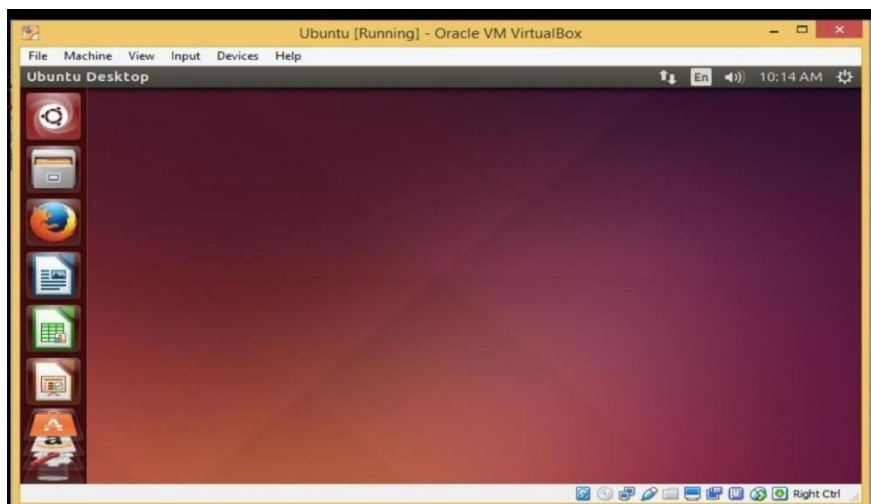
RESULT:

Thus the execution of sample C program using Ubuntu virtual machine is successfully completed

Ex. No.: 3   **INSTALL GOOGLE APP ENGINE. CREATE A HELLO WORLD APP**
AND OTHER SIMPLE WEB APPLICATIONS USING PYTHON/JAVA

Date:

Aim:

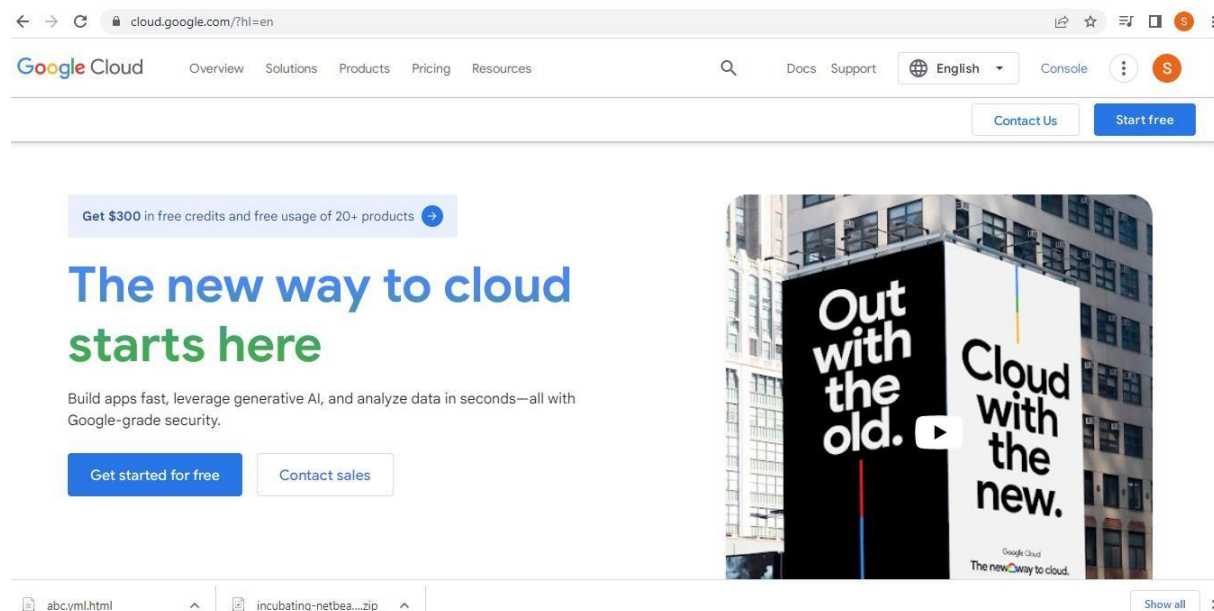To install Google App Engine. Create a hello world app and other simple web applications using Python or Java

Procedure

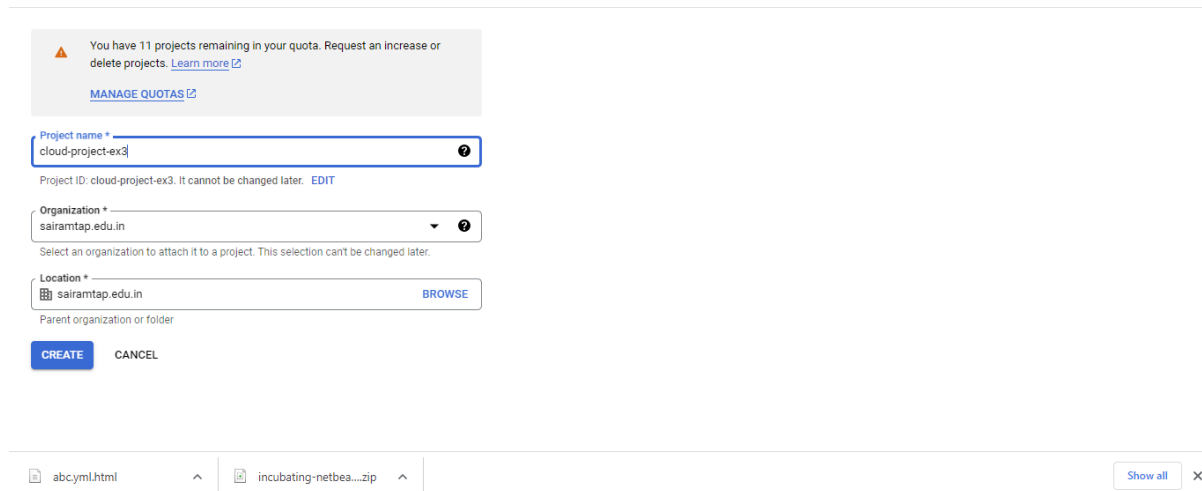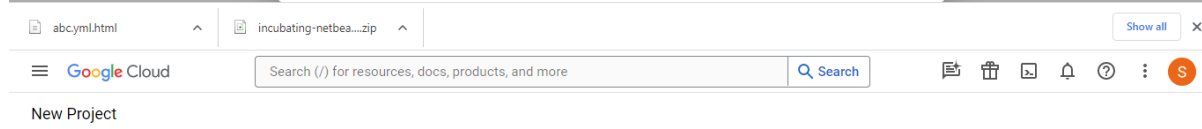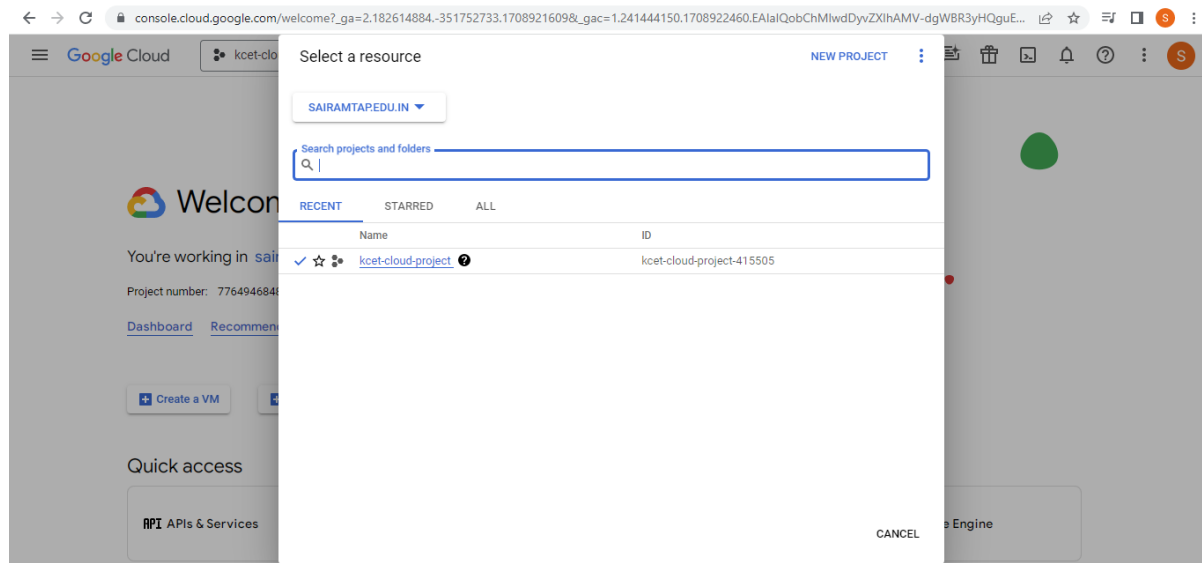Step1: Login to www.cloud.google.com

Step2 : Click Console

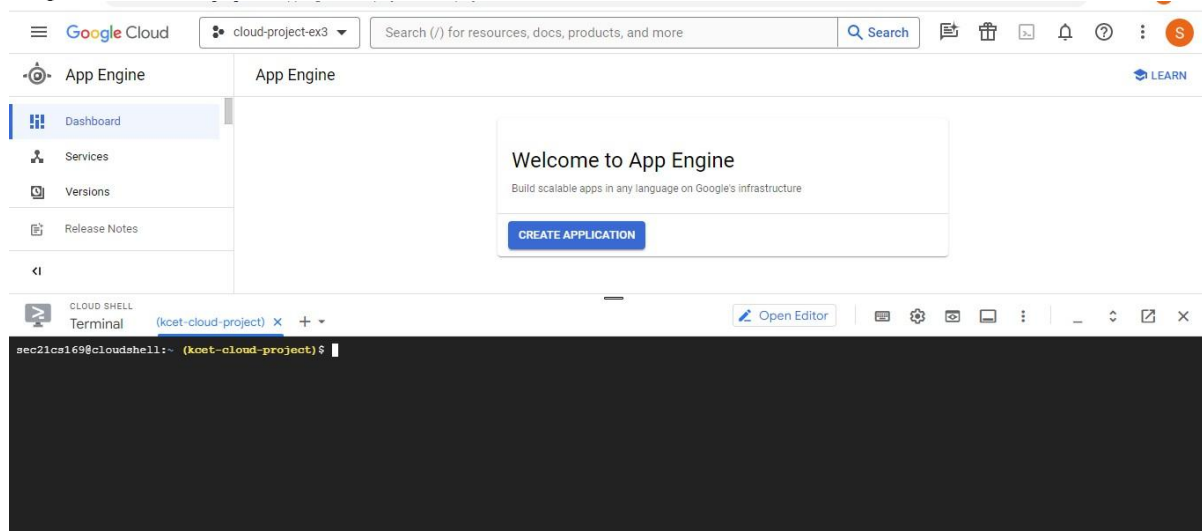Step 3 : Google Cloud Platform is shown

Step 4 : Click Dashboard in the Google cloud Plaform

Step 5 : Dashboard in the Google Cloud Plaform

Step 6 : Click New Project and give unique Project Name.
Example : kcet-cloud-project

☰ **Google** Cloud   :• kcet-clo   **Select a resource**   **NEW PROJECT**   ⋮

SAIRAMTAP.EDU.IN ▼

Search projects and folders
🔍 |

RECENT   STARRED   ALL

	Name	ID
✓ ☆ :•	kcet-cloud-project ❓	kcet-cloud-project-415505

🟢

You're working in sair

Project number: 77649468

Dashboard   Recommend

➕ Create a VM   ➕

Quick access

API APIs & Services   CANCEL   e Engine

📄 abc.yml.html   ⌃   📄 incubating-netbea....zip   ⌃   Show all   ✕

☰ **Google** Cloud   [Search (/) for resources, docs, products, and more]   🔍 Search   📥 🎁 🗔 🔔 ❓ ⋮ S

New Project

⚠ You have **11 projects** remaining in your quota. Request an increase or delete projects. Learn more ↗

MANAGE QUOTAS ↗

Project name *
cloud-project-ex3   ❓

Project ID: cloud-project-ex3. It cannot be changed later. EDIT

Organization *
sairamtap.edu.in   ▼   ❓

Select an organization to attach it to a project. This selection can't be changed later.

