

STELLA MARY'S COLLEGE OF ENGINEERING

(Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai & Accredited by NAAC)
Aruthengavilai, Kallukatti Junction, Azhikal Post Kanyakumari District – 629 202, Tamil Nadu



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CCS335 CLOUD COMPUTING LAB MANUAL

REGULATION – 2021

SEMESTER-V

2023-2024(Odd)

LAB MANUAL

Prepared by,
Dr. M. SUPRIYA,
AP/ Dept. of CSE

Institute Vision:

To emerge as a premiere institution, acknowledged as a center for excellence imparting technical education, creating technocrats who can address the needs of the society through exploration and experimentation and uplift mankind.

Institute Mission:

To provide an education that transforms students, through rigorous course-work and by providing an understanding of the needs of the society and the industry.

Department Vision:

To produce Computer Science professionals who can accomplish path-breaking solutions for a better society, through quality technical education, on gaining the required inter-personal, entrepreneurial and computing skills. .

Department Mission:

- To impart a holistic and experiential learning experience by making use of innovative teaching methodologies.
- To provide optimal technology solutions through collaborative and life-long learning for industry and societal needs with universal ethical values.
- To nurture leadership skills and facilitate various co-curricular and extra-curricular activities to implant the spirit of entrepreneurship.
- To provide industry-institute-interaction opportunities in order to motivate interdisciplinary research capabilities with an inquiring mind.

Programme Education Objectives (PEOs)

PEO1:

Graduates will be competent in creating innovative technologies through inter-disciplinary research and comprehensive skills sets that are suitable for the global computing industry.

PEO2:

Graduates will be capable of managing leading positions with a broad understanding of the application of ethics in evolving computer-based solutions for the societal needs.

PEO3:

Graduates will imbibe entrepreneurial qualities and develop their career by upgrading their, communication, analytical and professional skills constantly.

Programme Specific Outcomes(PSOs)

At the completion of the programme, the students will be able to:

PSO1:

Use data management techniques and algorithmic thinking for Software Design and Development practices.

PSO2:

Develop reliable IT solutions based on the expertise in Distributed Applications Development, Web Designing and Networking for various societal needs and entrepreneurial practices ethically.

PSO3:

Manage multidisciplinary environments effectively through their interpersonal and analytical skills and be responsible members and leaders of the society.

COURSE OBJECTIVES:

- To understand the principles of cloud architecture, models and infrastructure.
- To understand the concepts of virtualization and virtual machines.
- To gain knowledge about virtualization Infrastructure.
- To explore and experiment with various Cloud deployment environments.
- To learn about the security issues in the cloud environment

COURSE OUTCOMES:

CO1: Understand the design challenges in the cloud.

CO2: Apply the concept of virtualization and its types.

CO3: Experiment with virtualization of hardware resources and Docker.

CO4: Develop and deploy services on the cloud and set up a cloud environment.

CO5: Explain security challenges in the cloud environment.

PRACTICAL EXERCISES: 30 PERIODS

1. Install Virtualbox/VMware/ Equivalent open source cloud Workstation with different flavours of Linux or Windows OS on top of windows 8 and above.
2. Install a C compiler in the virtual machine created using a virtual box and execute Simple Programs
3. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
4. Use the 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. Install Hadoop single node cluster and run simple applications like wordcount.
8. Creating and Executing Your First Container Using Docker.
9. Run a Container from Docker Hub

NO.	DATE	NAME OF EXPERIMENT
1	08/08/23	Install Virtualbox/VMware/ Equivalent open source cloud Workstation with different flavours of Linux or Windows OS on top of windows 8 and above.
2	18/08/23	Install a C compiler in the virtual machine created using a virtual box and execute Simple Programs
3	30/08/23	Install Google App Engine. Create a hello world app and other simple web applications using python/java.
4	12/09/23	Use the GAE launcher to launch the web applications.
5	22/09/23	Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6	5/10/23	Find a procedure to transfer the files from one virtual machine to another virtual machine.
7	13/10/23	Install Hadoop single node cluster and run simple applications like wordcount.
8	25/10/23	Creating and Executing Your First Container Using Docker.
9	3/11/23	Run a Container from Docker Hub

Ex No:1	INSTALL VIRTUALBOX/VMWARE WORKSTATION
Date:	

AIM:

To find the procedure to install VirtualBox/VMWARE workstation with different flavours of Linux or Windows OS on the top of the Windows 7 or 8

PROCEDURE:

STEP1: Download the link. Download the software for Windows click and download begins.

STEP2: Download the installer file.

STEP3: Locate the downloaded installer file.

STEP4: User Access Control Warning. In UAC dialog box, Click yes to continue. Install splash screen will appear

STEP5: VMware workstation setup wizard dialog box click next to continue.

STEP6: End user license Agreement. Check “I accept the terms in the license Agreement” box and press next to continue.

STEP7: Custom setup options. Select the folder in which you would like to install the application.

STEP8: User experience setting.

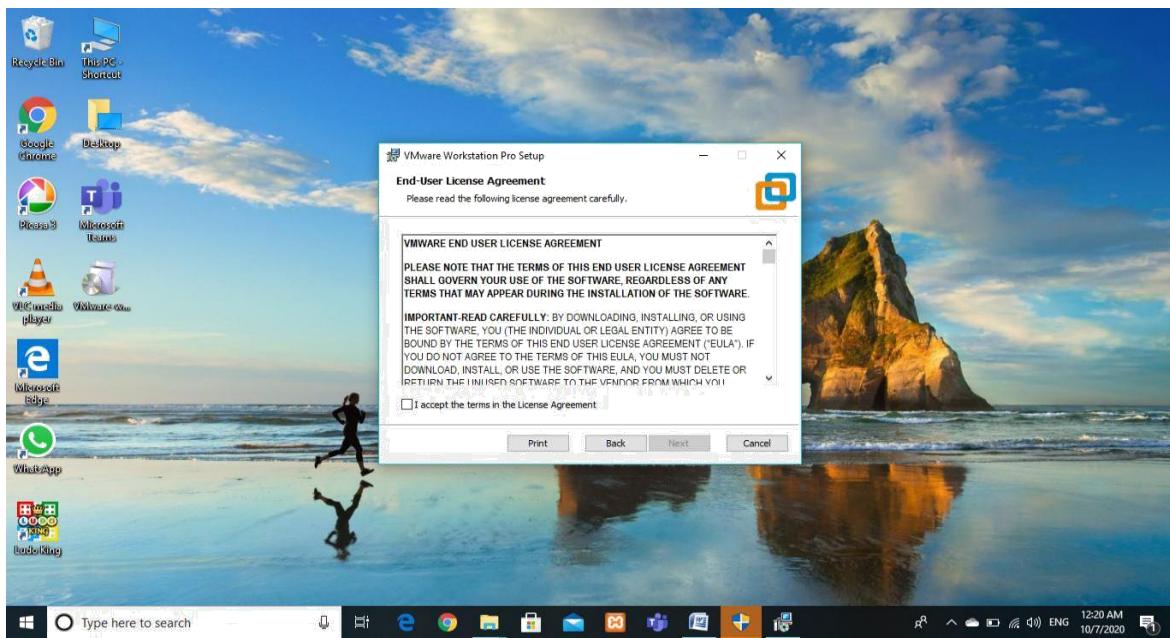
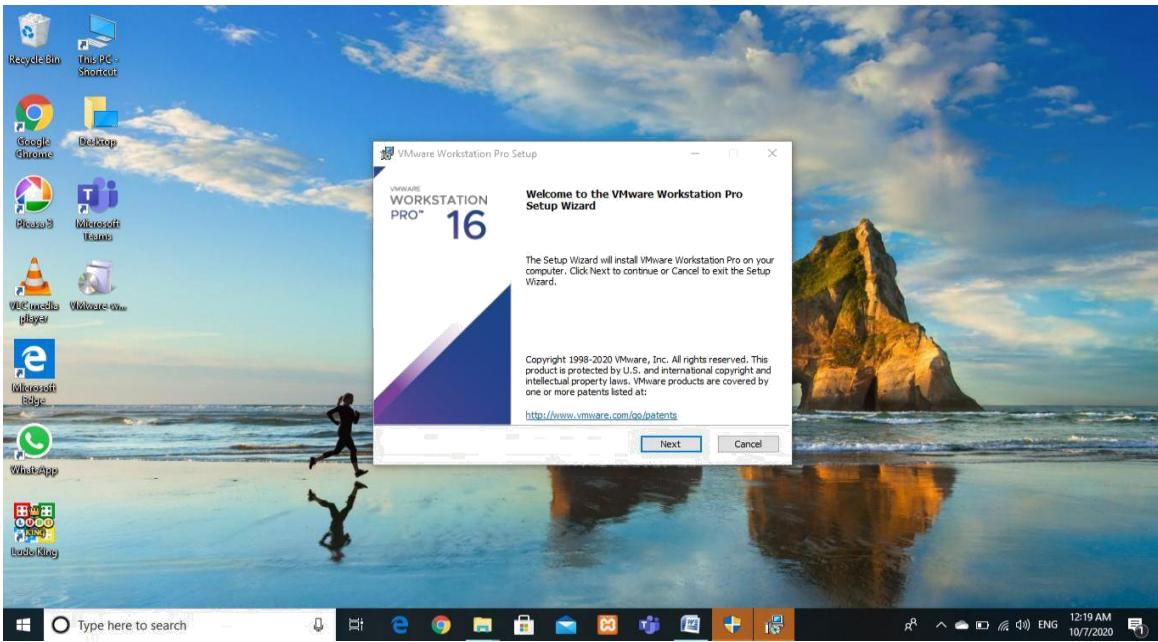
STEP9: Application shortcuts preference. Please select both the options, desktop and start menu and click next.

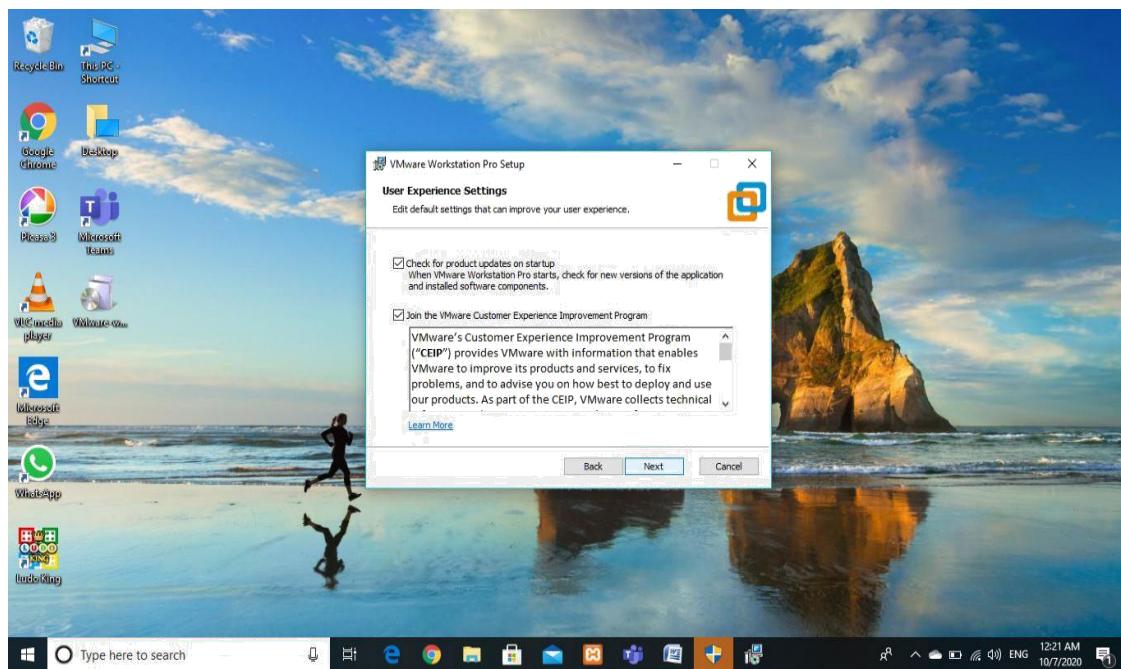
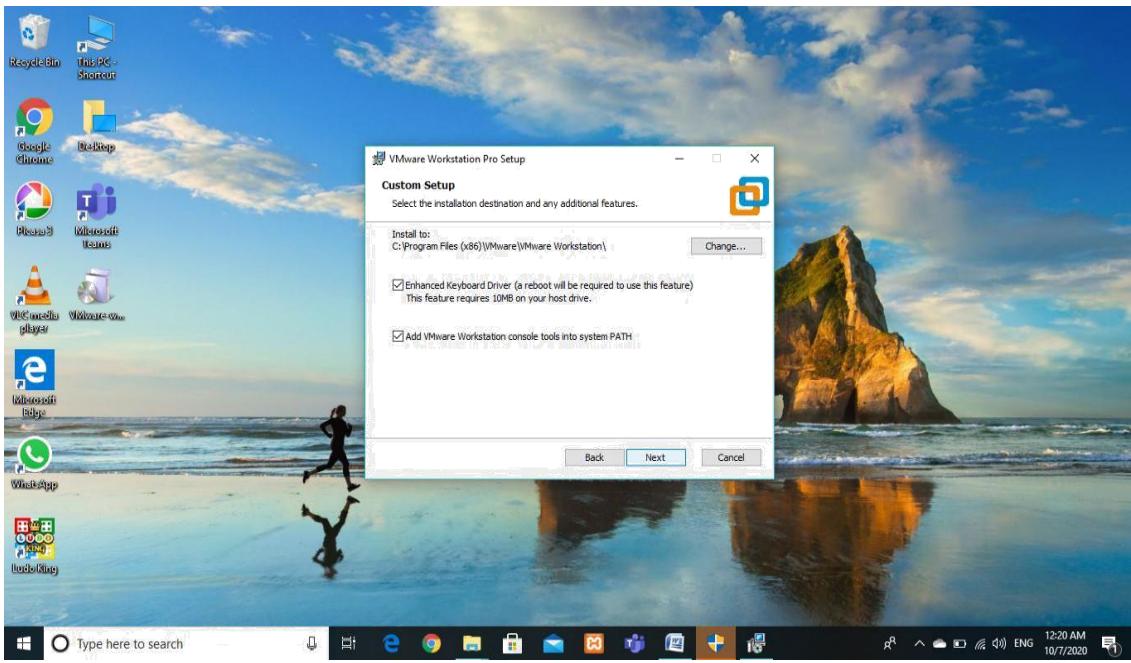
STEP10: Installation begins.