Location *
🏢 sairamtap.edu.in   BROWSE

Parent organization or folder

CREATE   CANCEL

📄 abc.yml.html   ⌃   📄 incubating-netbea....zip   ⌃   Show all   ✕

Step 7: Go to cloud shell

☰ **Google** Cloud   :• cloud-project-ex3 ▼   [Search (/) for resources, docs, products, and more]   🔍 Search   📥 🎁 🗔 🔔 ❓ ⋮ S

⊙ App Engine | App Engine   ☁ LEARN

▦ Dashboard
⌁ Services
◎ Versions
🗐 Release Notes

◁   **Welcome to App Engine**
                               Build scalable apps in any language on Google's infrastructure

                               CREATE APPLICATION

⟫ CLOUD SHELL
   Terminal   (kcet-cloud-project) ✕   + ▾   ✎ Open Editor   ▦ ⚙ ▣ ▢ ⋮ _ ↕ ⬈ ✕

sec21cs169@cloudshell:~ (kcet-cloud-project)$ ▮

Step 13 : Create a Directory PythonProject using mkdir command
Syntax : mkdir PythonProject

Step 14 : Click Editor to create Python application

```
sec21cs169@cloudshell:~ (kcet-cloud-project)$ mkdir PythonProject
sec21cs169@cloudshell:~ (kcet-cloud-project)$ ▯
```

Step 15 : Create main.py file with the following program

```
🐍 main.py    ✕    ! app.yml        ≡ requirements.txt

home > sec21cs100 > 🐍 main.py > ...
    1    import logging
    2    from flask import Flask
    3    app =Flask(__name__)
    4    @app.route('/')
    5    def hello():
    6        return 'Hello World'
    7    if __name__ == '__main__':
    8        app.run(host='127.0.0.1',port=8080, debug=True)
```

Step 16 : Create app.yml file with the following program

```
🐍 main.py    ✕    ! app.yml    ✕    ≡ requirements.txt

home > sec21cs100 > ! app.yml > ...
    1    runtime: pythonenv: flex
    2    entrypoint: gunicorn -b :$PORT main:app
    3    runtime_config: python_version: 3
```

Step 17 : Create requirements.txt file with the following text

```
🐍 main.py        ! app.yml        ≡ requirements.txt ✕

home > sec21cs100 > ≡ requirements.txt
    1    Flask==0.11.1
    2    gunicorn==19.6.0
```

Step 15 : Run the programs in terminal

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to macro-spider-415505.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
sec21cs100@cloudshell:~ (macro-spider-415505)$ flask --app main run
 * Serving Flask app 'main'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
^Csec21cs100@cloudshell:~ (macro-spider-415505)$ pythonmain.py
-bash: pythonmain.py: command not found
sec21cs100@cloudshell:~ (macro-spider-415505)$ python main.py
 * Serving Flask app 'main'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:8081
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 111-921-772
```

Step 15 : preview the output using the preview button

```
← → C  🔒 8080-cs-070d3e0a-98a7-4f05-bd80-65ff400929bb.cs-asia-southeast1-ajrg.cloudshell.dev/?authuser=0&redirectedPreviously=true
```

Hello World

Result

Thus the installation of Google App Engine, create a hello world app and other simple web applications using Python

Ex. No.: 4   Use GAE Launcher to launch the web applications
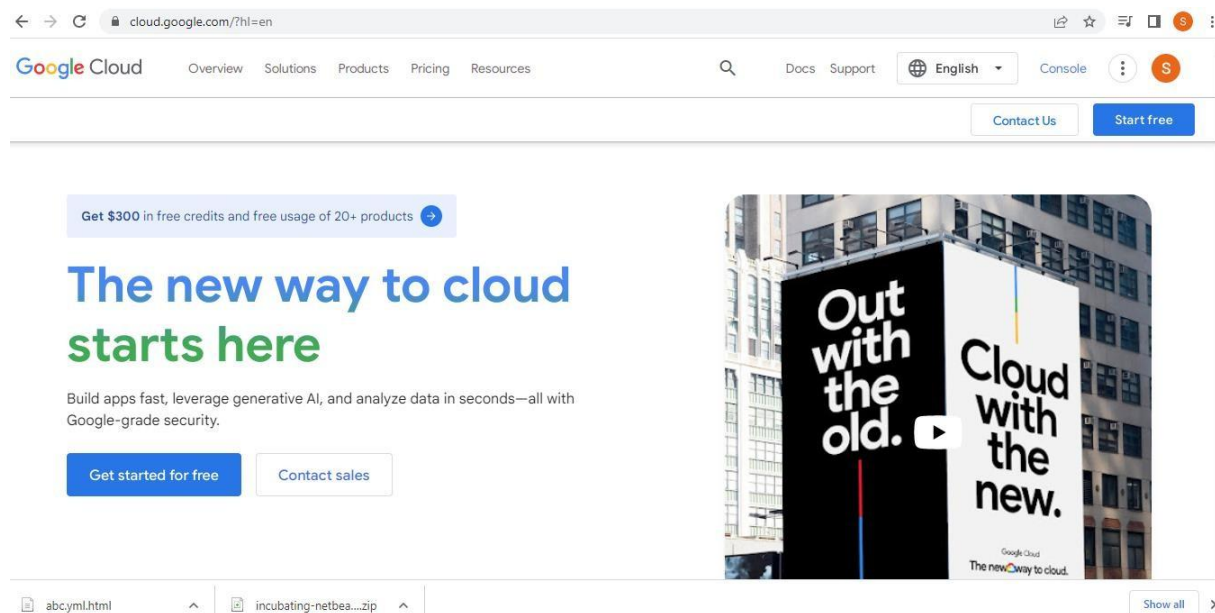
Date:

Aim:

To create a Google App Engine launcher to launch web applications

Procedure

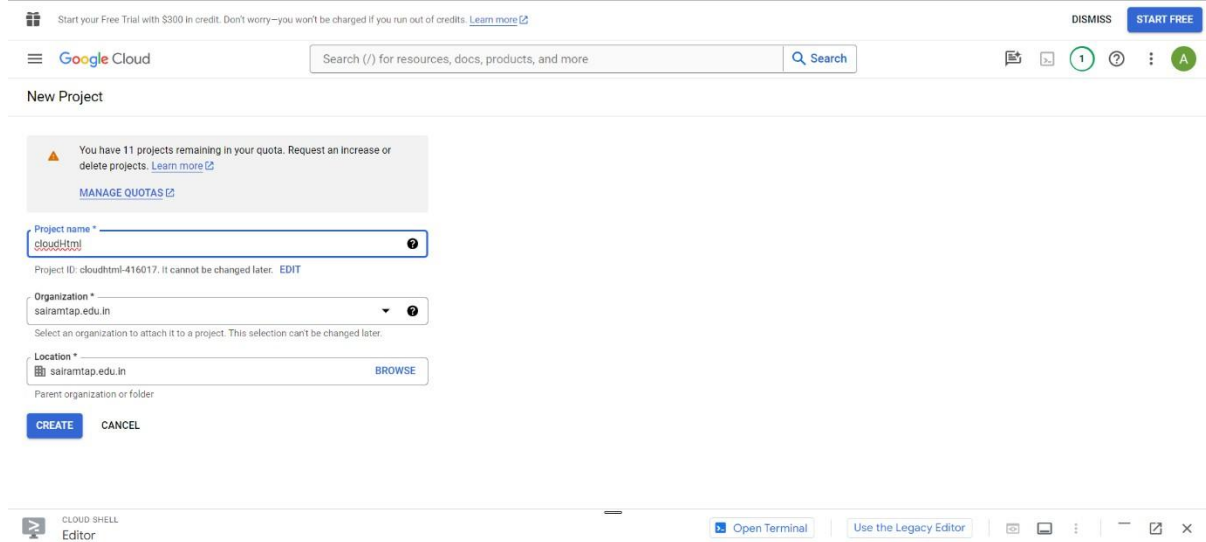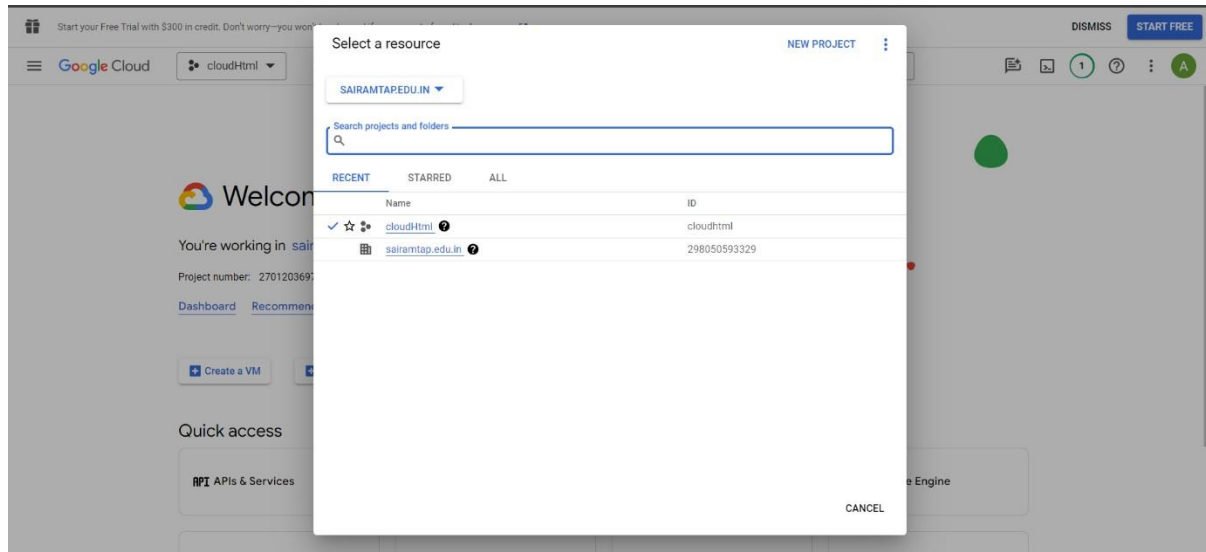Step1 : Login to www.cloud.google.com

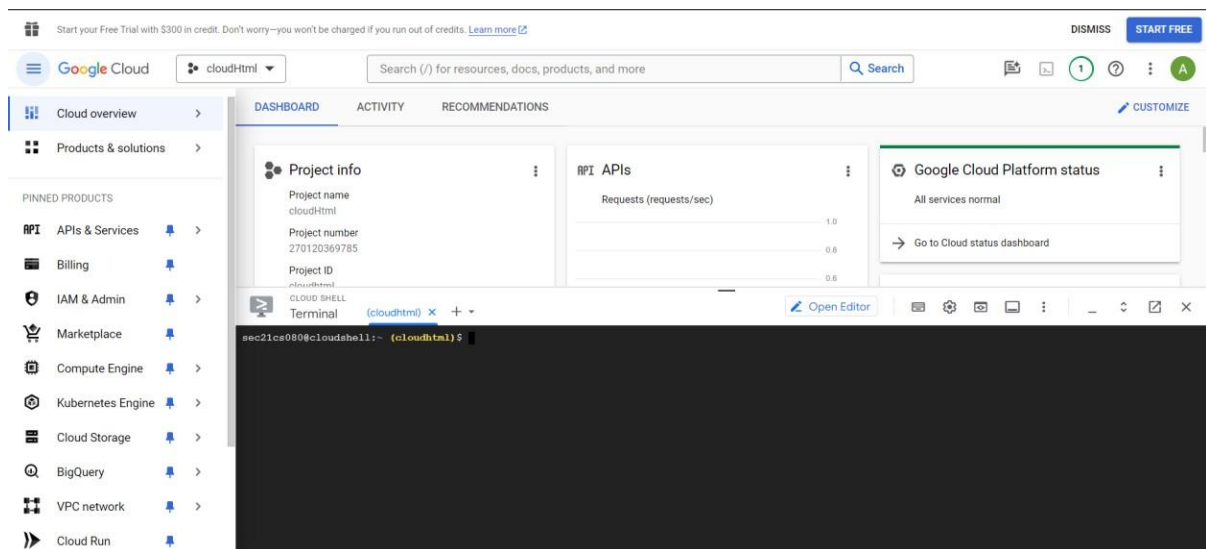Step2 : Goto Console

Step 3 : Google Cloud Platform is shown

Step 4 : Click Dashboard in the Google Cloud Plaform

Step 5 : Click New Project and give unique Project Name.
Example : cloudHtml

Step 6: Go to cloud shell

Step 7 : Create a Directory www using mkdir command

Syntax : mkdirwww

```
sec21cs080@cloudshell:~ (cloudhtml)$ mkdir www
sec21cs080@cloudshell:~ (cloudhtml)$ cd www
sec21cs080@cloudshell:~/www (cloudhtml)$ █
```

Step 8 :And create the files and folder in the following manner

```
sec21cs080@cloudshell:~ (cloudhtml)$ mkdir www
sec21cs080@cloudshell:~ (cloudhtml)$ cd www
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir css
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir images
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir js
sec21cs080@cloudshell:~/www (cloudhtml)$ touch index.html
sec21cs080@cloudshell:~/www (cloudhtml)$ █
```

Step 9: Click Editor to create Web page using HTML

Step 10 :Updateindex.html file with the following program

```html
<> index.html ×

www > <> index.html > ⊘ html
1    <!DOCTYPE html>
2  ∨ <html lang="en">
3  ∨ <head>
4        <meta charset="UTF-8">
5        <meta name="viewport" content="width=device-width, initial-scale=1.0">
6        <title>Document</title>
7    </head>
8  ∨ <body>
9        <p>Hello World!!!</p>
10   </body>
11   </html>
```

Step 11 : Run the programs in terminal

```
sec21cs080@cloudshell:~ (cloudhtml)$ mkdir www
sec21cs080@cloudshell:~ (cloudhtml)$ cd www
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir css
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir images
sec21cs080@cloudshell:~/www (cloudhtml)$ mkdir js
sec21cs080@cloudshell:~/www (cloudhtml)$ touch index.html
sec21cs080@cloudshell:~/www (cloudhtml)$ python -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
127.0.0.1 - - [02/Mar/2024 17:15:34] "GET /?authuser=1 HTTP/1.1" 200 -
█
```

Step 12 : Preview the output using the preview button
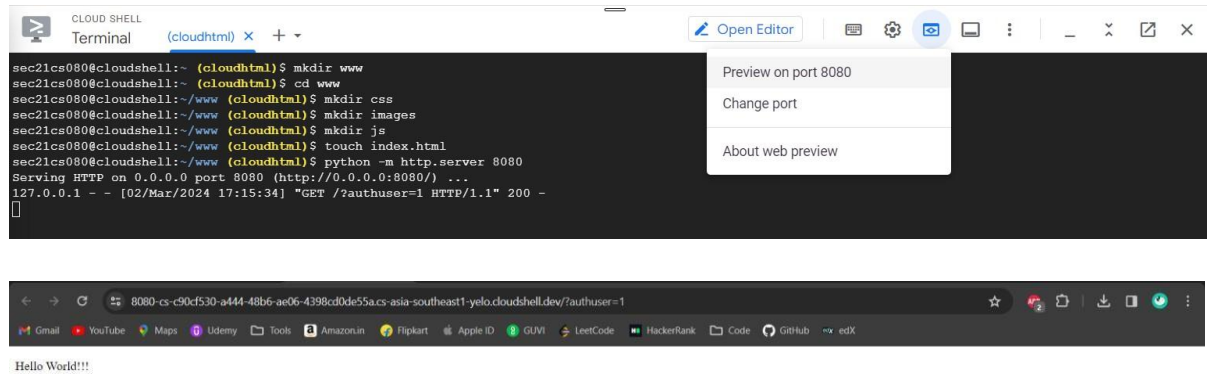
Result

Thus the creation of Google App Engine launcher to launch web applications has been dine successfully and its output is verified.

**Ex. No.: 5**   **SIMULATE A CLOUD SCENARIO WITH CLOUDSIM AND RUN A SCHEDULING ALGORITHM**

Date:

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

PROCEDURE
1. Download NetBeans and Install it.
2. Download CloudSim and unzip it.
3. Once Netbeans IDE is installed, start creating a new Project via *File > New Project*. Choose "Java" under Categories and "Java Application" under Projects.
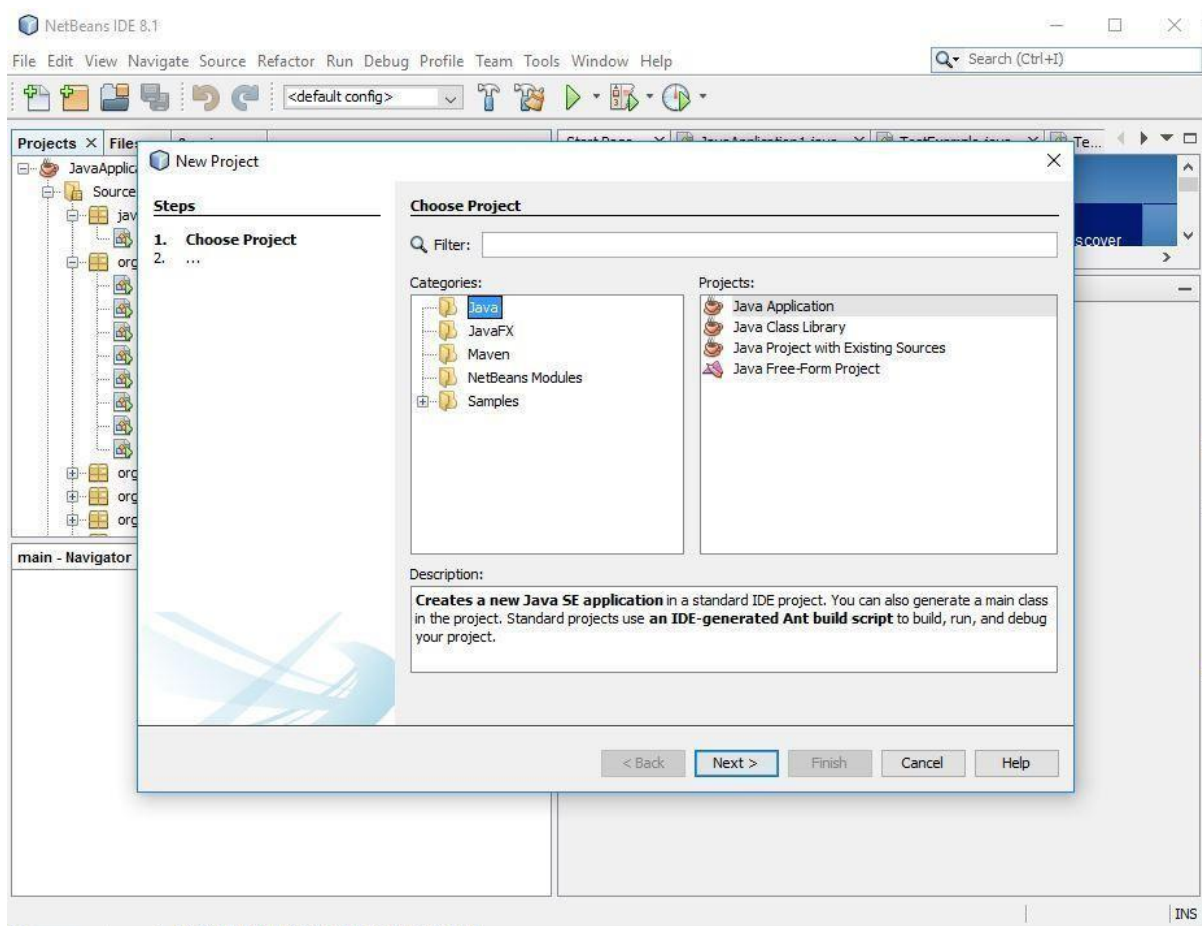
4. Name your project and click "Finish" button.
   On the left sidebar, right click on the project and select "Properties" as shown below:
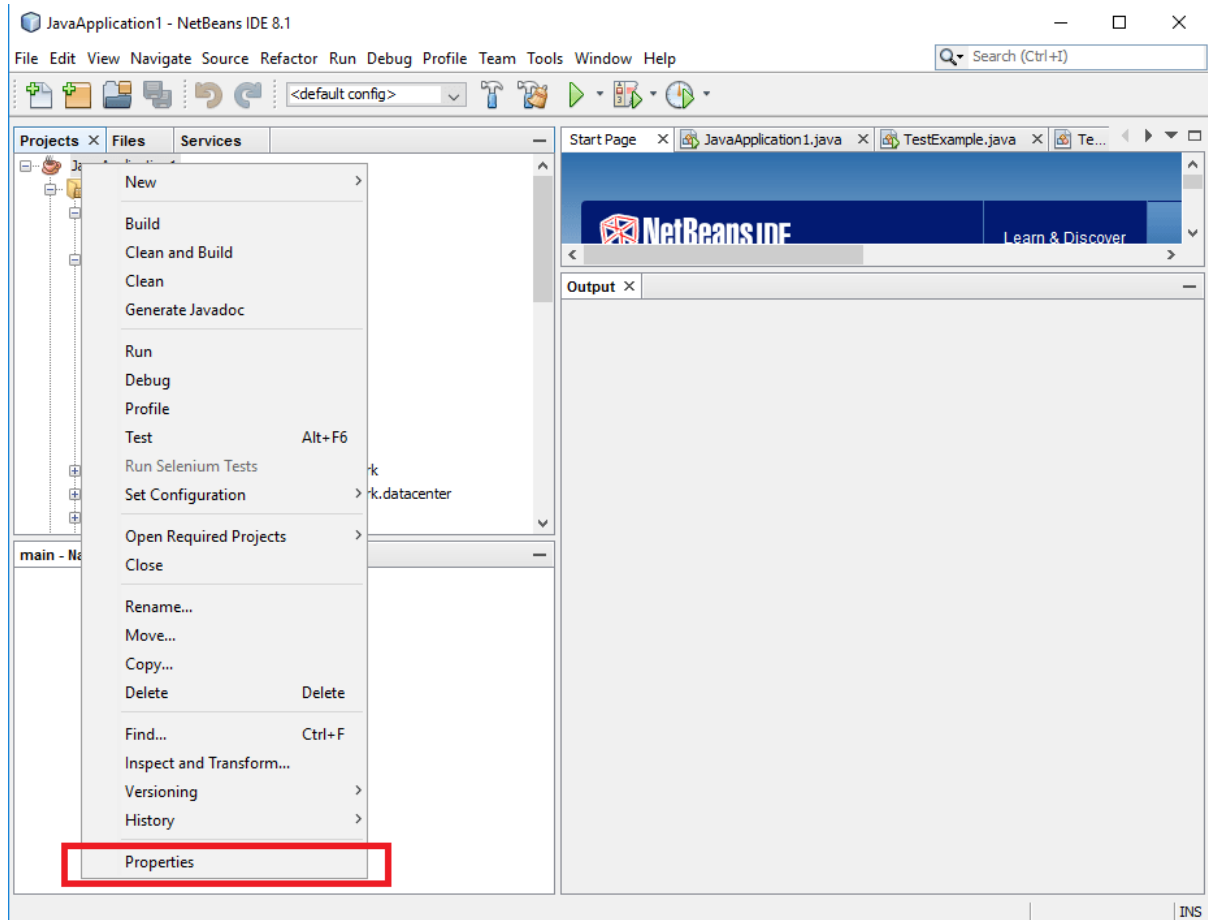
5. In the properties dialog, select "Libraries" and click on "Add Library" button at the right. In the "Add Library" dialog, select cloudsim jar file (will be located in jars folder) downloaded in step 2. Now you can drag and drop *example programs (under examples/org/cloudbus/cloudsim)* from the CloudSim directory into your application. That's it, your CloudSim toolkit it ready to use.