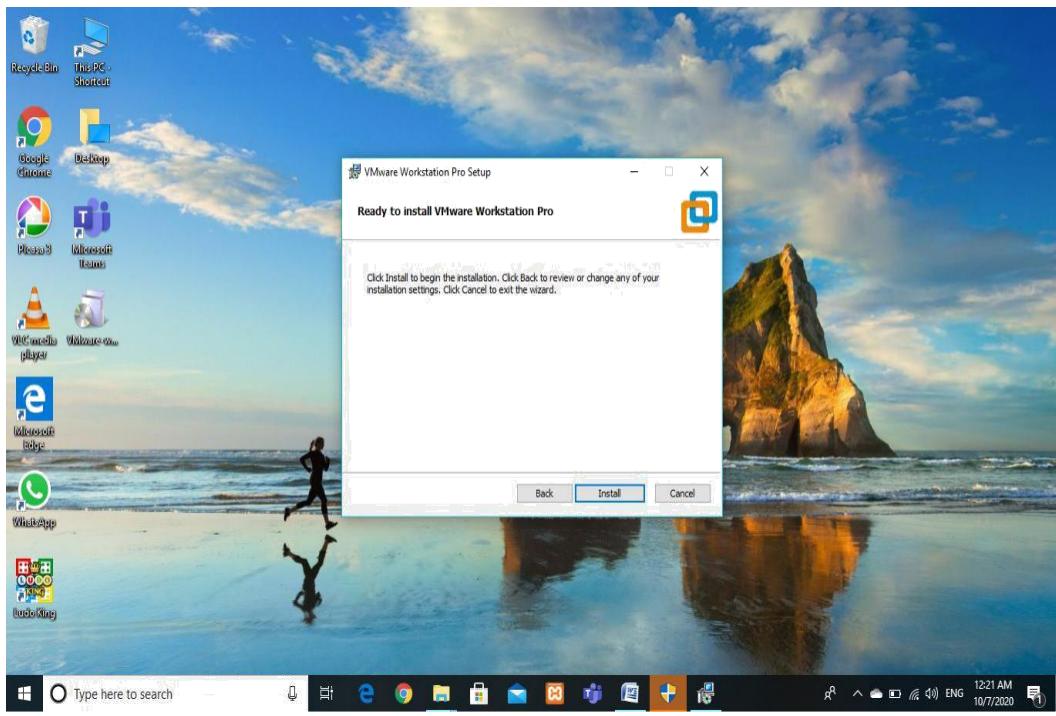
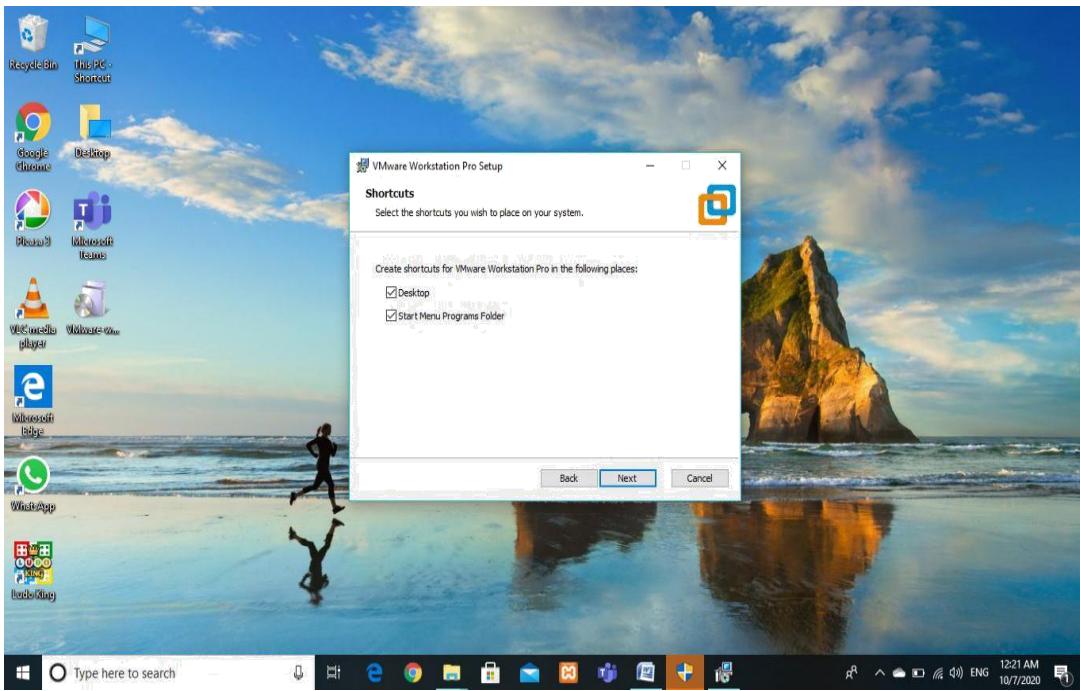
STEP11: Launch the VMware workstation and License.

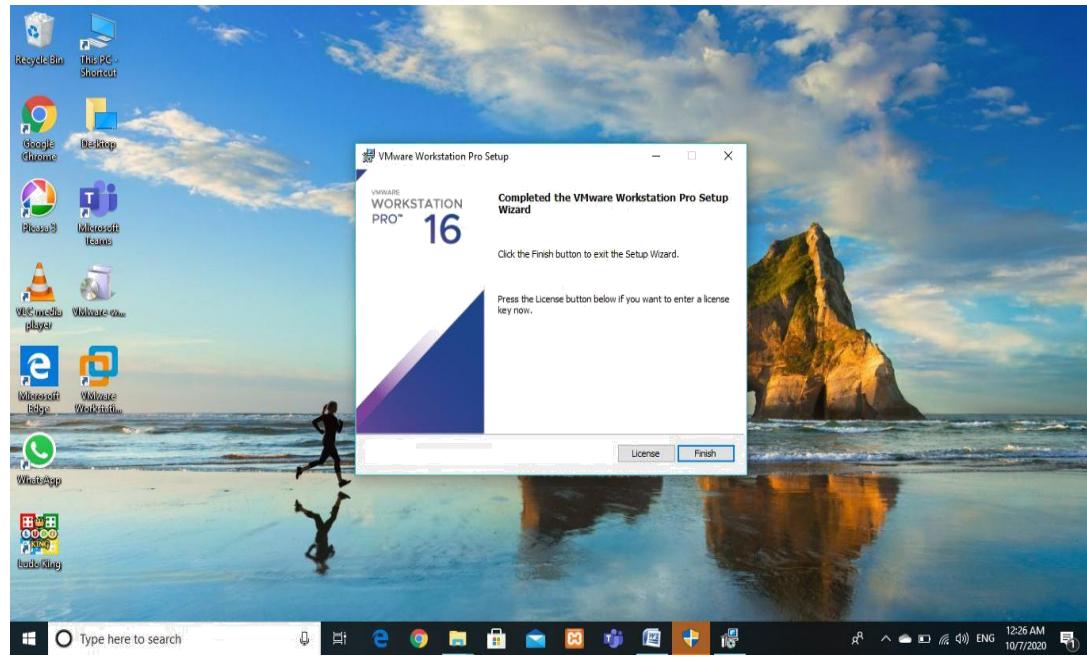
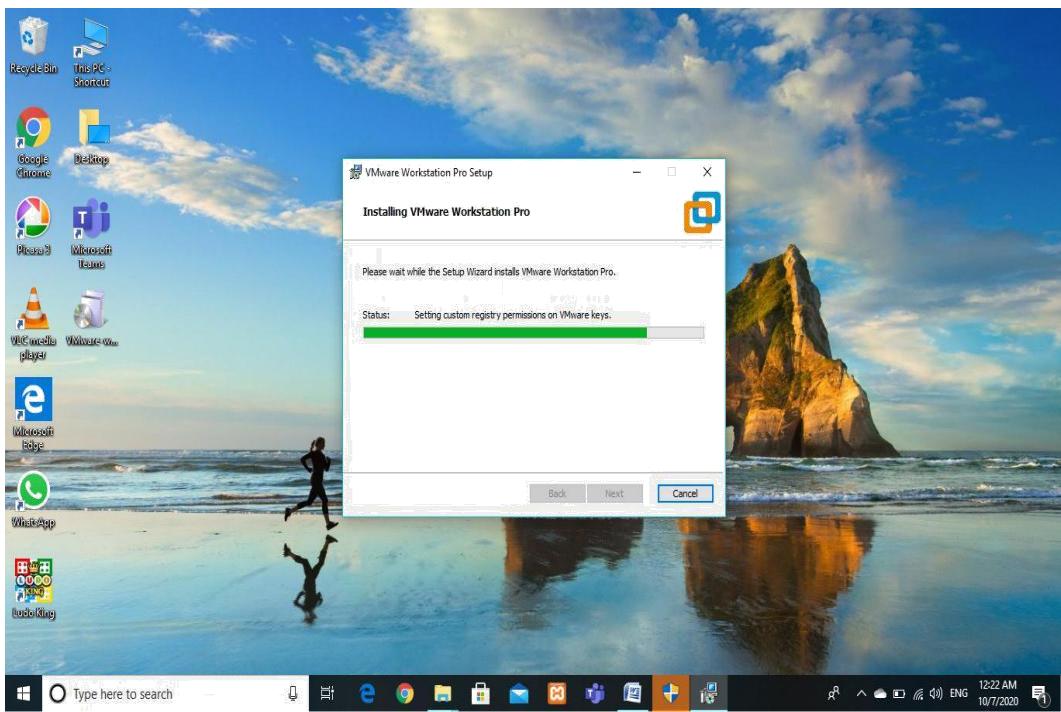
STEP12: The virtualbox/VMware software is ready

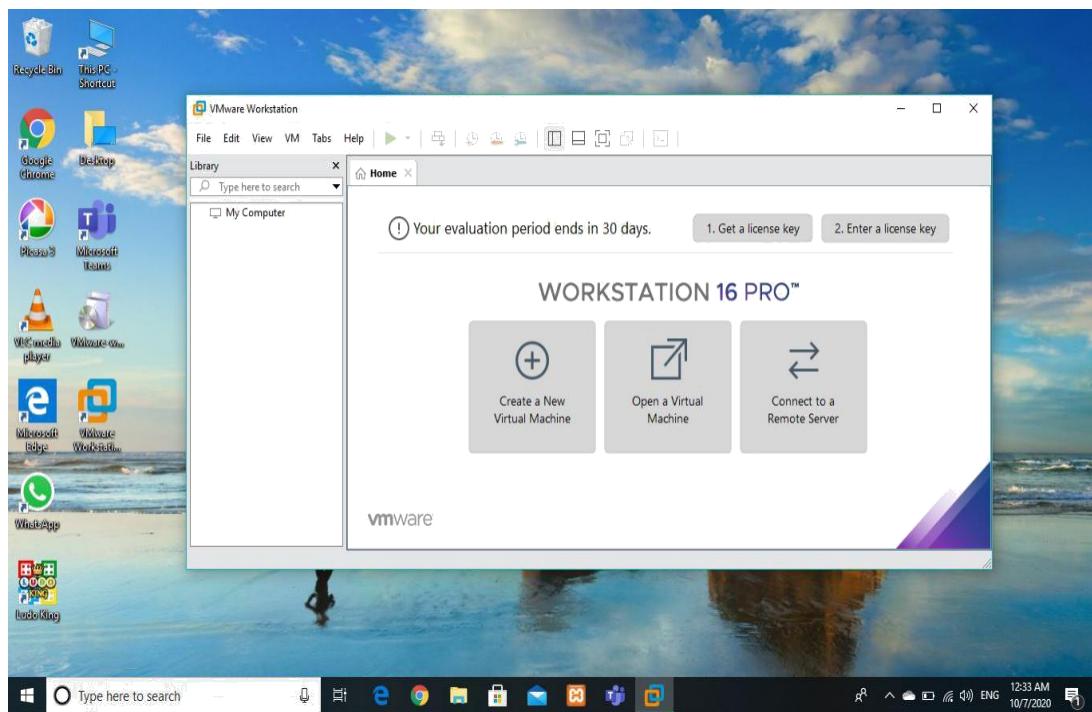
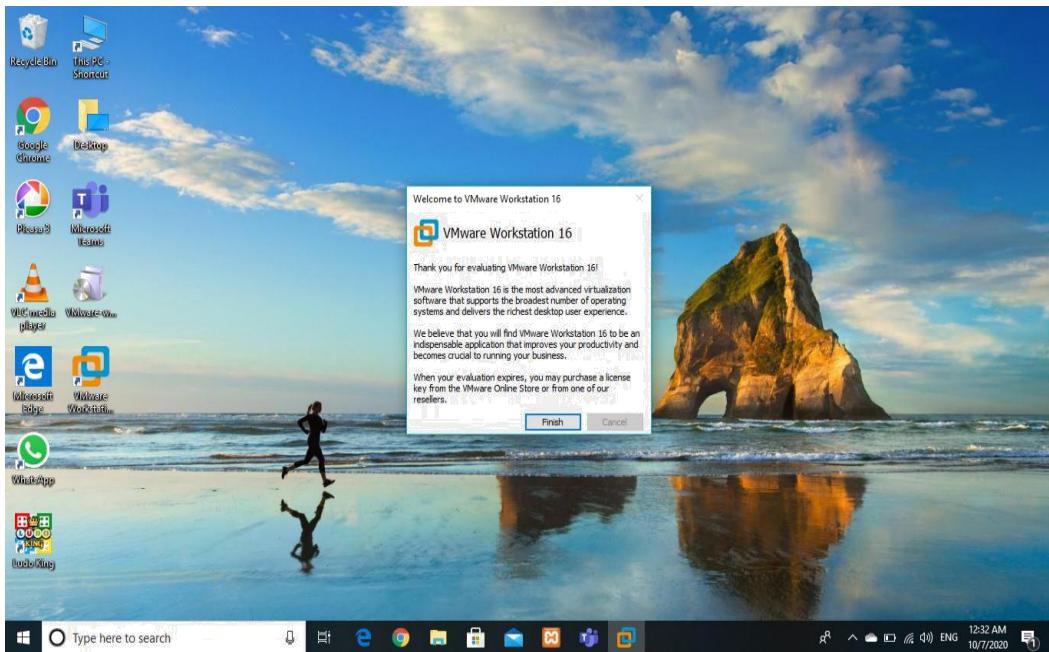












CONCLUSION:

Thus the installation of VMware workstation with different flavour of Windows OS on the top of Windows was successfully completed.