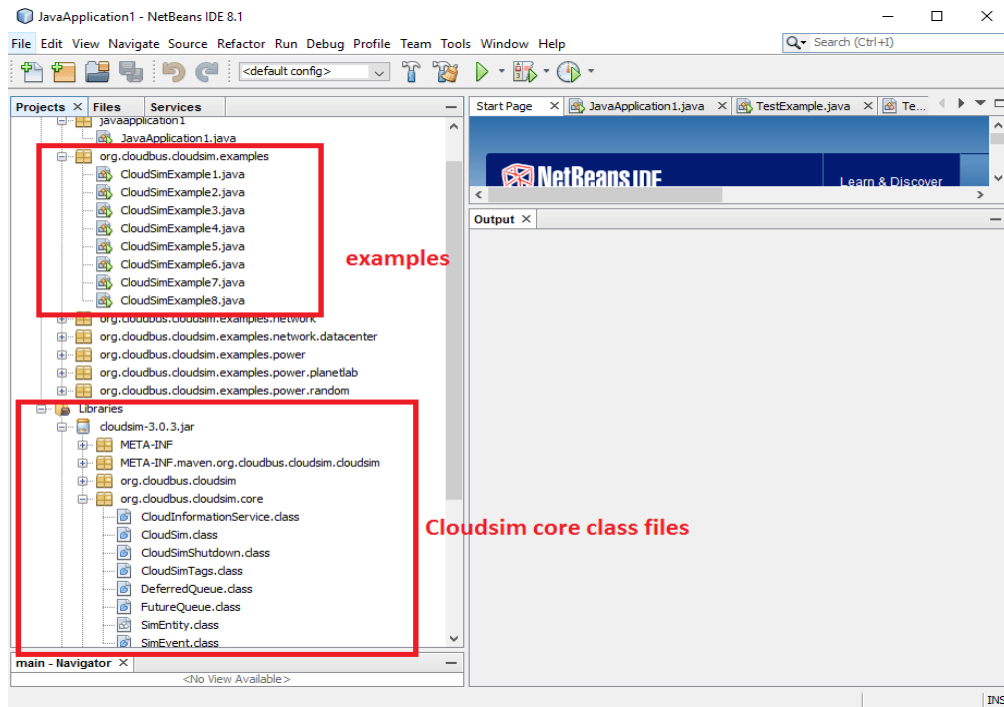
6. Expand *org.cloudbus.cloudsim.examples* folder to find all example programs. Select your desired example and start running the simulation. The output will be displayed at the bottom pane as shown below:
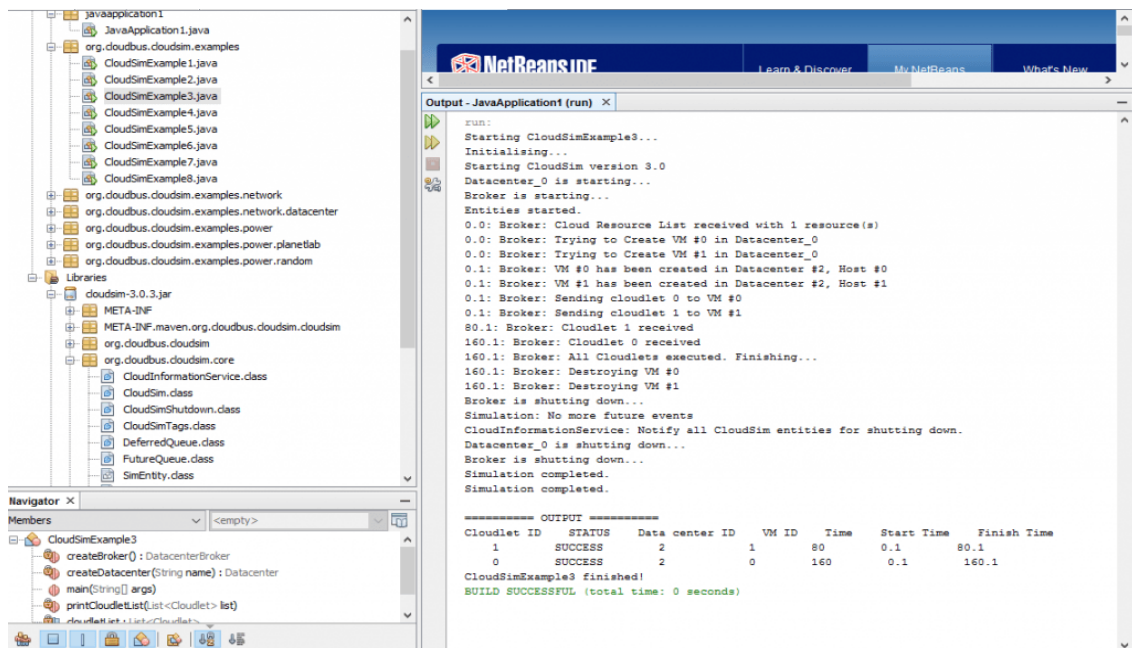
RESULT:

Thus a Cloud Scenario using CloudSim has been generated and a scheduling algorithm that was not present on it has been run successfully

**Ex. No.: 6**   **Find a procedure to transfer files from one virtual machine to another virtual machine**

Date:

AIM:

To install Virtualbox with different flavors of linux or windows OS on top of windows host operating system.
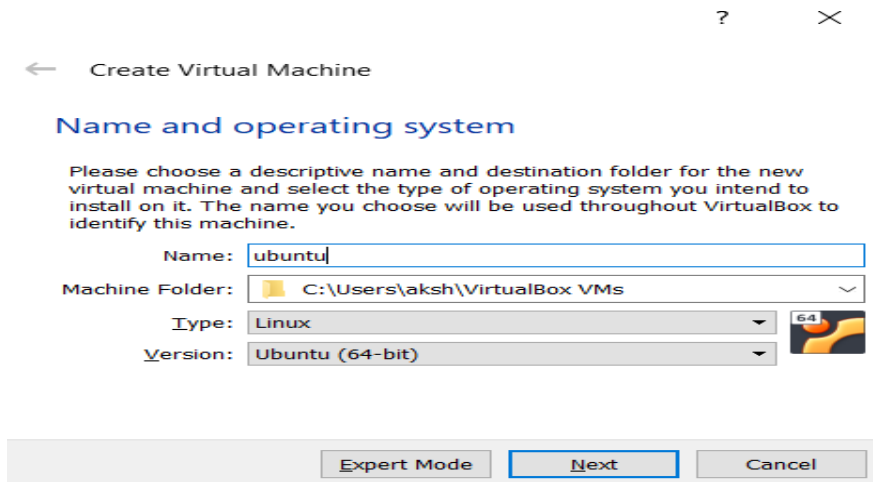
PROCEDURE:

Steps to create a virtual machine [Ubuntu] in VirtualBox:

1. Open the Oracle VMVirtualBox.

2. Click the Newiconor„Ctrl+N‟to create a new virtual machine
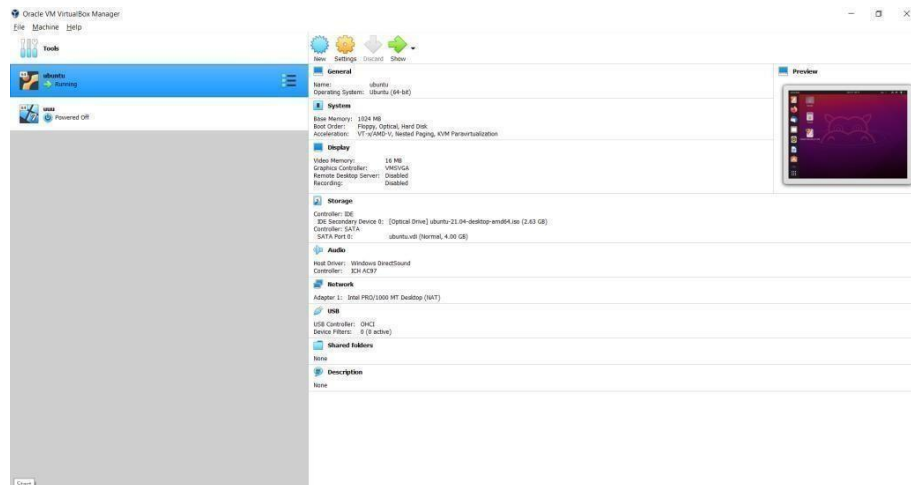
3.Enter a name for the new virtual machine. Choose the Type and Version. Note that VirtualBox automatically changes 'Type' to Linux and 'Version' to 'Ubuntu (64 bit)' if the name is given as „Ubuntu‟. Click Next.

3. Select the amount of RAM to use(eg: 1000 MB). The ideal amount of RAM will automatically be selected. Do not increase the RAM into the red section of the slider; keep the slider in the greensection.

4. Accept the default 'Create a virtual hard drive now' and click the 'Create' button.

5. Choose the hard disk filetypeasVMDK(VirtualBoxDisk).ClickNext.

6. Click Next to accept the default option„ Dynamically allocated" for storage on a physical hard drive.

7. Select the size of the virtual hard disk(eg:8.0 GB) and click create.

8. The newly created virtual machine will be displayed in the dashboard.

9. Download the ISO file [Ubuntu disk image file]. The latest version of Ubuntu iso file can be downloaded from the link https://ubuntu.com/download/desktop

10. For previous versions, gotohttp://releases.ubuntu.com. Choose the preferred version of Ubuntu and download the iso file.

Setup ubuntu:

1. To set up the Ubuntu disk image file (iso file) go to settings.

2. Click Storage. Under the „Storage Devices" section click„Empty".

3. In the Attributes section, click the disk image and then "Choose Virtual Optical Disk File".

4. Browse and select the downloaded iso file. Click ok.
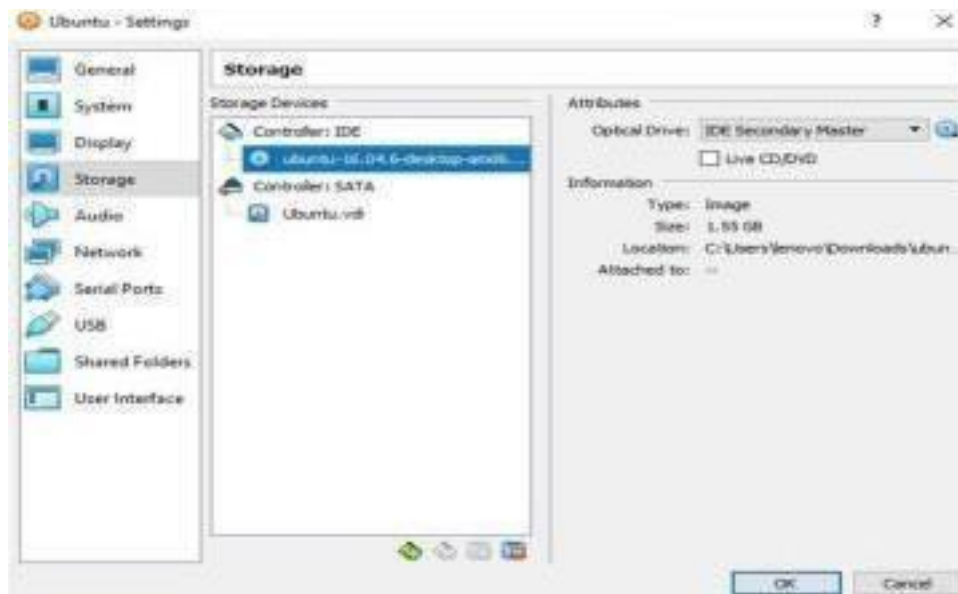
5. Select the newly created virtual machine in the dashboard and click the start button.
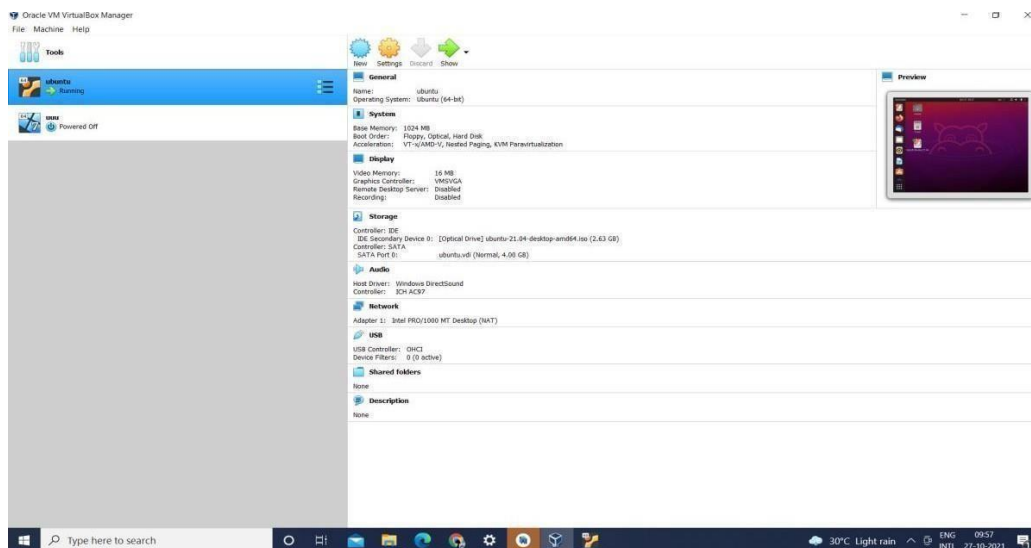
6. In the welcome screen, click the "Install Ubuntu" button
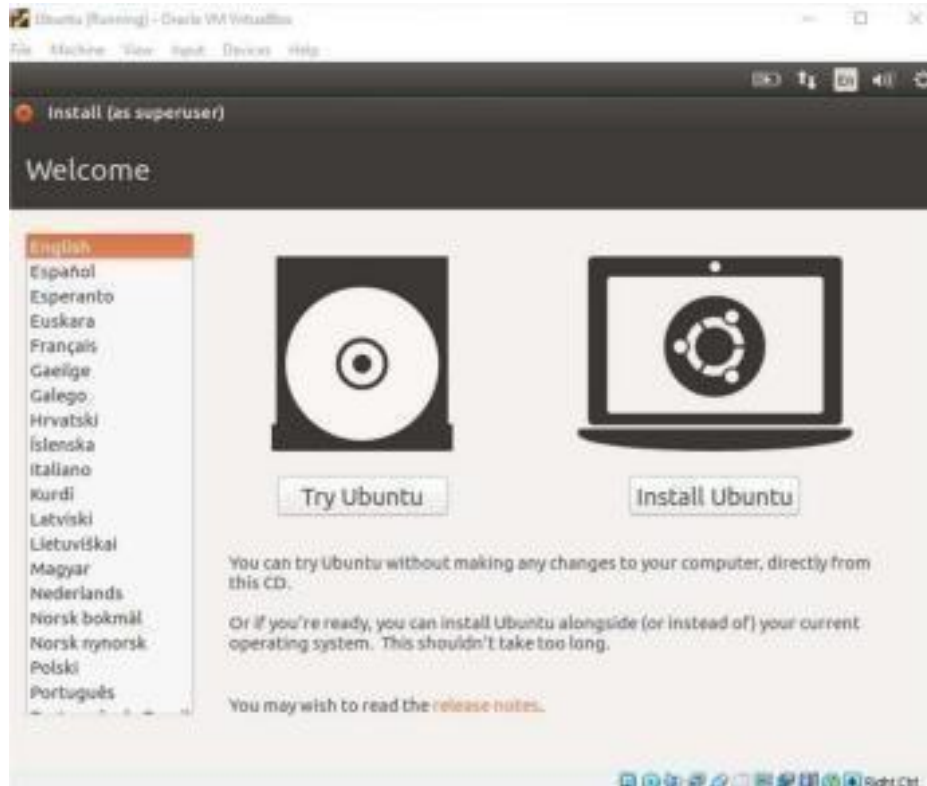
7. Click 'Continue' button.

8. Make sure 'Erase disk and install Ubuntu' option is selected and click 'Install Now' button.
9. Choose the default and click continue.

10. Setup up your profile by creating username and password.