Ex No:2	SETTING UP C PROGRAMMING ENVIRONMENT
Date:	

AIM:

To install a C program compiler in the virtual machine created using virtual box and execute simple program.

PROCEDURE:

(i) Install Virtual Box:

- 1) Download Virtual box platform package for your OS.
- 2) Open the installation package and finish installing virtual box.
- 3) When finished installation, close the Window

(ii) Download Linux:

- 1) Download the latest version of Ubuntu(32-bit)

(iii) Install Linux Using Virtual Box:

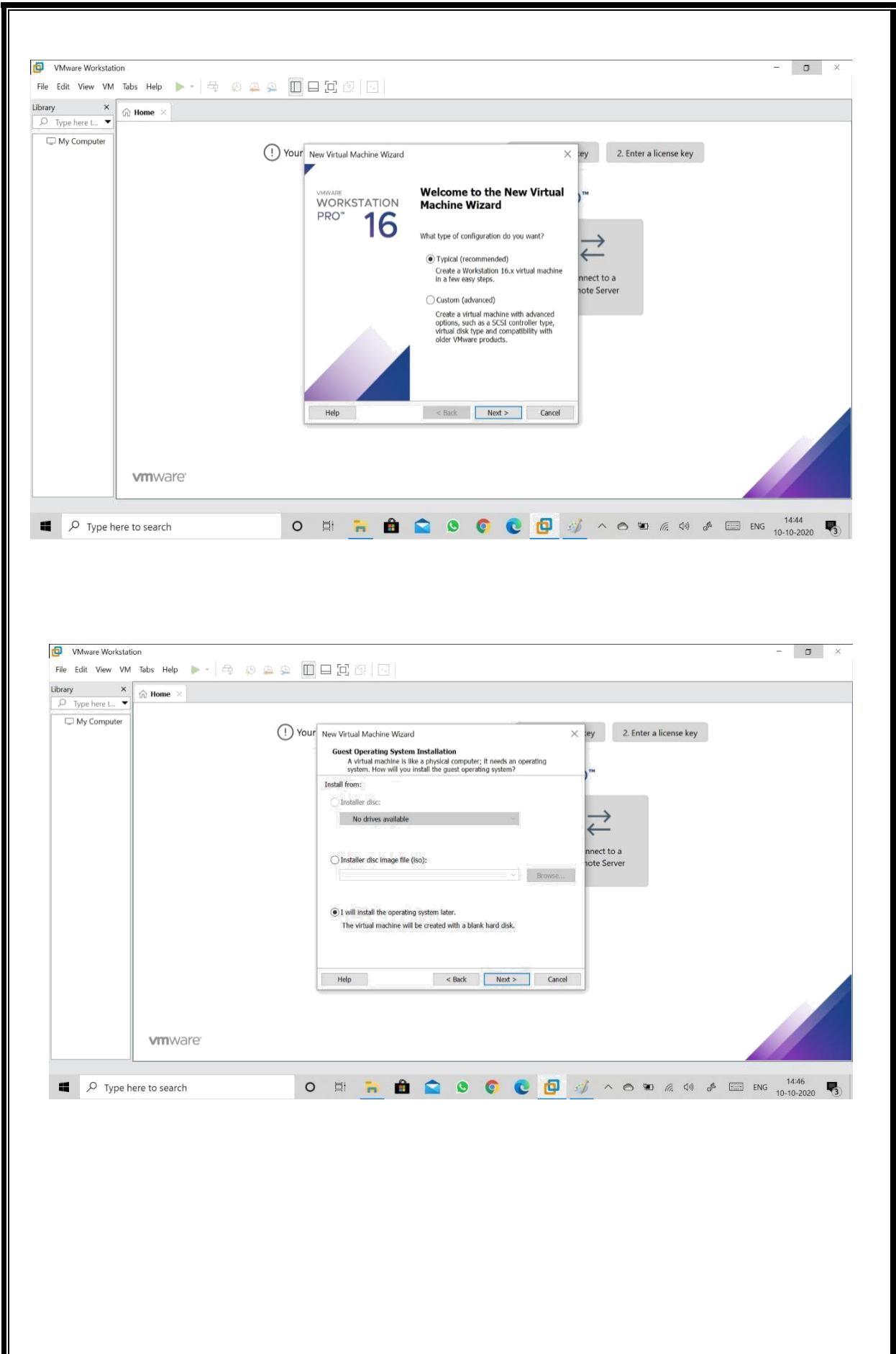
- 1) Run virtual box by double clicking the icon.
- 2) Click “New” button on the top left corner.
- 3) Click “Continue” on the pop up Window.
- 4) Type VM name select “Linux” for the OS and choose “Ubuntu” for version.
- 5) Choose the amount of memory do allocate(S12 to 1024)
- 6) Choose create a new Virtual hard Disk.
- 7) Choose VDI(Virtual Box Disk Image)
- 8) Choose “dynamically allocated” Click continue. This way the size of you virtual hard disk , will grow and use.
- 9) Click the folder and choose the Ubuntu iso file you downloaded
- 10) Select the size of the Virtual Disk (5GB) and click continue.
- 11) Click create.

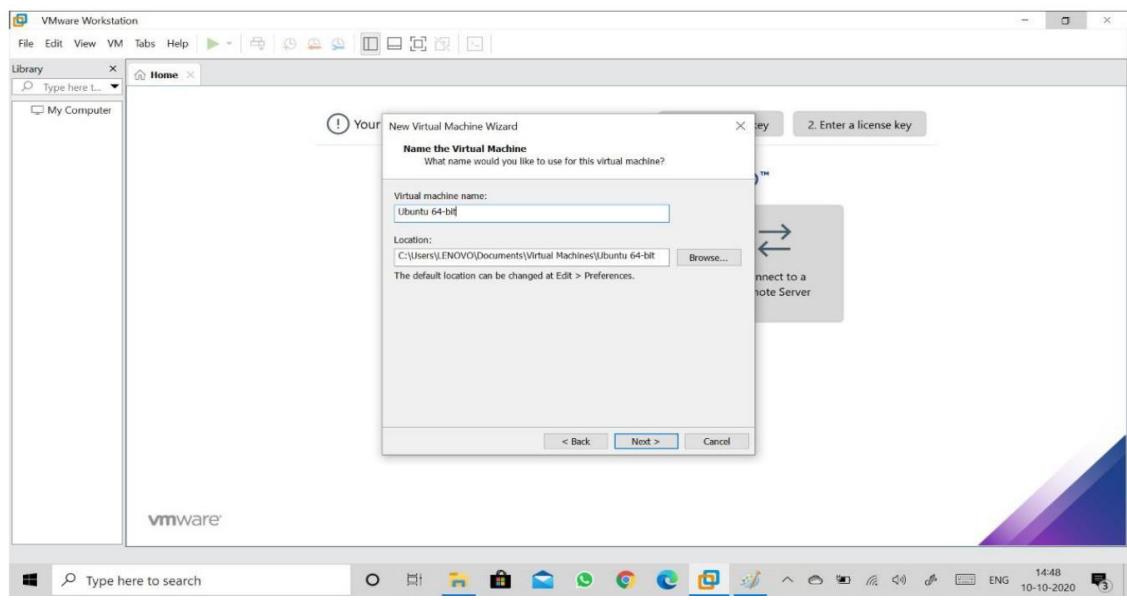
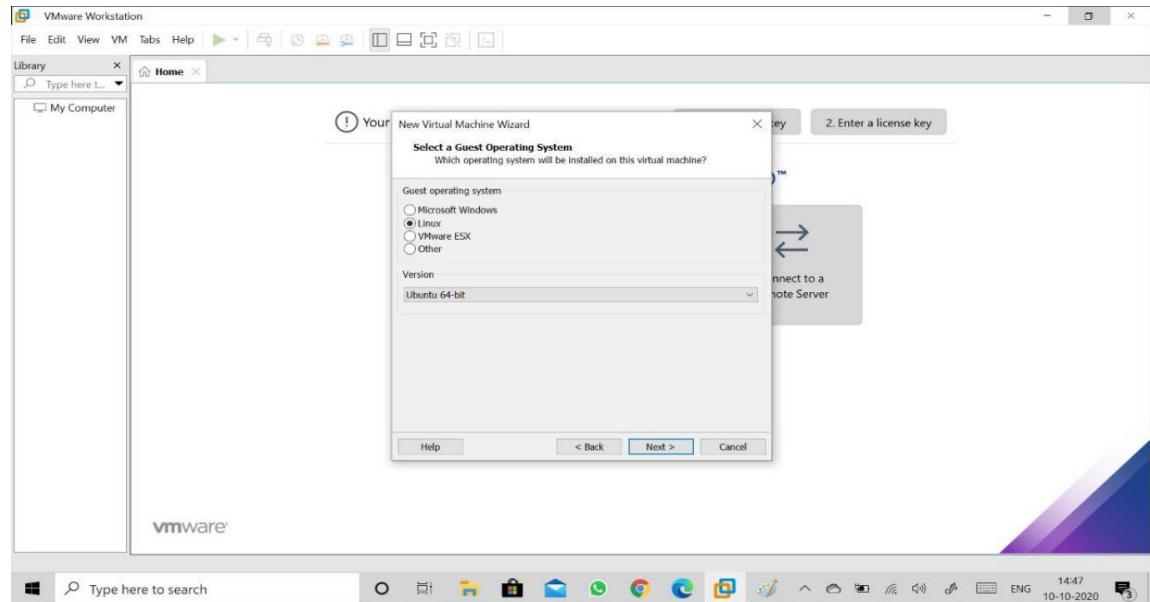
(iv) Running Linux:

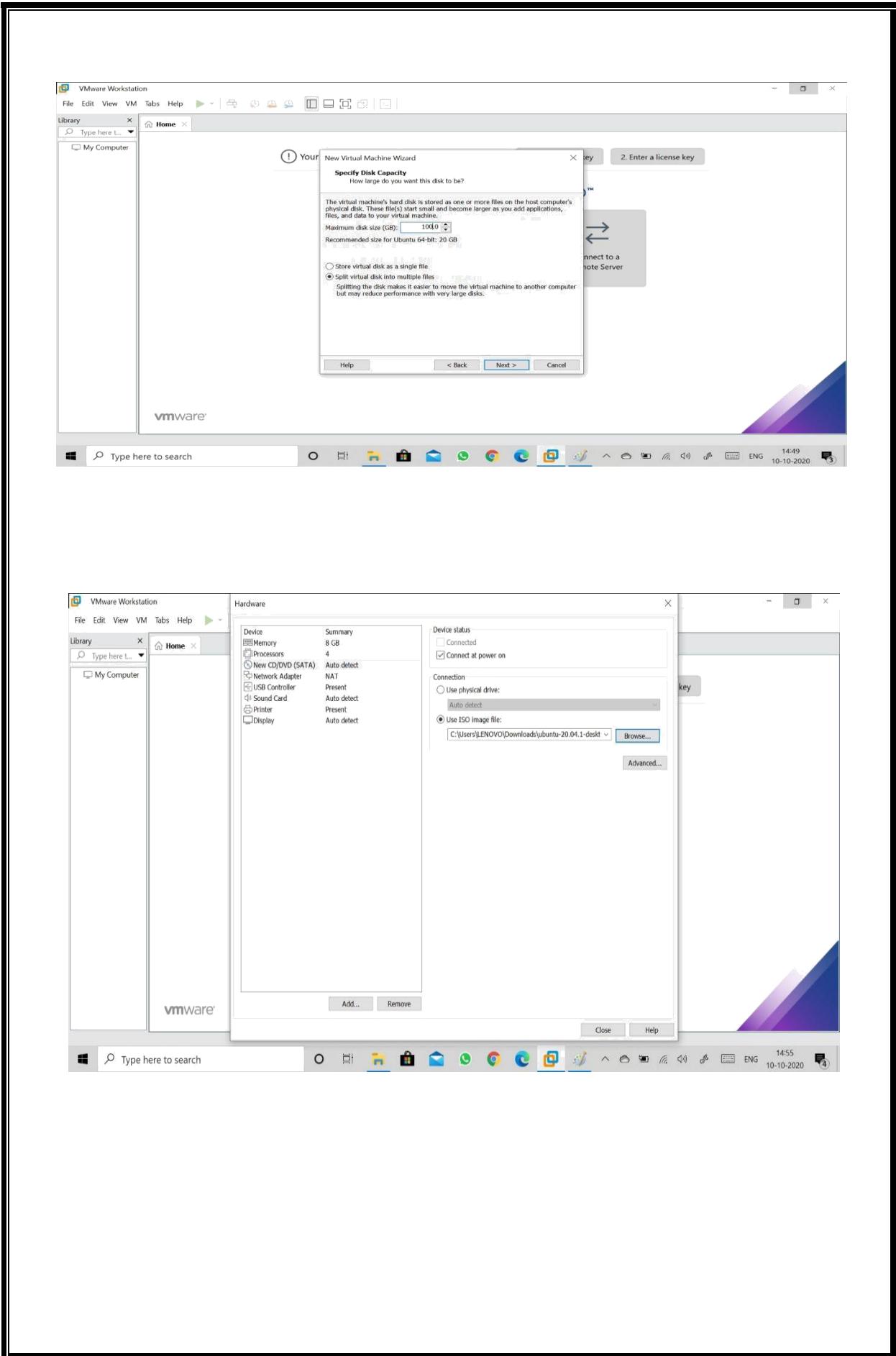
- 1) Choose Ubuntu from left column and click start.
- 2) Click continue on pop up Window.
- 3) Choose the Downloaded Ubuntu iso file and click start and install Ubuntu.
- 4) Check “Download” updated and click forward.
- 5) Choose “Erase disk and install Ubuntu”.
- 6) Click install when finished click restart.

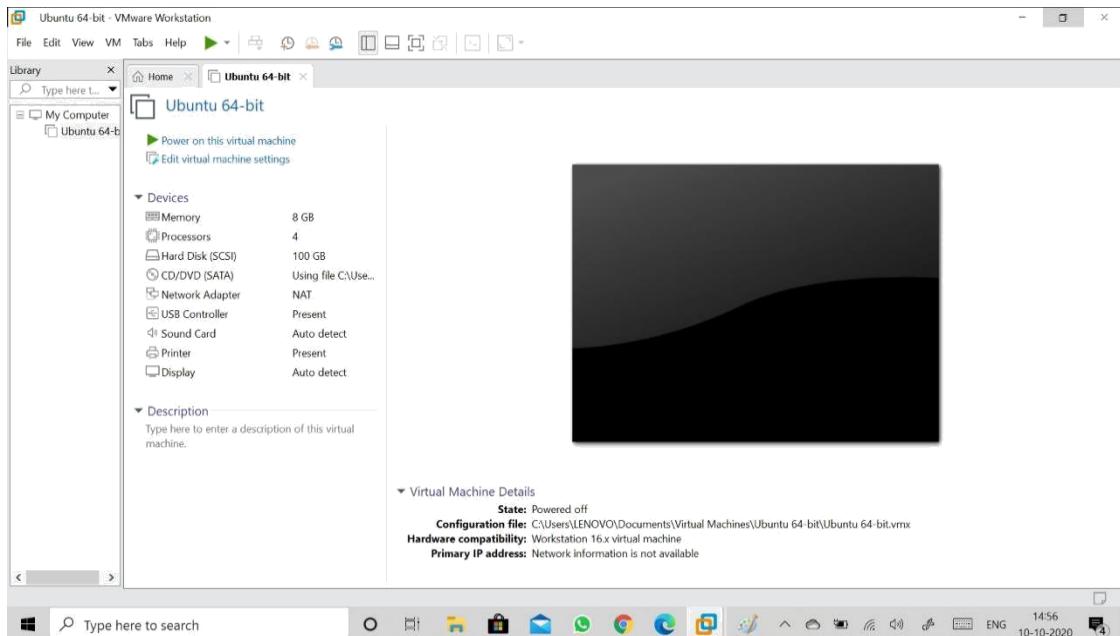
(v) C programming on Linux:

- 1) Open terminal.
- 2) Open gedit by typing “gedit&” on terminal.
- 3) Type the C program and save as “filename.c”.
- 4) Type “ls ” on the terminal to see all files under current folder.
- 5) Confirm that the program file in the directory.
- 6) Type “gcc filename.c” to compile and type “ls” to confirm that a new executable file “a.out” is created.
- 7) Type “./a.out” on the terminal to run the program.









A screenshot of a Text Editor window titled "sample.c". The code displayed is:

```
1 #include<stdio.h>
2 void main()
3 {
4     printf("Hello world\n");
5 }
```

A screenshot of a Linux desktop environment showing a terminal window. The terminal window has a dark background and contains the following text:

```
Activities Terminal Oct 10 10:34
codebind@codebind:~/Desktop$ gcc sample.c
codebind@codebind:~/Desktop$ ./a.out
Hello world
codebind@codebind:~/Desktop$
```

The terminal window is titled "Terminal". The date and time "Oct 10 10:34" are displayed at the top. The command "gcc sample.c" is run to compile the C program "sample.c". The resulting executable "a.out" is then run, outputting the text "Hello world". The terminal window has a standard Linux window title bar with icons for minimize, maximize, and close.

CONCLUSION:

Thus the C compiler was installed in the virtual machine created using Virtual box and the C program was executed successfully.

Ex No:3	INSTALLING AND RUNNING THE GOOGLE APP ENGINE ON WINDOWS
Date:	

AIM:
To install google app engine and create hello world app and other simple web application using Python / Java.

PROCEDURE:

STEP 1: Download and install Python.

STEP 2: Download the google app engine SDK.

STEP 3: Download the Windows Installer.

STEP 4: Double click on the google App Engine Installer click through the installation wizard and it should install the App Engine.

STEP 5: Make a folder for your google App Engine Application and then make a sub folder called “ae-01-trivial”

STEP 6: Using a text editor such as JEdit, 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

Then create a folder called index.py with three lines

Print"Content-type text/Plain"

Print""

Print"Hello there"

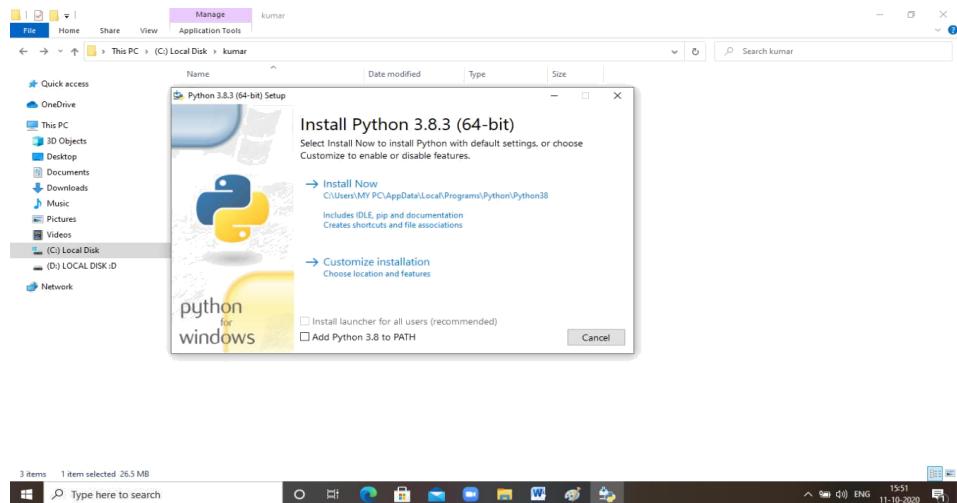
STEP 7: Start the Google App Engine Launcher program using file-> Add existing Application , select the ae-01- trivial folder. Once you have selected the application ; Press Run

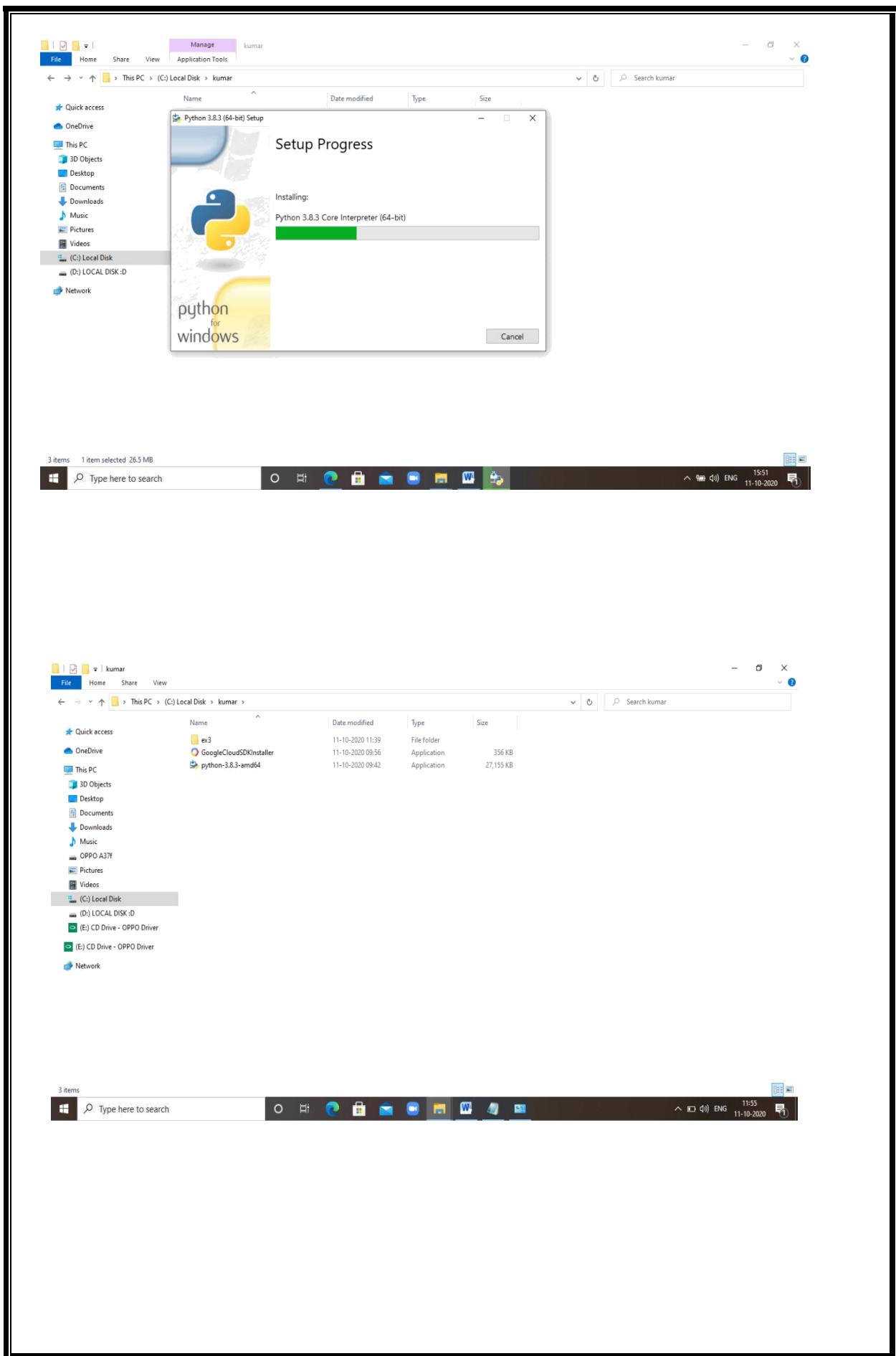
STEP 8: After a few moments, your application will start and the Launcher will show a green icon next to it. Then press Browse do open a browser pointing at your application which is running at <http://localhost:8080/>

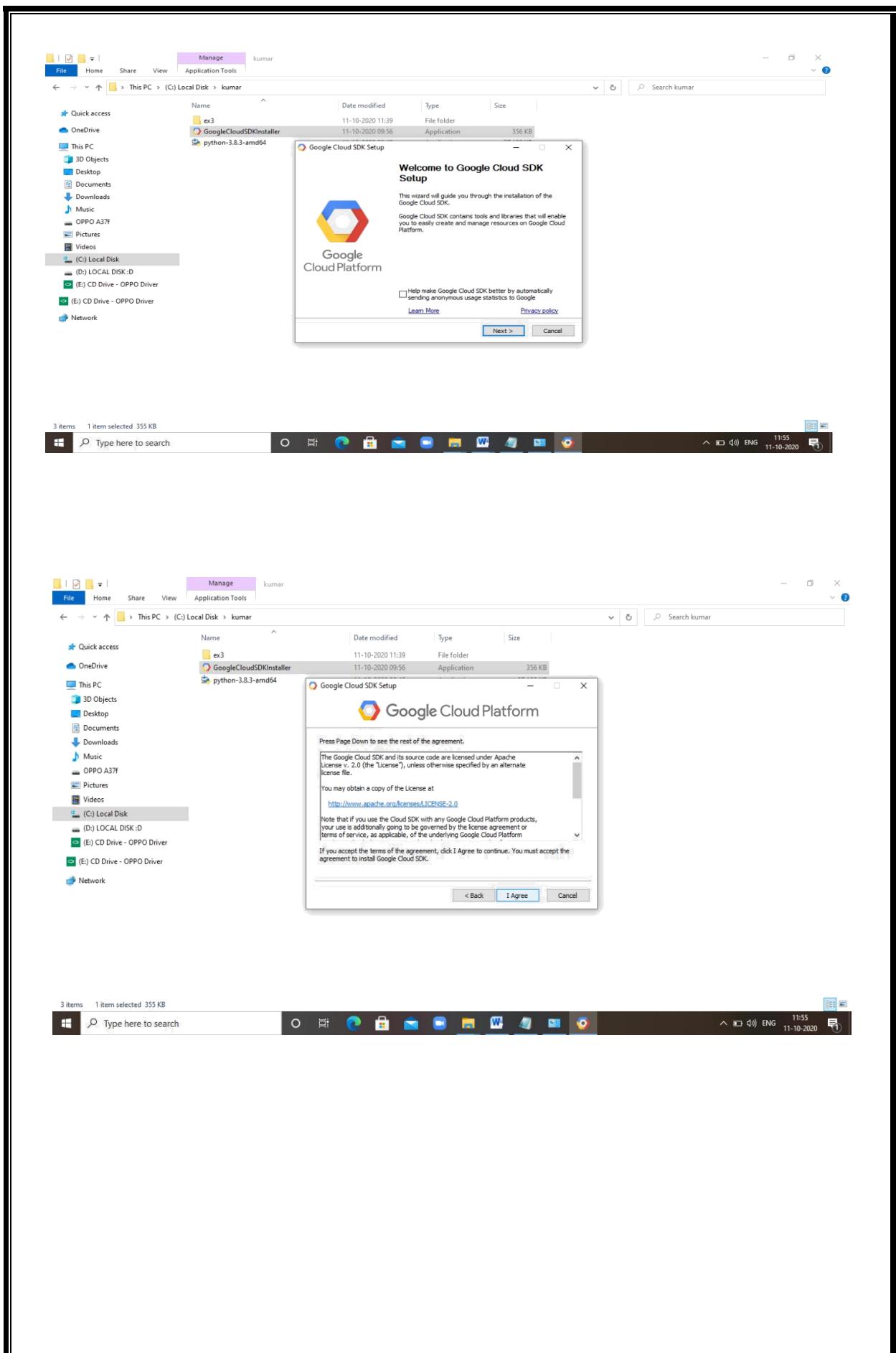
STEP 9: Paste <http://localhost:8080/> into your browser and you should see your applications output.