11. After installation is complete, click 'Restart Now' button and follow the instructions.

12. The Ubuntu OS is ready to use. Login with the username and password.

13. Choose the default and click continue.

14. Setup up your profile by creating username and password.

15. After installation is complete, click 'Restart Now' button and follow the instructions.

20. The Ubuntu OS is ready to use. Login with the username and password.

TRANSFERING FILES FROM ONE VM TO ANOTHER VM:

Step 1: Creating virtual machines in Oracle VirtualBox with Ubuntu and another OS establishes separate environments for file transfer. Virtualization ensures isolation and allows experimentation without affecting the host system.
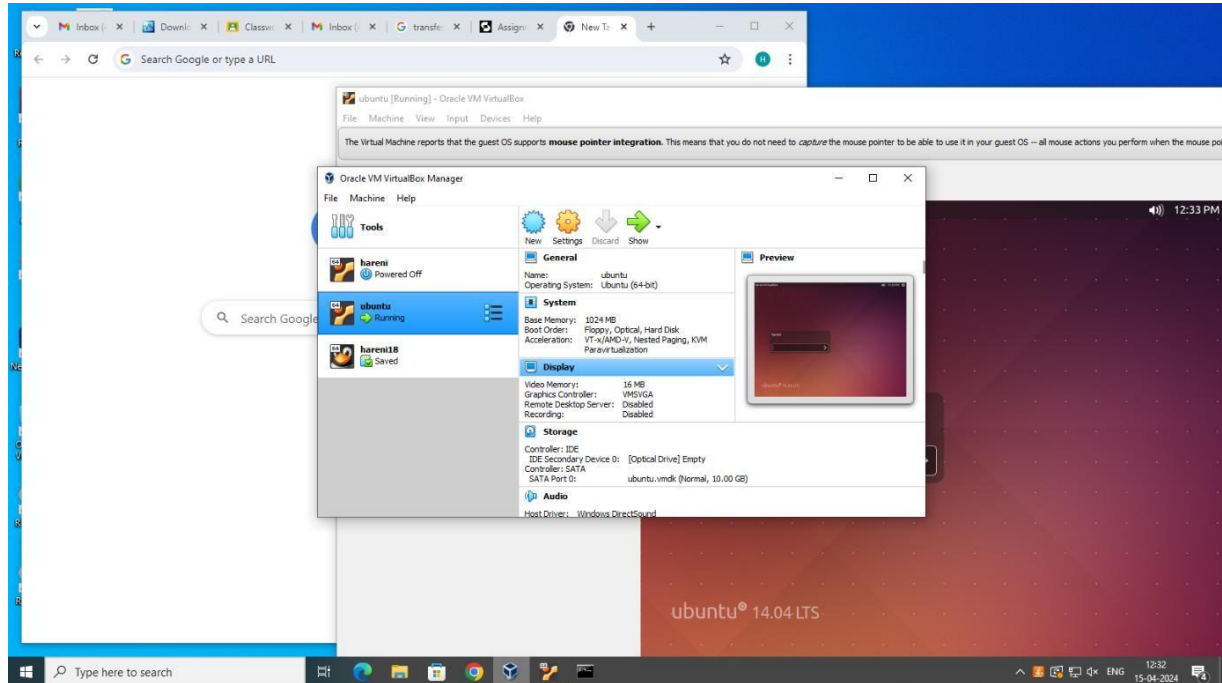
Step 2: Editing the wired connections in Ubuntu facilitates network configuration. Adjusting settings ensures proper communication between virtual machines and the host system, crucial for file sharing.

Step 3: Using the command prompt to execute "ipconfig" retrieves essential network information. This command provides IP addresses required for establishing connections between the virtual machines.

Step 4: Copying and pasting IP address, DNS address, NAT gateway, and subnet mask into IPv4 settings configures network parameters accurately. This step ensures proper addressing and routing for communication between the virtual machines.

Step 5: Changing NAT gateway to bridge adapter in network settings enables direct communication between the virtual machines and the external network. Bridge adapter mode facilitates seamless data transfer without network address translation.

Step 6: Connecting to the server from one virtual machine to another using the IP address initiates the file transfer process. This step establishes a network connection, essential for accessing files stored on the second virtual machine.

Step 7: Providing the username and password for authentication ensures secure access to files on the second virtual machine. Authentication prevents unauthorized users from accessing sensitive data during file transfer.
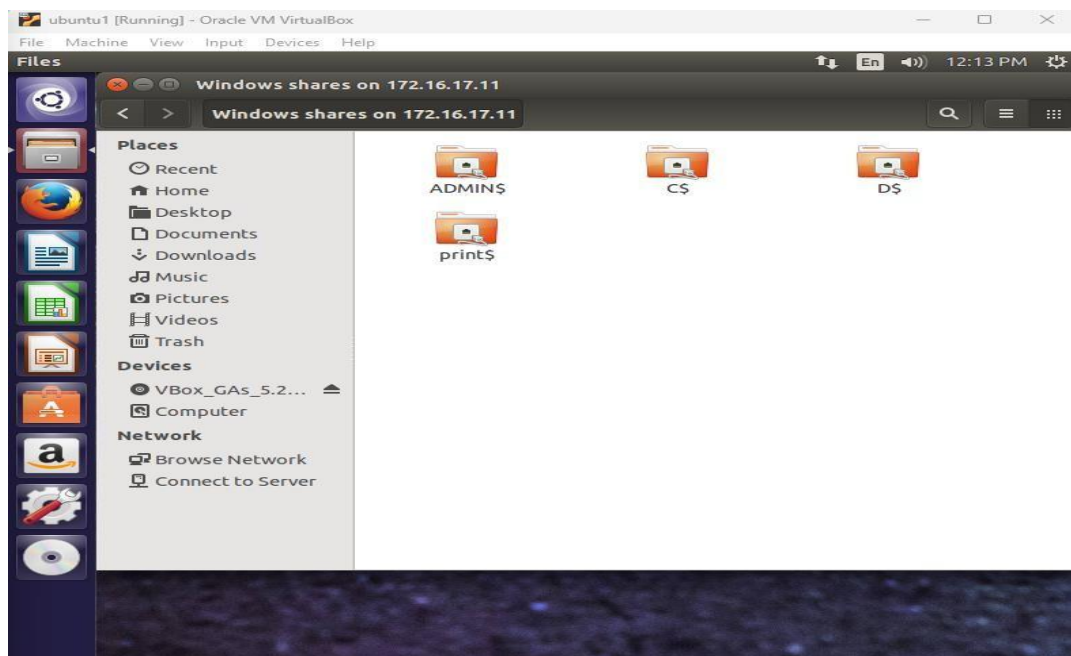
Step 8: Granting access to specific files on the second virtual machine and selecting them for sharing limits the scope of data transfer. This step ensures that only necessary files are accessible, enhancing security and privacy during the file transfer process.
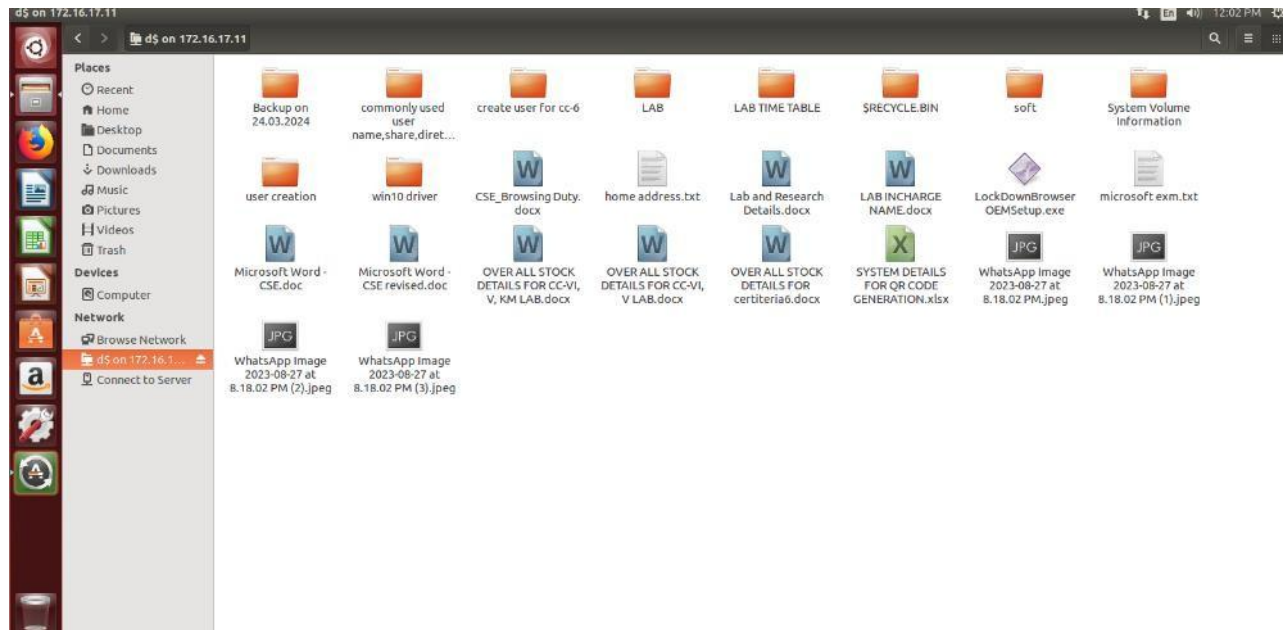
RESULT:

The Virtualbox installation is completed and the Virtual machine is created on top of the Windows host operating system.

Ex.no:7 Find a procedure to launch virtual machine using trystack.
Date:

Aim:
To Find a procedure to launch virtual machine using trystack.
Open trystack.org and Login.