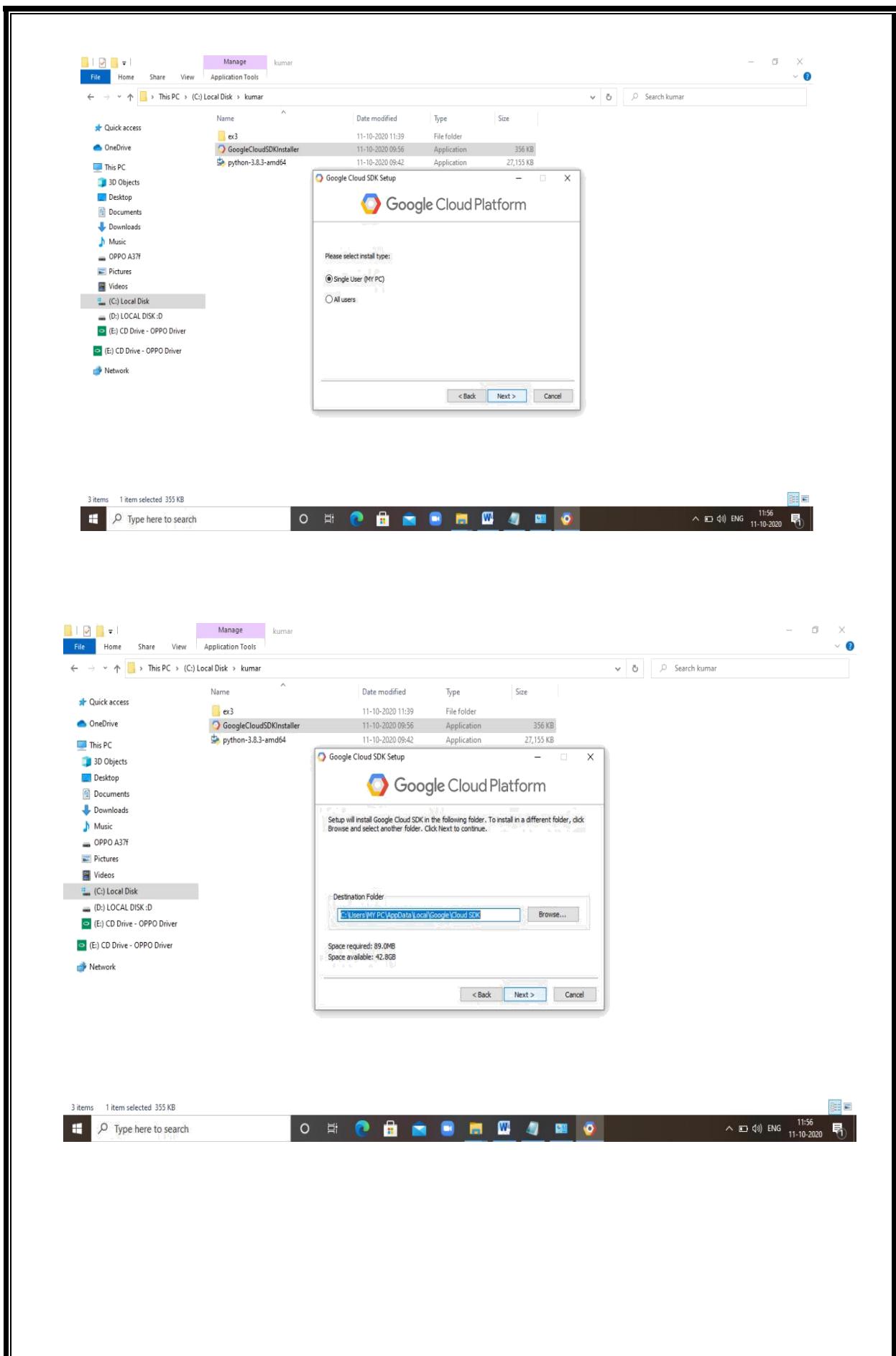
STEP 10: Select your application in the Launcher and press the logs button to bring up a log Window.

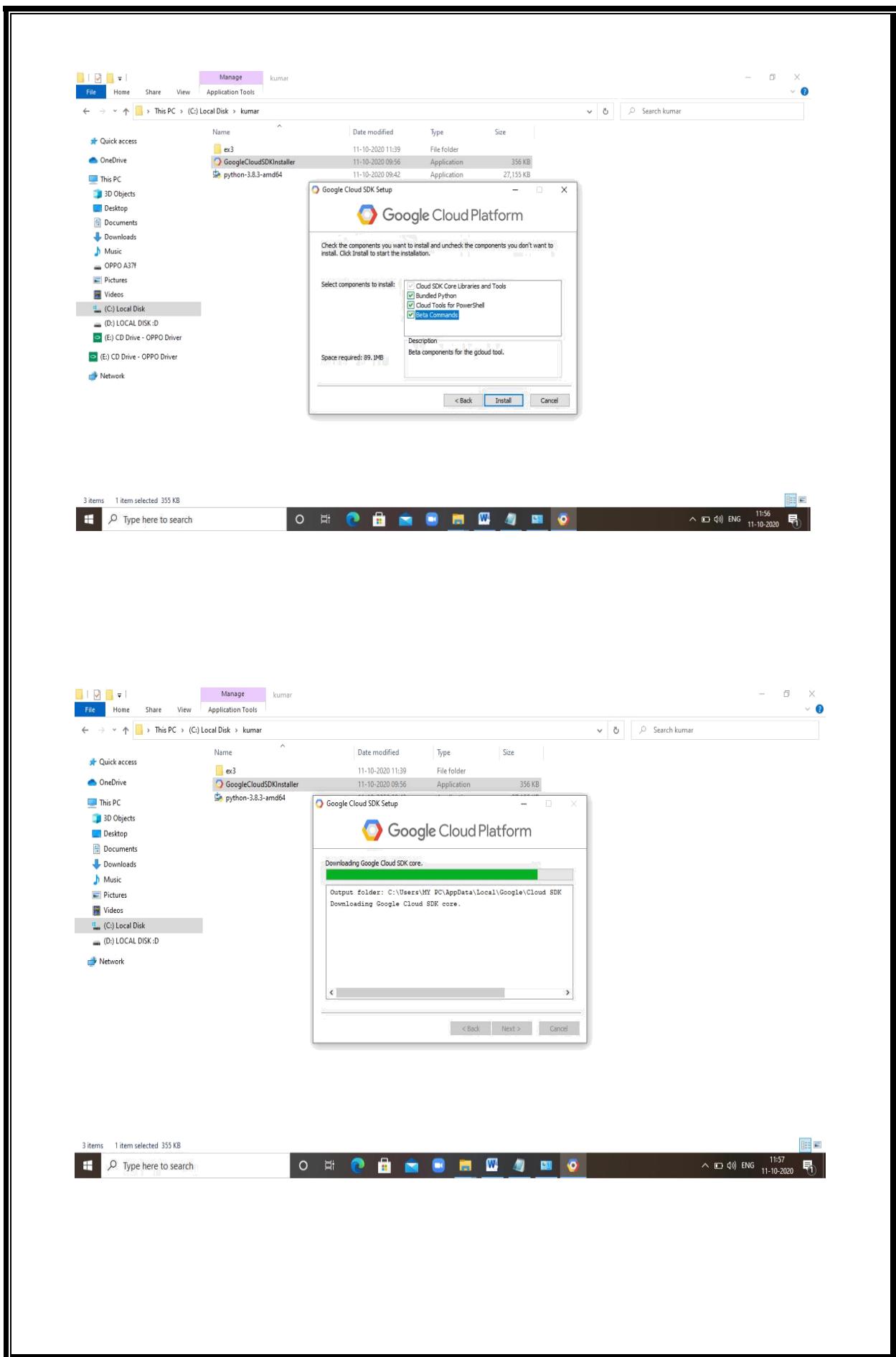
STEP 11: To shut down the server , use the launcher select you application and press the stop button.

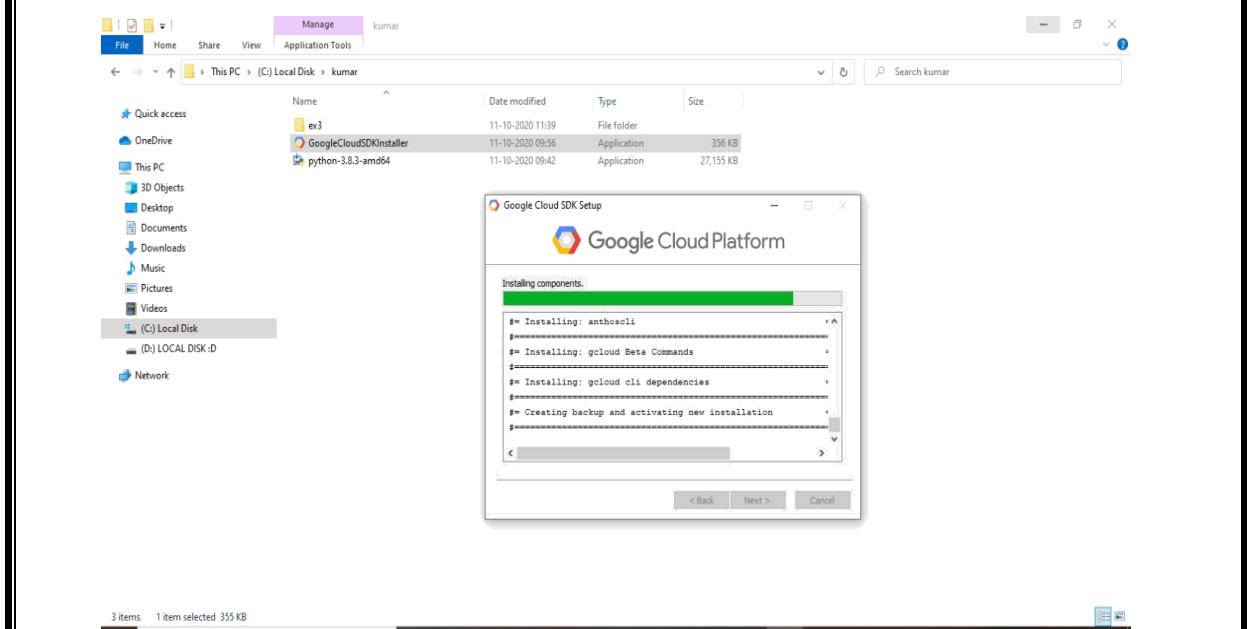
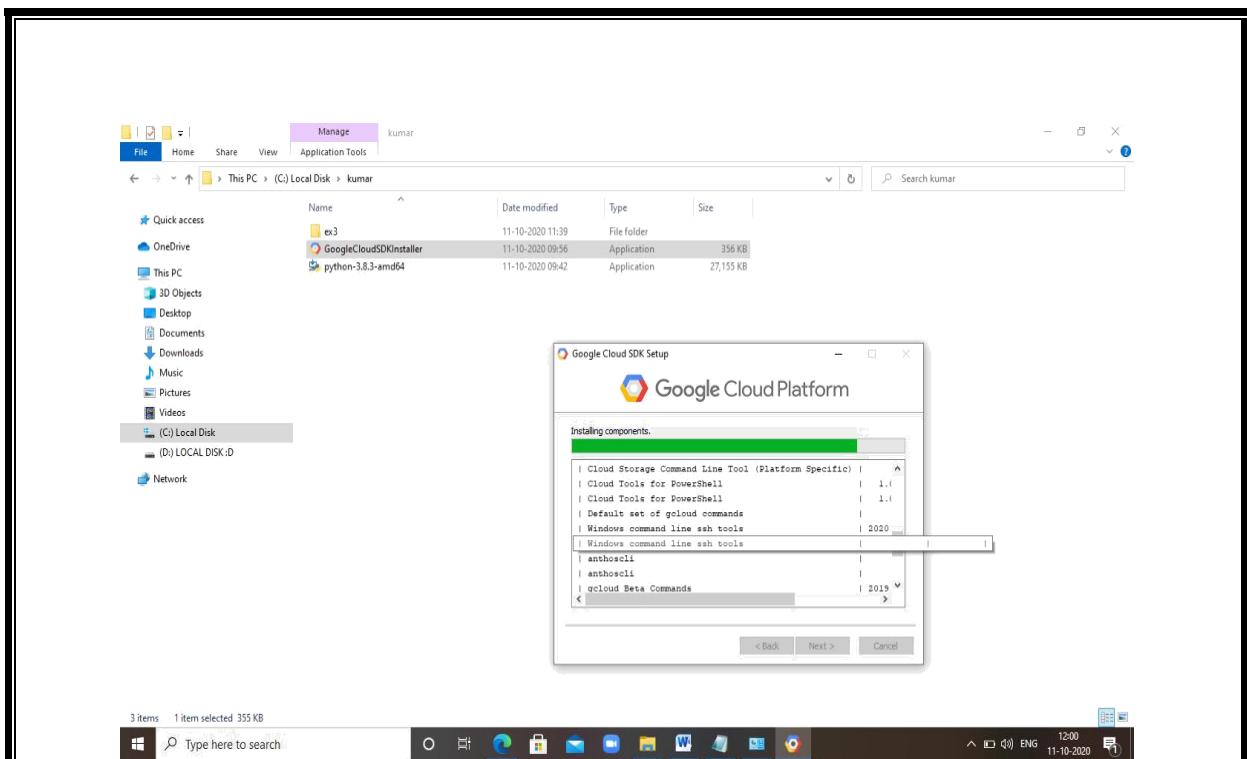