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, Fill Network Address with appropriate CIDR, for example 192.168.1.0/24.
   Use private network CIDR block as the best practice and Select IP Version with appropriate
   IP version, in this case IPv4 and 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. 1. Go to Compute > Instances and then click Launch Instance.
2. In Details tab, Fill Instance Name, >Select Flavor, for example m1.medium>Fill Instance
   Count with >Select Instance Boot Source with Boot from Image > Select Image Name with
   Ubuntu 14.04 amd64 (243.7 MB) if you want install Ubuntu 14.04 in your virtual machine.
3. 3. In Access & Security tab,
   1. Click [+] button of Key Pair to import key pair.
   2. In Import Key Pair dialog, > Fill Key Pair Name with your machine > Fill Public Key with
   your SSH public key (usually is in ~/.ssh/id_rsa.pub).
   3. Click Import key pair.
   4. In Security Groups, mark/check default.
4. 4. In Networking tab, 1. In Selected Networks, select network that have been created in Step
   1, for example internal.
5. Click Launch. If you want to create multiple instances, you can repeat step 1-5. I created one
   more instance with instance name Ubuntu 2.
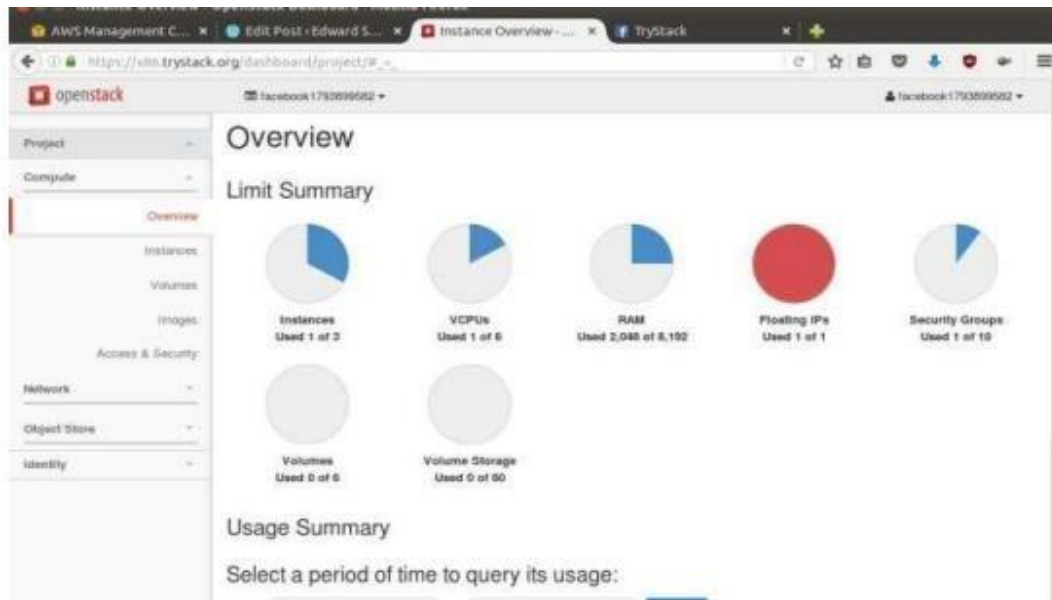
Step 3: Create Router

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:
   Select External networks with external>Then OK.
5. Click Add Interface button.
   Select Subnet > Click Add interface.
6. Go to Network > Network Topology
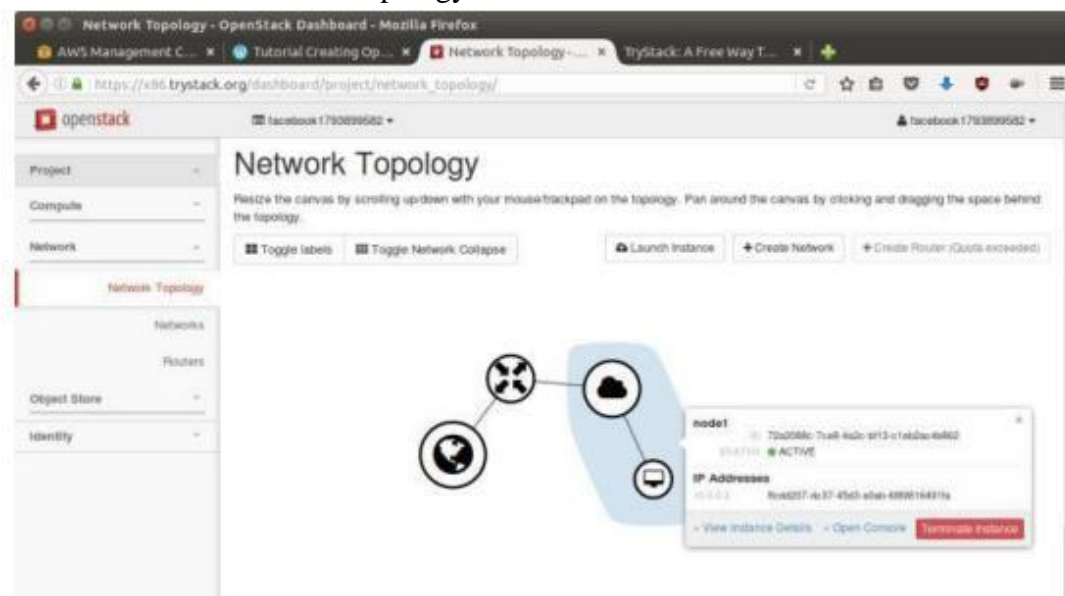
Step 4: Configure Floating IP Address

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 VirtualMachineis Launched using Trystack.

Ex. No.: 8 A   HADOOP INSTALLATION - ONE NODE HADOOP CLUSTER

Date:

AIM:
To install hadoop environment in ubuntu.

PROCEDURE:
1. Unzip the files jdk and hadoop.
2. Set the path of jdk and hadoop in /etc/profile file
3. Check the updation of file by source command.
4. Update 5 files as soon in the below command.
5. Format the path.
6. Start the dfs and check the no. of nodes running.
7. Start the yarn and check the no. of nodes running.
8. Open the browser and check whether the hadoop is installed correctly.
9. Add a file and check whether we can view the file.
10. After completing the process stop dfs and yarn properly.

COMMANDS:

1. cloud@ubuntu:~$ ls
    Desktop   Downloads exp2.odt oddeven Public   Videos
    Documents exp1.odt Music   Pictures Templates VirtualBox VMs
2. cloud@ubuntu:~$ cd Downloads
3. cloud@ubuntu:~/Downloads$ tar zxvf jdk-8u60-linux-x64.gz
4. cloud@ubuntu:~/Downloads$ cd jdk1.8.0_60/
5. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ pwd
    /home/cloud/Downloads/jdk1.8.0_60 - *COPY THE PATH*
6. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ sudo nano /etc/profile
A file opens and after the below two lines add in last lines