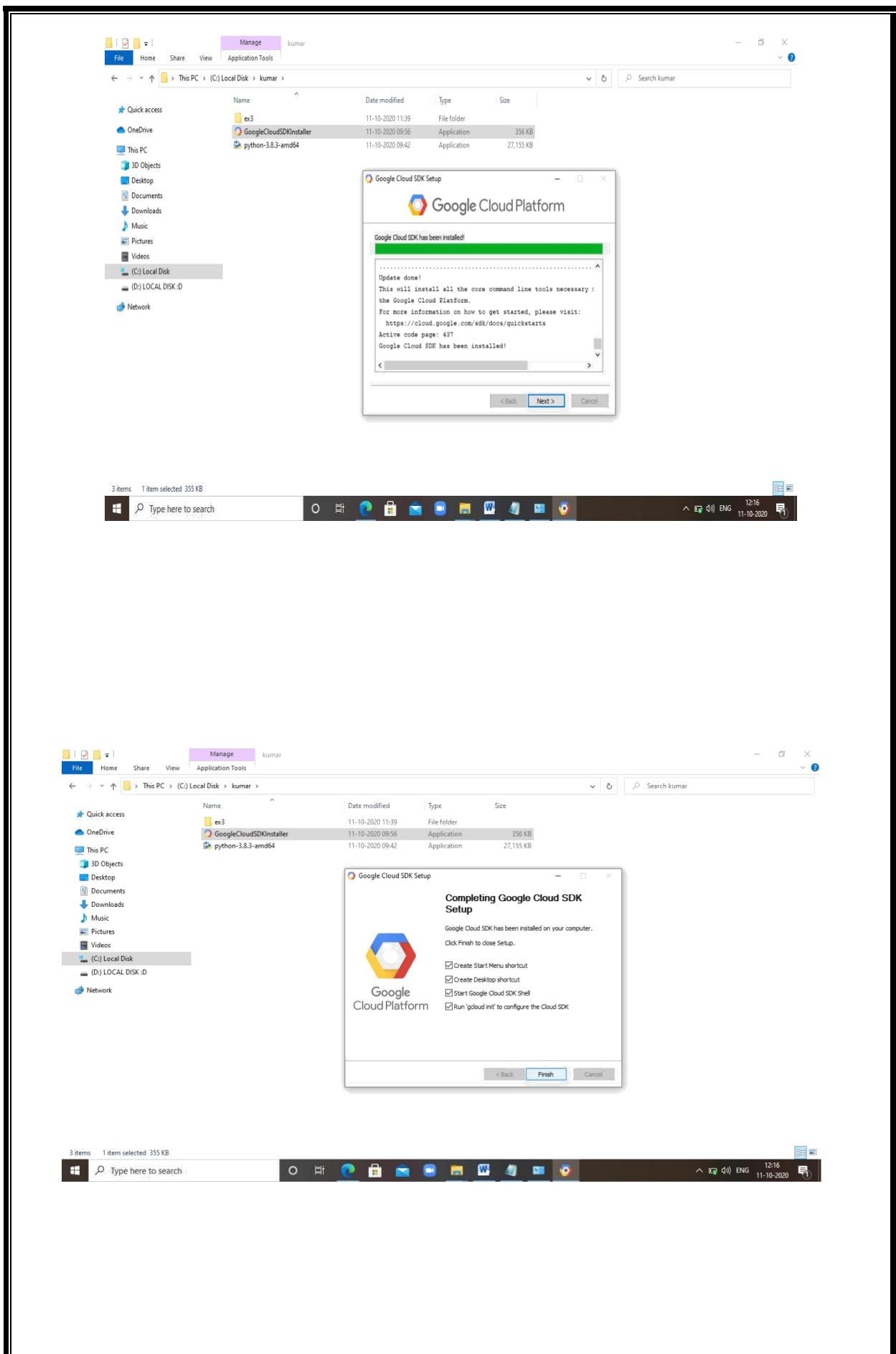


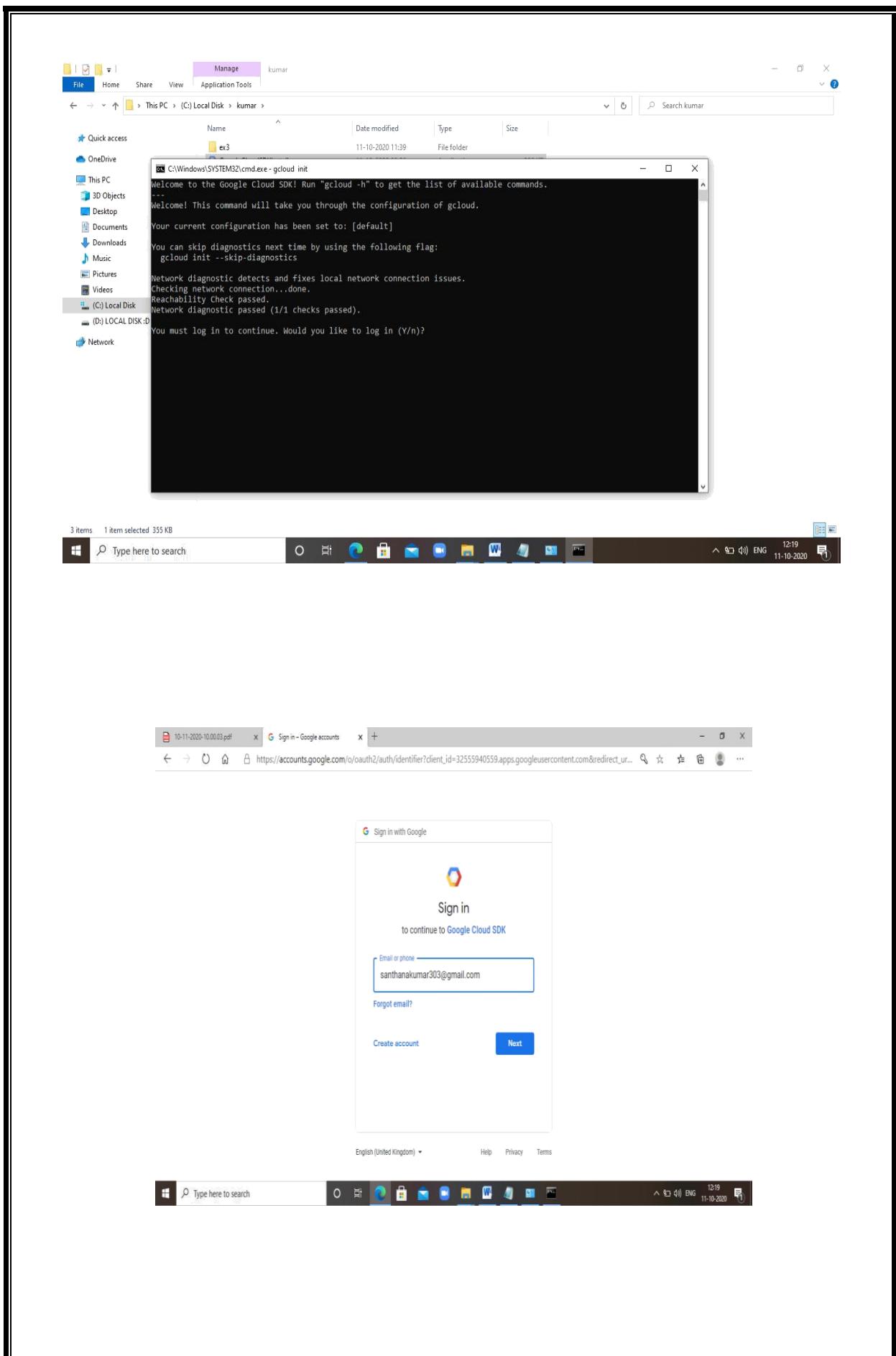


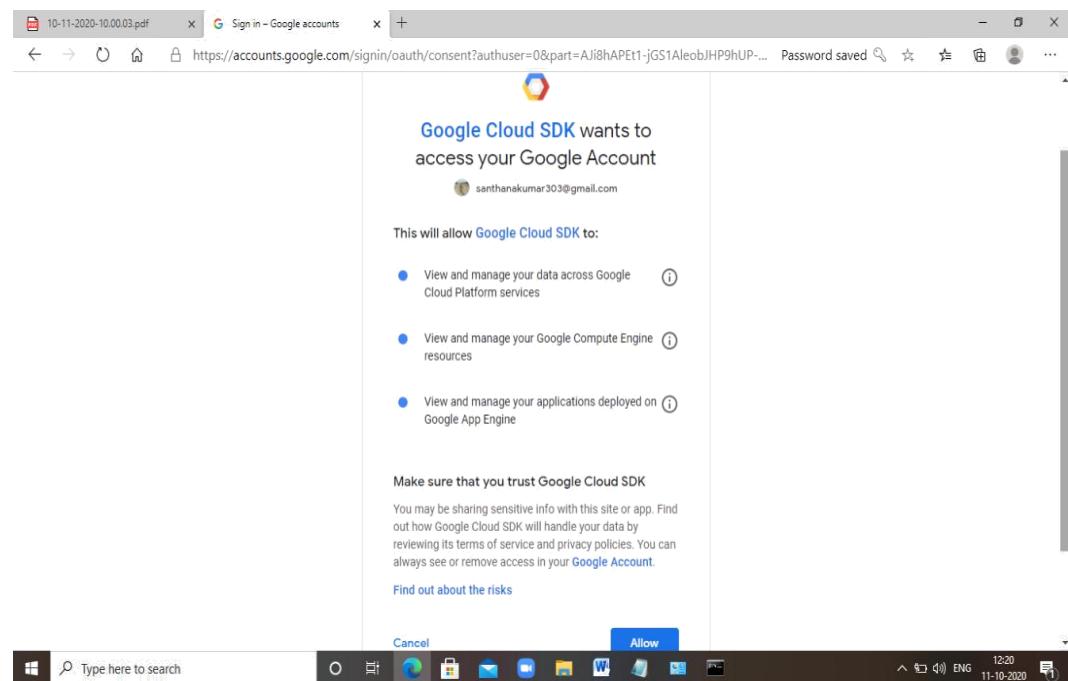
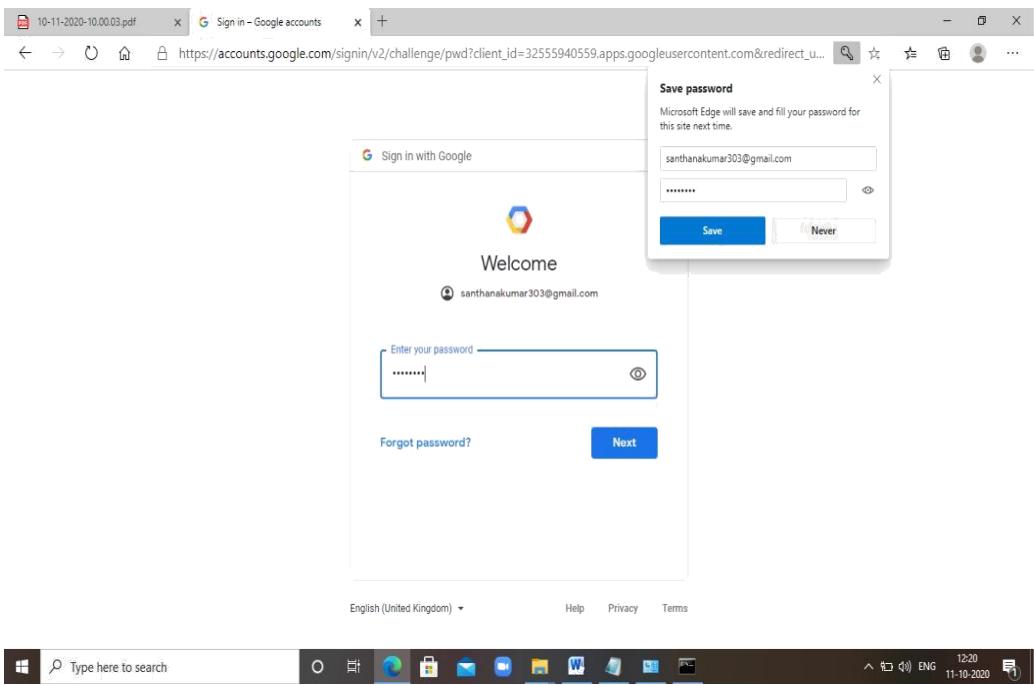


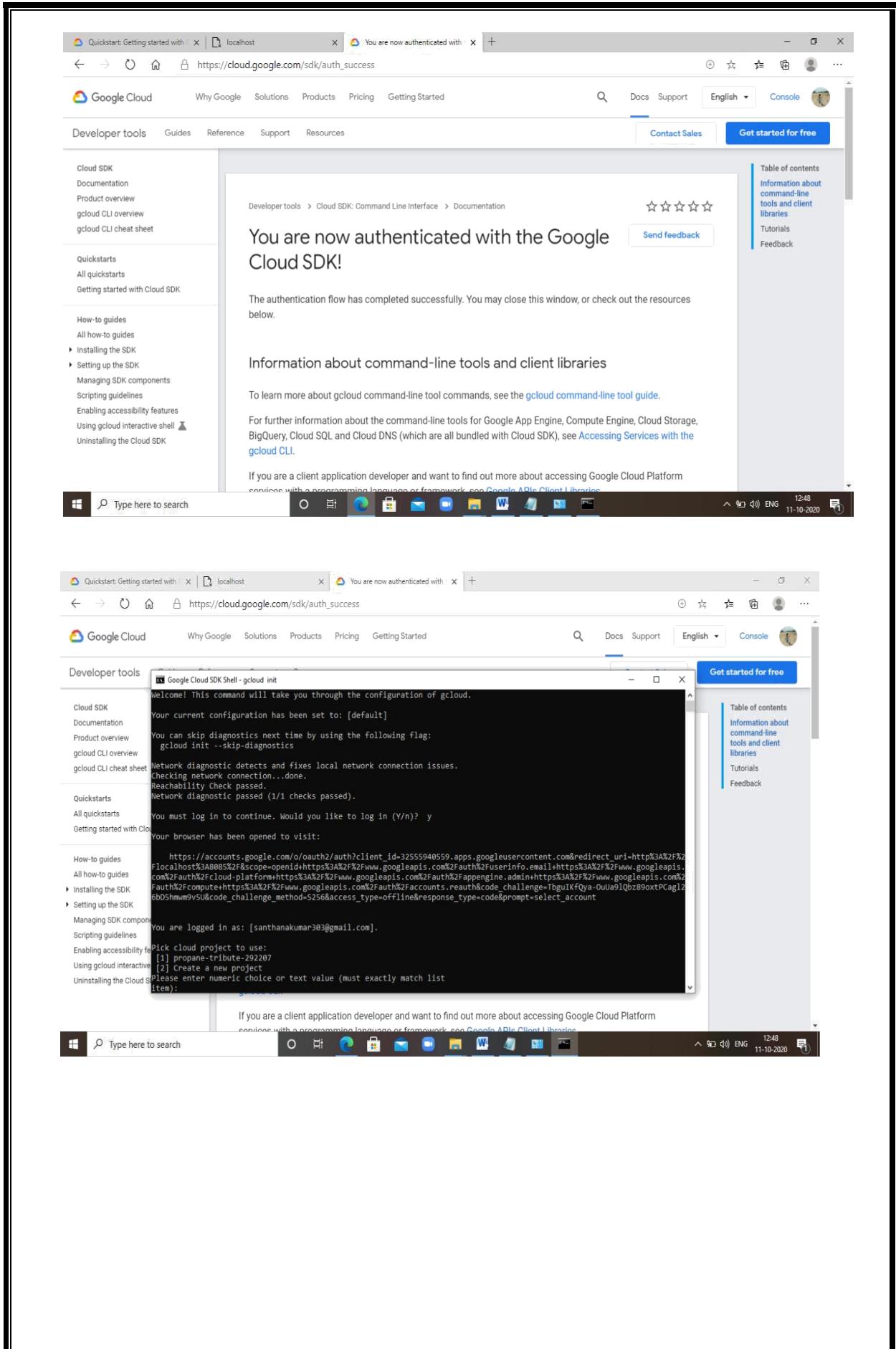






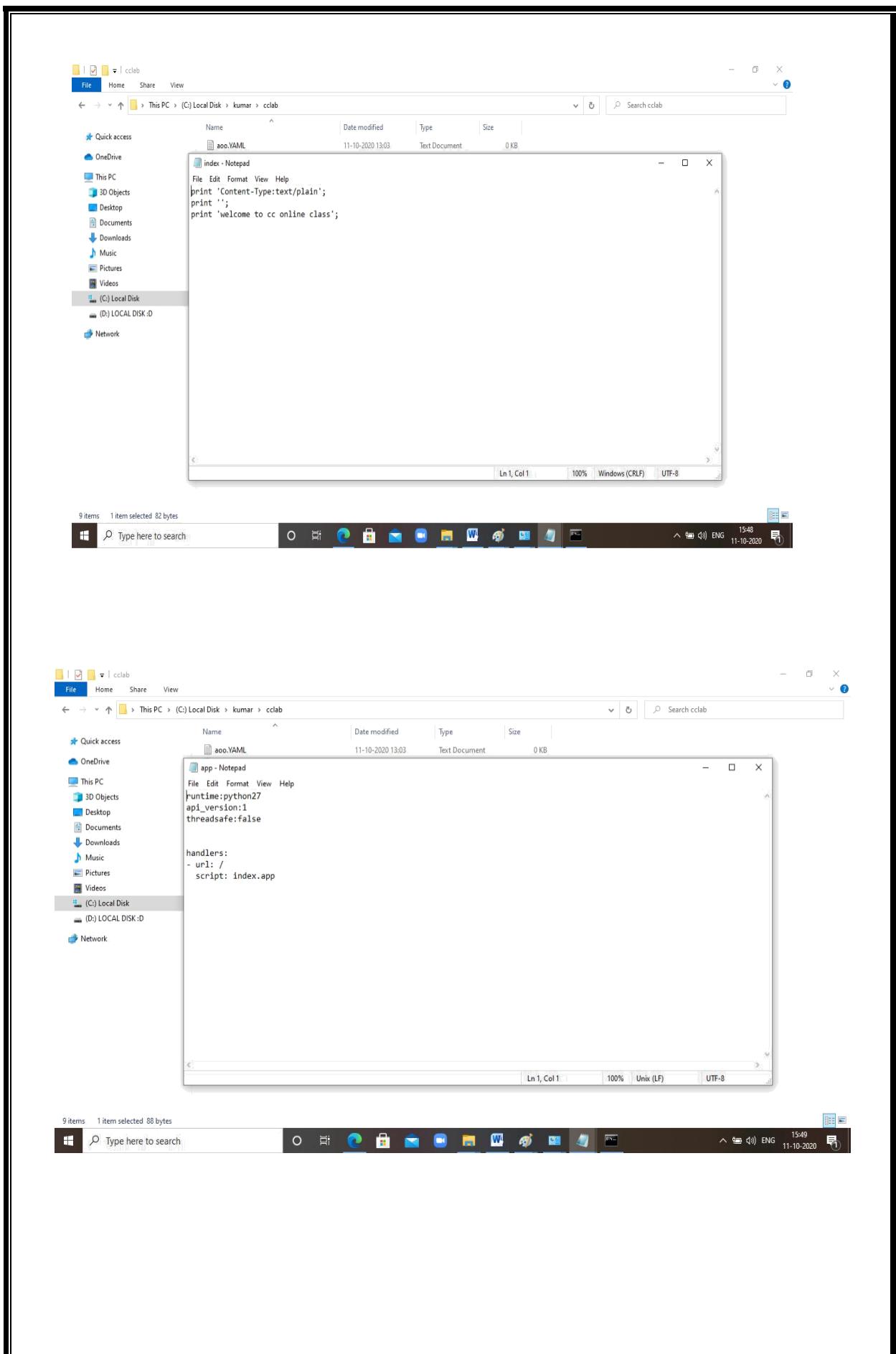


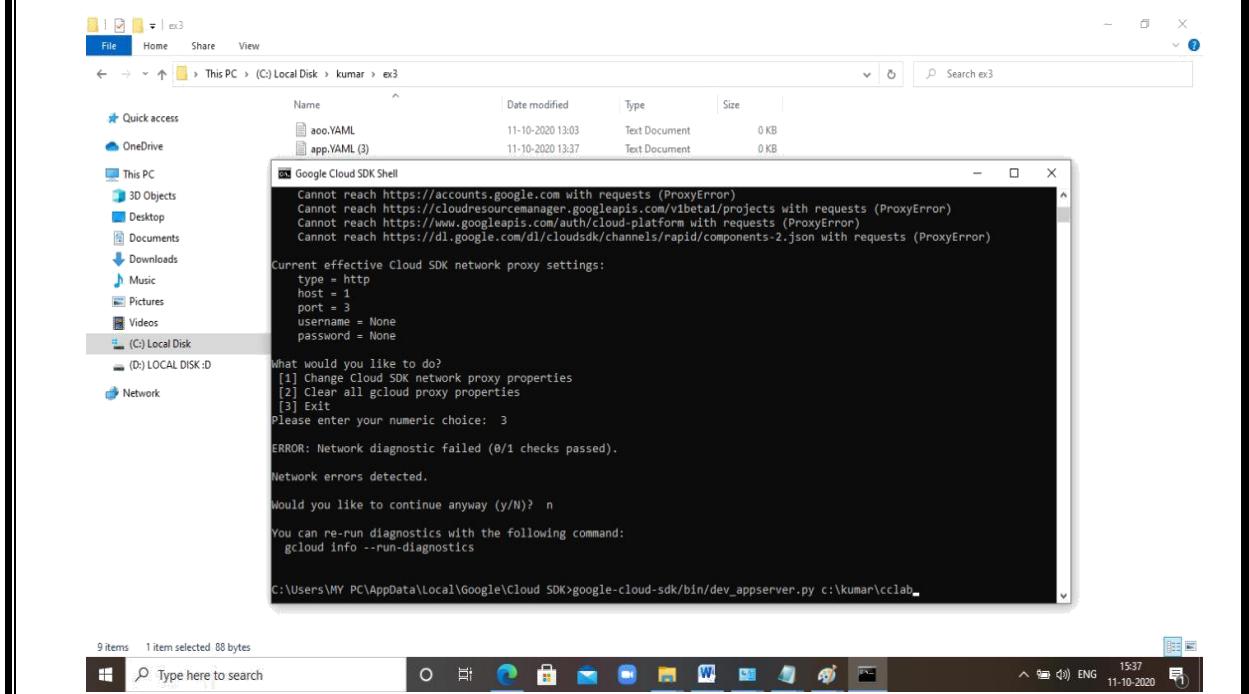
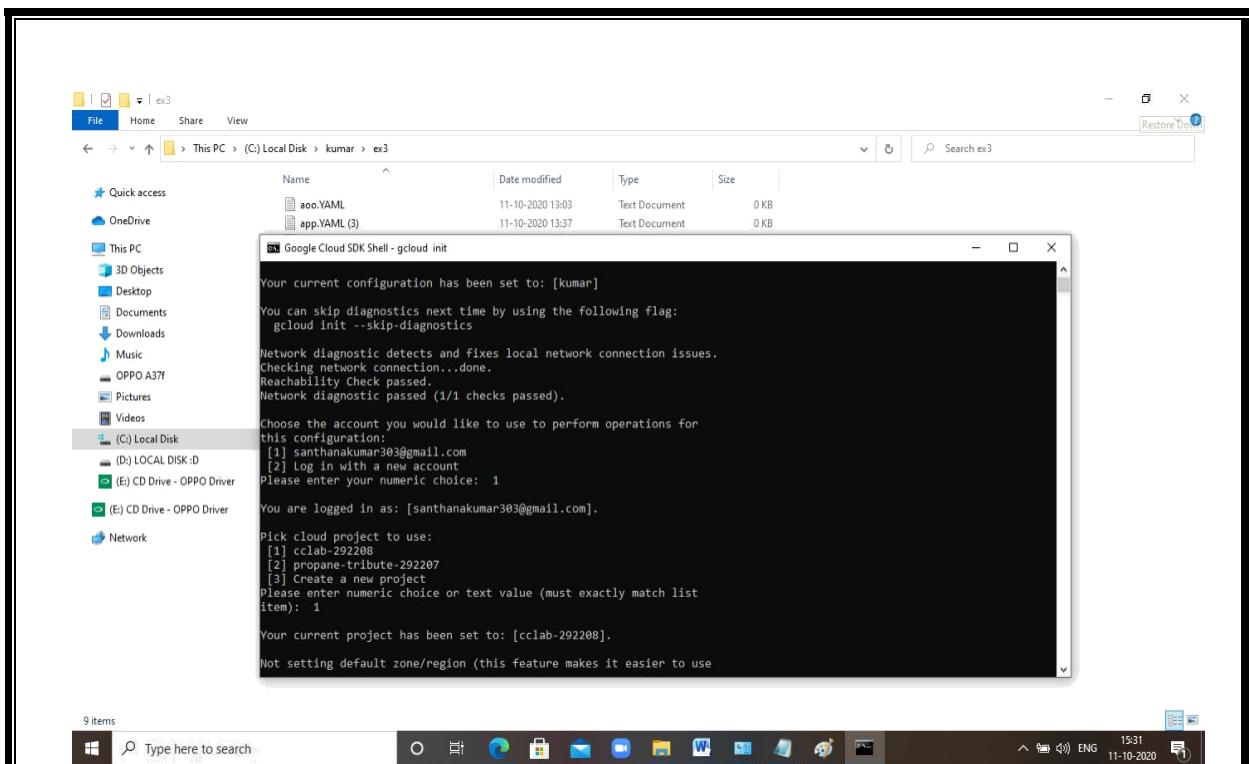


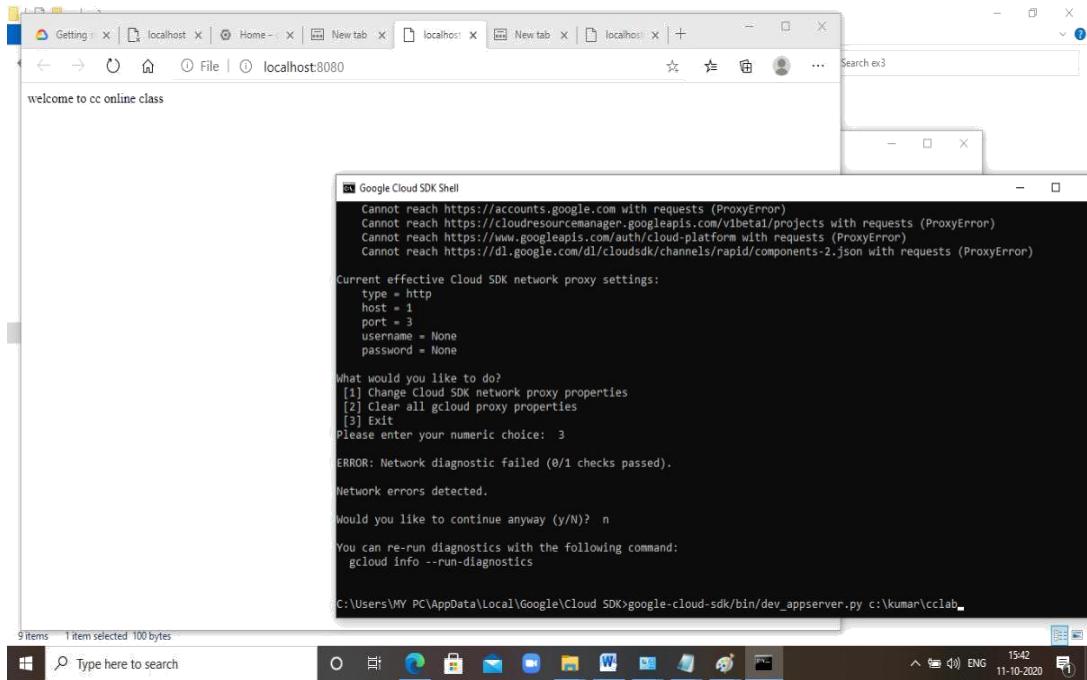


The screenshot shows the Google Cloud Platform dashboard for a new project. At the top, there are four tabs: 'Cloud Computing Services | Go', 'You are now authenticated with ...', 'Getting started with Python | D', and 'Home - Google Cloud Platform'. A message at the top left says 'Your free trial is waiting: activate now to get \$300 credit to explore Google Cloud products' with a 'Learn more' link. On the right, there are 'DISMISS' and 'ACTIVATE' buttons. The main header has a blue bar with 'Google Cloud Platform' and 'Select a project' dropdown, followed by a search bar 'Search products and resources'. The left sidebar lists various services: Home, Marketplace, Billing, APIs & Services, Support, IAM & Admin, Getting started, Security, and Anthos. Under COMPUTE, there are App Engine and Compute Engine. The central dashboard area displays a message 'To view this page, select a project.' and a section for 'Select a recent project' showing 'My First Project' (Project ID: propane-tribute-292207, Organisation: No organization, Accessed 1 hour ago). The bottom status bar shows the URL https://console.cloud.google.com/projectselector2/home/dashboard?_ga=2.39839782.460881051.1602400179-343462229.1602400179, the time 14:24, and the date 11-10-2020.

The screenshot shows the Google Cloud Platform dashboard for a project named 'cclab'. The URL in the address bar is https://console.cloud.google.com/home/dashboard?_ga=2.39839782.460881051.1602400179-343462229.1602400179&project=cclab-292208. The dashboard features a 'DASHBOARD' tab, which is currently selected, and 'ACTIVITY' and 'RECOMMENDATIONS' tabs. A message at the top says 'How Google Cloud is helping during COVID-19' with a 'Learn more' link. The 'DISMISS' button is visible. The left sidebar is identical to the first screenshot. The central dashboard includes sections for 'Project info' (Project name: cclab, Project ID: cclab-292208, Project number: 448473482634), 'API APIs' (Requests (requests/sec) chart from 13:30 to 14:15), 'Google Cloud Platform status' (All services normal), and 'Monitoring' (Set up alerting policies, Create uptime checks, View all dashboards, Go to Monitoring). A tooltip at the bottom center says 'Now viewing project "cclab" in organisation "No organisation"'.







welcome to cc online class

```
Google Cloud SDK Shell
Cannot reach https://accounts.google.com with requests (ProxyError)
Cannot reach https://clouresourcemanager.googleapis.com/v1beta1/projects with requests (ProxyError)
Cannot reach https://www.googleapis.com/auth/cloud-platform with requests (ProxyError)
Cannot reach https://dl.google.com/dl/cloudsdk/channels/rapid/components-2.json with requests (ProxyError)

Current effective Cloud SDK network proxy settings:
  type = http
  host = 1
  port = 3
  username = None
  password = None

What would you like to do?
[1] Change Cloud SDK network proxy properties
[2] Clear all gcloud proxy properties
[3] Exit
Please enter your numeric choice: 3

ERROR: Network diagnostic failed (0/1 checks passed).

Network errors detected.

Would you like to continue anyway (Y/N)? n
You can re-run diagnostics with the following command:
gcloud info --run-diagnostics

C:\Users\MY PC\AppData\Local\Google\Cloud SDK\bin>google-cloud-sdk/bin/dev_appserver.py c:\kumar\cclab\
```

CONCLUSION:

Thus the google app engine was installed and a simple application using python was created and executed successfully.