```
# /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
# and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).


Type the lines as follows
JAVA_HOME=/home/cloud/Downloads/jdk1.8.0_60
PATH=$PATH:$JAVA_HOME/bin


export PATH JAVA_HOME
```

7. *cloud@ubuntu:~/Downloads/jdk1.8.0_60$ source /etc/profile*

8. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ sudo apt-get install openssh-server
[sudo] password for cloud:

9. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ ssh-keygen
Press enter key, no need to type enter key.

10. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ ssh-copy-id -i localhost
cloud@localhost's password:

11. cloud@ubuntu:~/Downloads/jdk1.8.0_60$ cd ..

12. cloud@ubuntu:~/Downloads$ tar zxvf hadoop-2.7.0.tar.gz
13. cloud@ubuntu:~/Downloads$ cd hadoop-2.7.0/
14. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ pwd
      /home/cloud/Downloads/hadoop-2.7.0 *- COPY THE PATH*
15. cloud@ubuntu:~/Downloads/ hadoop-2.7.0$ sudo nano /etc/profile

A file opens and after the below two lines added in last line

```
# /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
# and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).


Type the lines as follows
JAVA_HOME=/home/cloud/Downloads/jdk1.8.0_60
HADOOP_PREFIX=/home/cloud/Downloads/hadoop-2.7.0



PATH=$PATH:$JAVA_HOME/bin
PATH=$PATH:$HADOOP_PREFIX/bin


export PATH JAVA_HOME HADOOP_PREFIX
```

16. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ source /etc/profile
17. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ cd etc
18. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc$ cd hadoop
19. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ nano hadoop-env.sh
Goto the end of the file and enter the last line

```
export JAVA_HOME=/home/cloud/Downloads/jdk1.8.0_60
export HADOOP_PREFIX=/home/cloud/Downloads/hadoop-2.7.0
```

20. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ nano core-site.xml
Inside the configuration tag insert the following lines

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

21. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ nano hdfs-site.xml
Inside the configuration tag insert the following lines

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

22. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ cp mapred-site.xml.template mapred-site.xml
23. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ nano mapred-site.xml
Inside the configuration tag insert the following lines

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

24. cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ nano yarn-site.xml

```
<configuration>

<!-- Site specific YARN configuration properties -->
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
</configuration>
```

25. **cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc/hadoop$ cd ..**

26. **cloud@ubuntu:~/Downloads/hadoop-2.7.0/etc$ cd ..**
27. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hadoop namenode -format**
28. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/start-dfs.sh**
29. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ jps**
    5632 DataNode
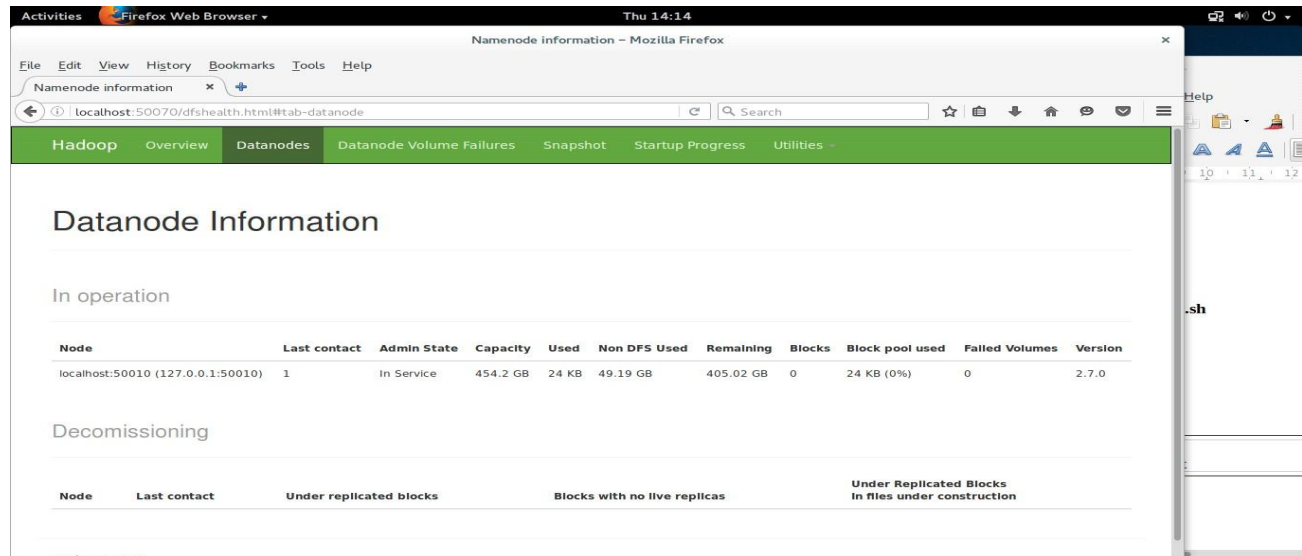   5428 NameNode
    5979 Jps
    5851 SecondaryNameNode

30. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/start-yarn.sh**
31. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ jps**
    5632 DataNode
    6209 NodeManager
    6050 ResourceManager
    5428 NameNode
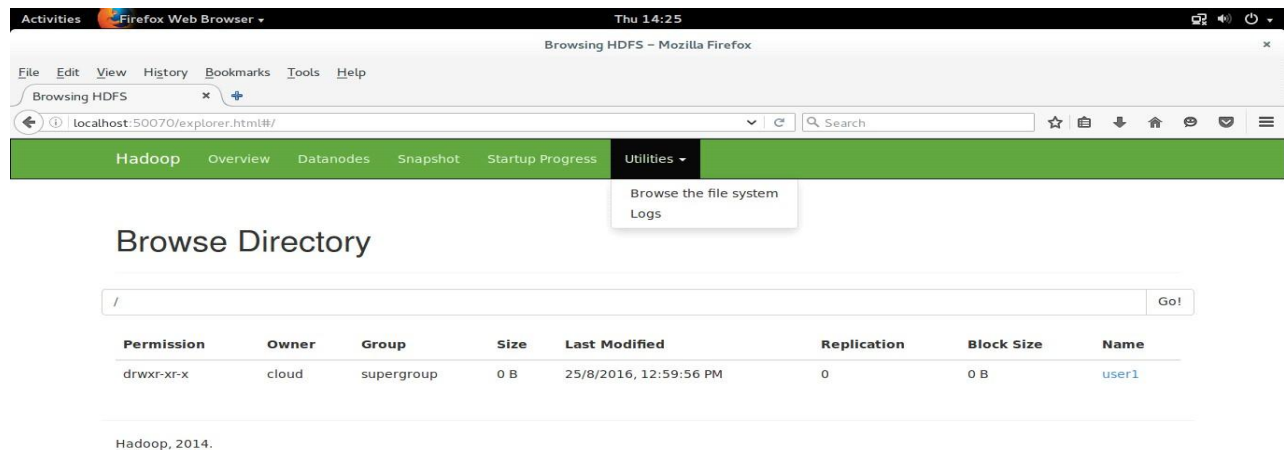   6522 Jps
    5851 SecondaryNameNode

32. Open a browser and type as: http:\\localhost:50070
Choose DataNode. U can see a single node created in it

U can see the folder created inside utilities tab.
33. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -mkdir /user1

34. Open a new terminal

```
cloud@ubuntu:~/Downloads/hadoop-2.7.0$          cd          ..
cloud@ubuntu:~/Downloads$ tar zxvf mrsampledata.tar.gz
                        file1.txt
                        file2.txt
                        file3.txt
                        file4.txt
                        file5.txt
```
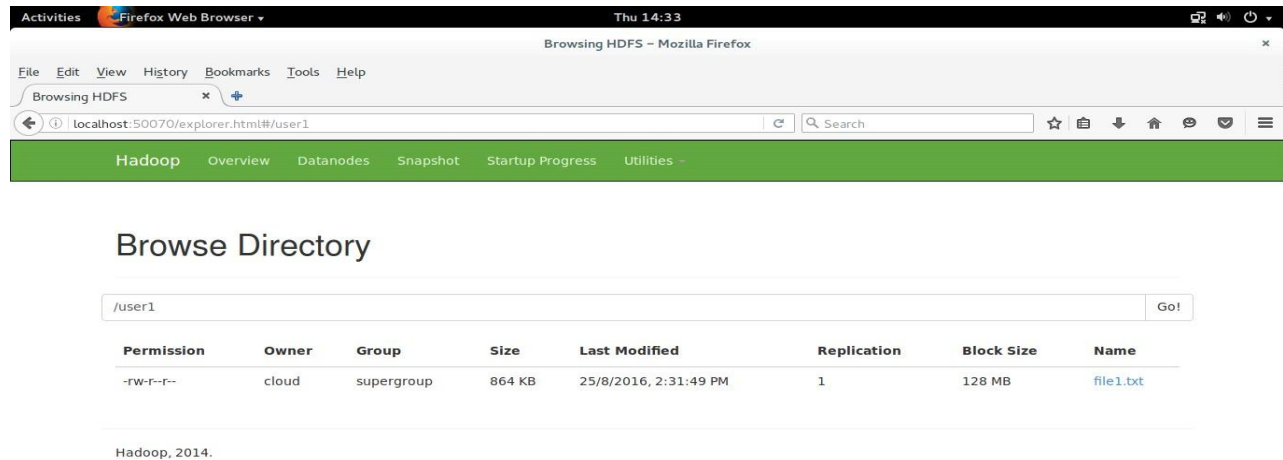
35. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -put ../file1.txt /user1

Inside the browser. Click the folder name user1 and you can see the file1.txt file inside the user1 folder

36. After completing all the process, stop the dfs and yarn as follows:
    cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/stop- yarn.sh
    cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/stop- dfs.sh

RESULT:
        Thus hadoop is installed in the physical machine and executed successfully.

Ex. No.: 8 B   **WORD COUNT PROGRAM - USING MAP AND REDUCE TASK**

Date:

AIM:

 To execute wordcount program using map and reduce task

PROCEDURE:

1. Format the path.
2. Start the dfs and check the no. of nodes running.
3. Start the yarn and check the no. of nodes running.
4. Open the browser and check whether the hadoop is installed correctly.
5. Add a file and check whether we can view the file.
6. Implement the grep command for the file added and see the result.
7. Implement the wordcount command for the file added and see the result.
8. After completing the process stop dfs and yarn properly.

COMMANDS:

1. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hadoop namenode -format**
2. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/start-dfs.sh**
3. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ jps**

                        5632 DataNode
                        5428 NameNode
                        5979 Jps
                        5851 SecondaryNameNode

4. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/start-yarn.sh**
5. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ jps**

                        5632 DataNode
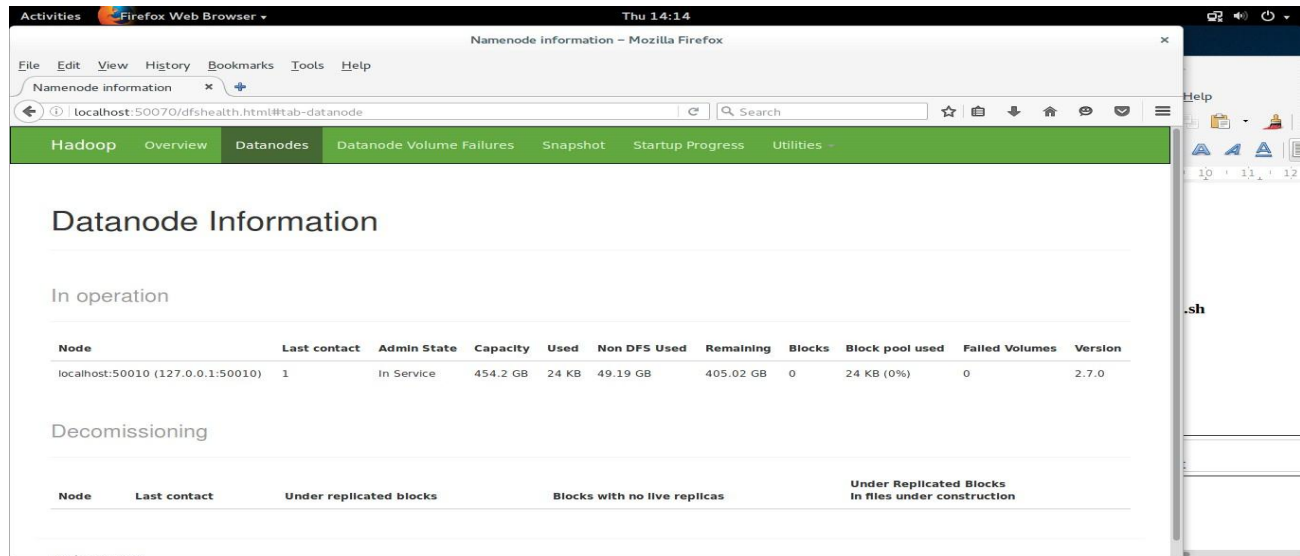   6209 NodeManager
                        6050 ResourceManager
                        5428 NameNode
                        6522 Jps
                        5851 SecondaryNameNode

6. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/stop-dfs.sh**
7. **cloud@ubuntu:~/Downloads/hadoop-2.7.0$ sbin/stop-yarn.sh**
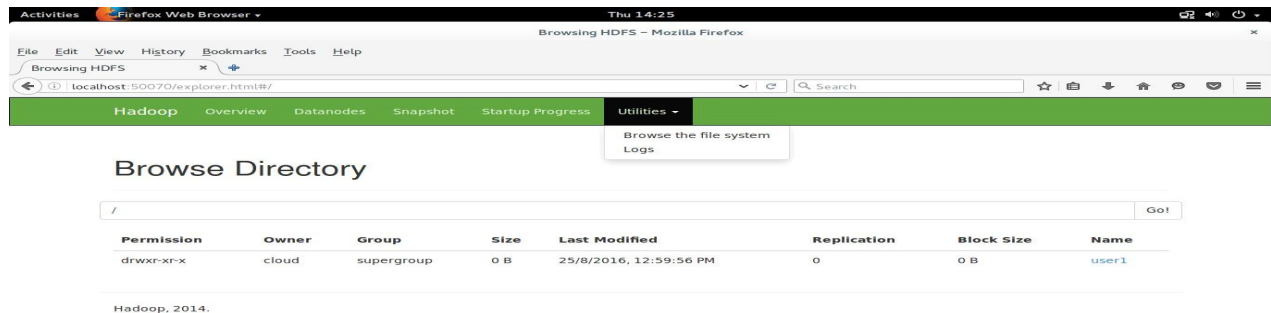8. Open a browser and type as: http:\\localhost:50070
Choose DataNode. U can see a single node created in it

U can see the folder created inside utilities tab.
9. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -mkdir /user1

10. Open a new terminal
11. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ cd ..
12. cloud@ubuntu:~/Downloads$ tar zxvf mrsampledata.tar.gz

                    file1.txt
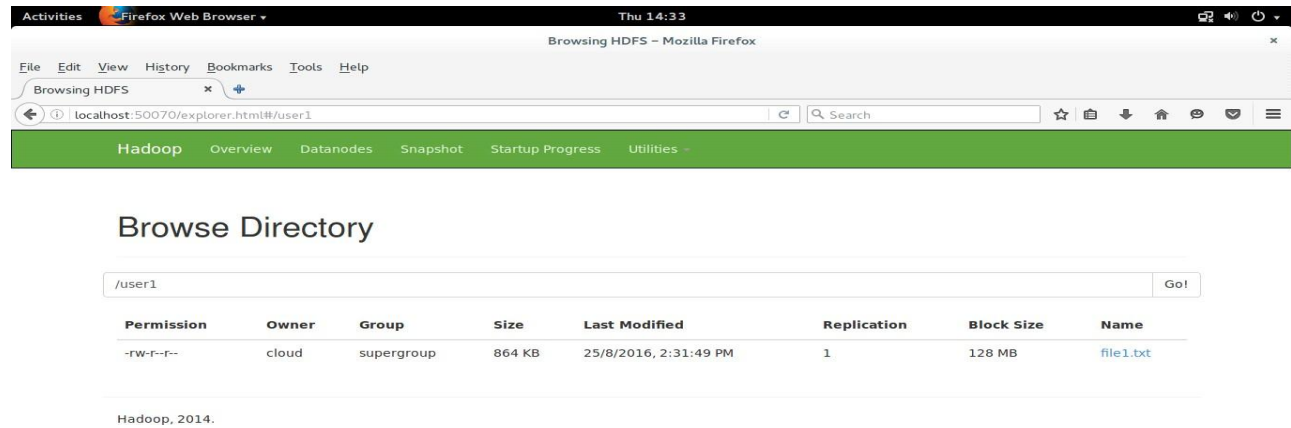   file2.txt
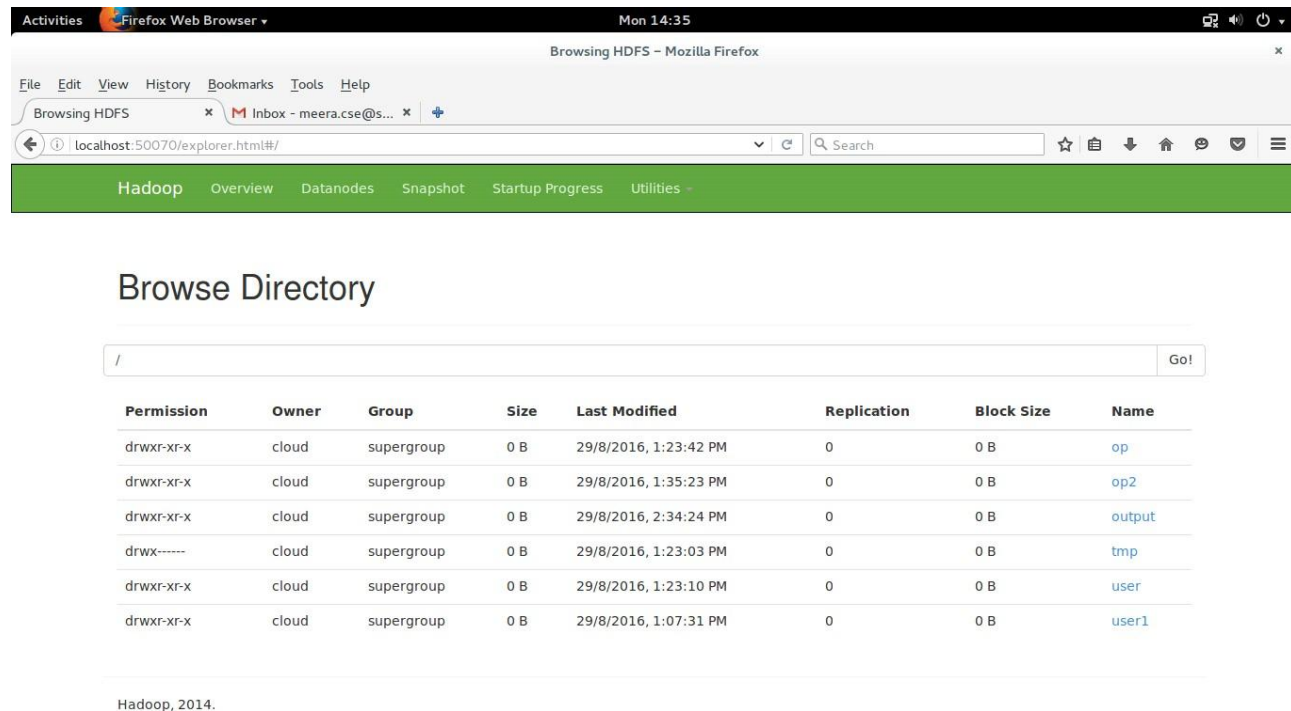   file3.txt
                    file4.txt
                    file5.txt

13. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -put ../file1.txt /user1

14. Inside the browser. Click the folder name user1 and you can see the file1.txt file inside the user1 folder

15. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0.jar grep /user1/* /output '(CSE)'

16. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0.jar wordcount /user1/file1.txt /output1

17. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -cat /output/*
9894   CSE

18. cloud@ubuntu:~/Downloads/hadoop-2.7.0$ bin/hdfs dfs -cat /output1/*
B.ARCH   9864
B.TECH(BIO) 9964
B.TECH(IT)   10000
BE(AME)   9853
BE(CIVIL)   10043
BE(CSE)   9894
BE(ECE)   10048
BE(EEE)   9937
BE(ICE)   9872
BE(MECH)   9873
You can also see the ouput by
1) output
Inside Browser Directory
output --> part-r-00000 --> Click Download

2) output1
Inside Browser Directory
output --> part-r-00000 --> Click Download

RESULT:
        Thus the wordcount program using map and reduce task is executed successfully.