EX NO:4	USE GAE LAUNCHER TO LAUNCH THE WEB APPLICATIONS
DATE:	

AIM:

To use GAE launcher to launch the web applications.

PROCEDURE:

STEP 1: Create a new cloud console project or retrieve the project ID of an existing project.

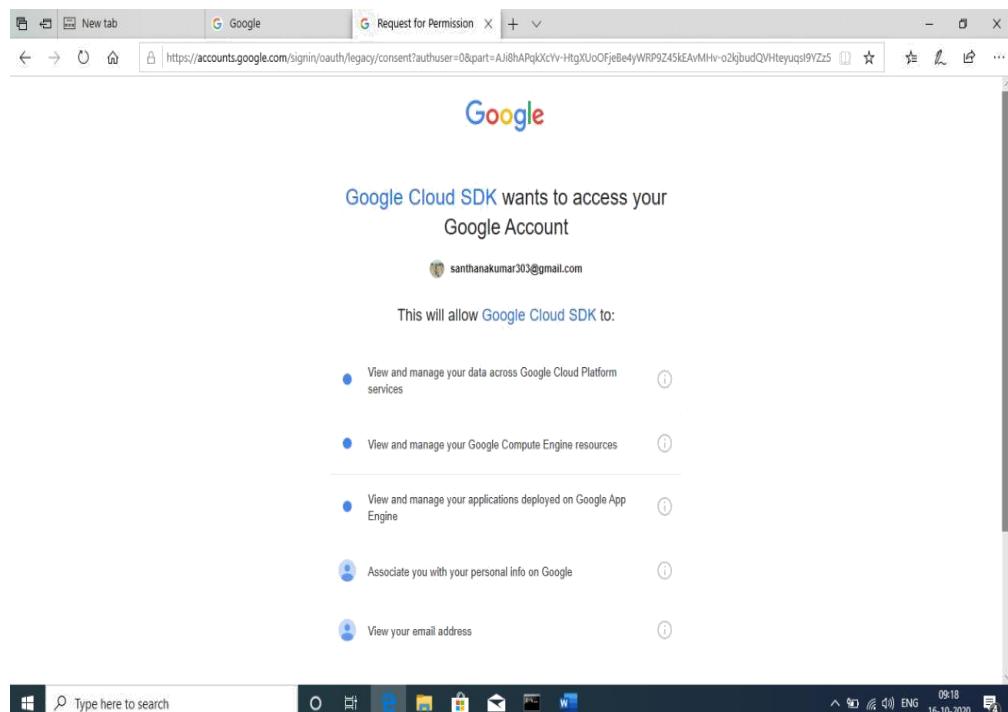
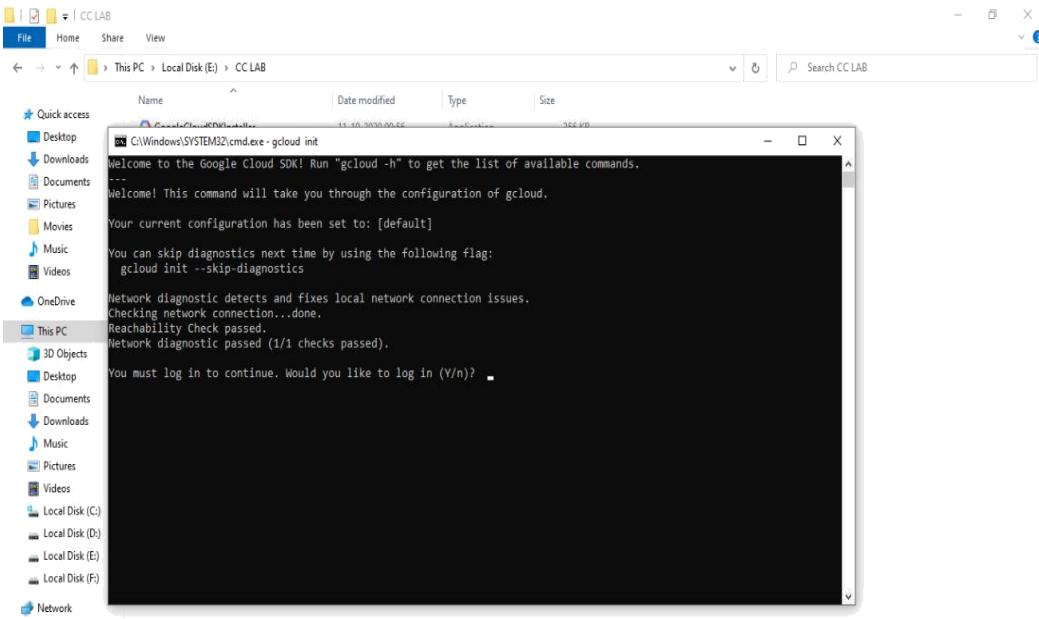
STEP 2: Install and then initialize the Google Cloud SDK.

STEP 3: The basic structure for the project consisted of two files, namely , app.yaml which continue the configure settings of your AppEngine application Next , the www.directory to store all of your static files, such as HTML, CSS, image and Java Script.

STEP 4: In the google SDK shell, initialize the cloud. Then deploy your application to AppEngine using gcloud app deploy.

. **STEP 5:** To view your application and launch it use the command gcloud app browse.

STEP 6: Now we can view the app with the corresponding link shown in the shell like https://PROJECT_ID.REGION_ID.r.appspot.com.



This will allow Google Cloud SDK to:

- View and manage your data across Google Cloud Platform services
- View and manage your Google Compute Engine resources
- View and manage your applications deployed on Google App Engine
- Associate you with your personal info on Google
- View your email address

By clicking "Allow", you allow this app and Google to use your information in accordance with their respective terms of service and privacy policies. You can change this and other Account Permissions at any time.

Type here to search

09:18 ENG 16-10-2020

You are now authenticated X + v https://cloud.google.com/sdk/auth_success

Google Cloud Why Google Solutions Products Pricing Getting Started

Developer tools Guides Reference Support Resources Contact Sales Get started for free

Cloud SDK Documentation Product overview gcloud CLI overview gcloud CLI cheat sheet

Quickstarts All quickstarts Getting started with Cloud SDK

How-to guides All how-to guides ▾ Installing the SDK ▾ Setting up the SDK Managing SDK components Scripting guidelines Enabling accessibility features Using gcloud interactive shell ▾ Uninstalling the Cloud SDK

Developer tools > Cloud SDK: Command Line Interface > Documentation

You are now authenticated with the Google Cloud SDK!

The authentication flow has completed successfully. You may close this window, or check out the resources below.

Information about command-line tools and client libraries

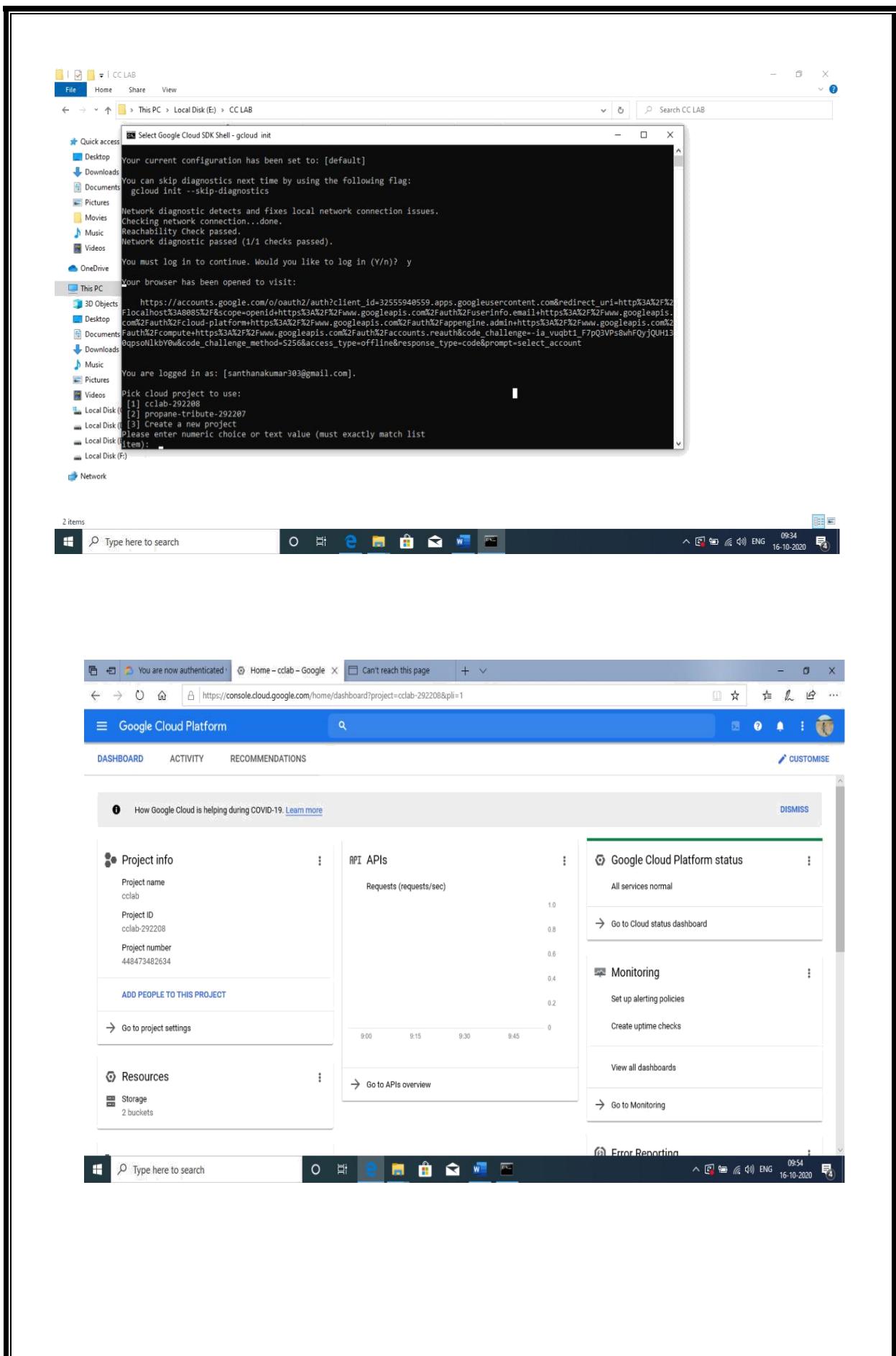
To learn more about gcloud command-line tool commands, see the [gcloud command-line tool guide](#).

For further information about the command-line tools for Google App Engine, Compute Engine, Cloud Storage, BigQuery, Cloud SQL and Cloud DNS (which are all bundled with Cloud SDK), see [Accessing Services with the gcloud CLI](#).

If you are a client application developer and want to find out more about accessing Google Cloud Platform services with a programming language or framework, see [Google APIs Client Libraries](#).

Type here to search

09:34 ENG 16-10-2020



The screenshot shows a Windows desktop environment with a browser window open to the Google Cloud IAM & Admin settings page. A modal dialog box titled "Select a project" is displayed, listing recent projects. The project "cclab" is selected, indicated by a checkmark. Other visible projects include "My First Project". The dialog has "NEW PROJECT" and "OPEN" buttons at the bottom.

Select a project

Search projects and folders

RECENT ALL

Name	ID
cclab	cclab-292208
My First Project	propane-tribute-292207

CANCEL OPEN

Google Cloud Platform

You have 23 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

MANAGE QUOTAS

Project name * CC LAB ?

Project ID: cc-lab-4-292704. It cannot be changed later. [EDIT](#)

Location * No organisation [BROWSE](#)

Parent organisation or folder

CREATE CANCEL

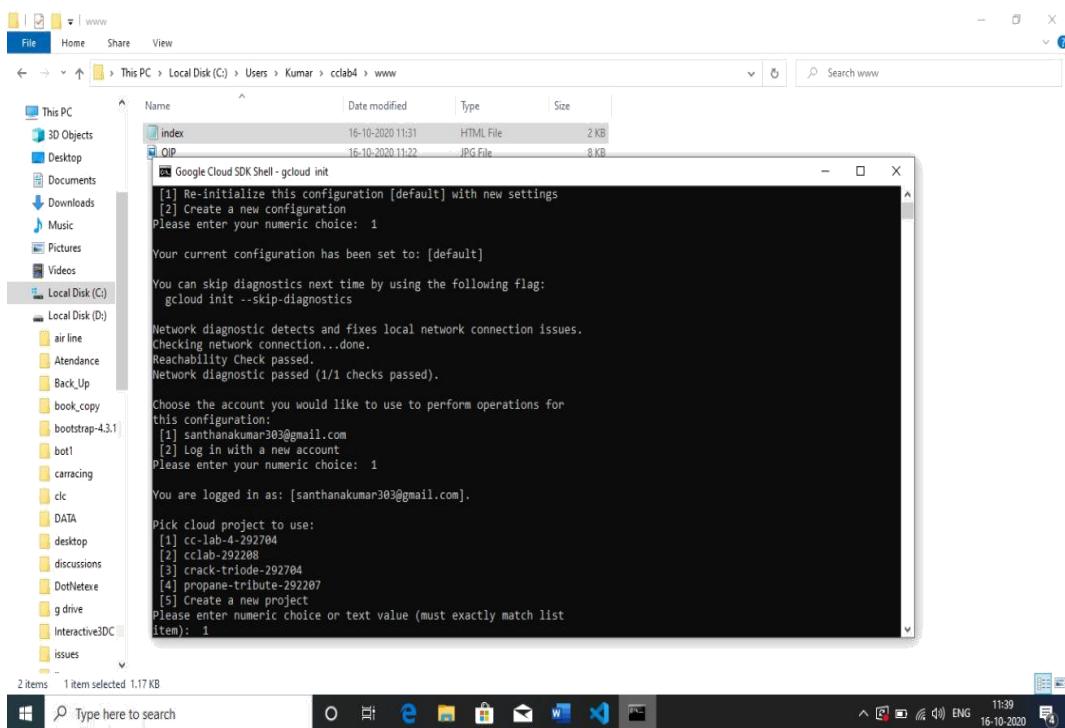
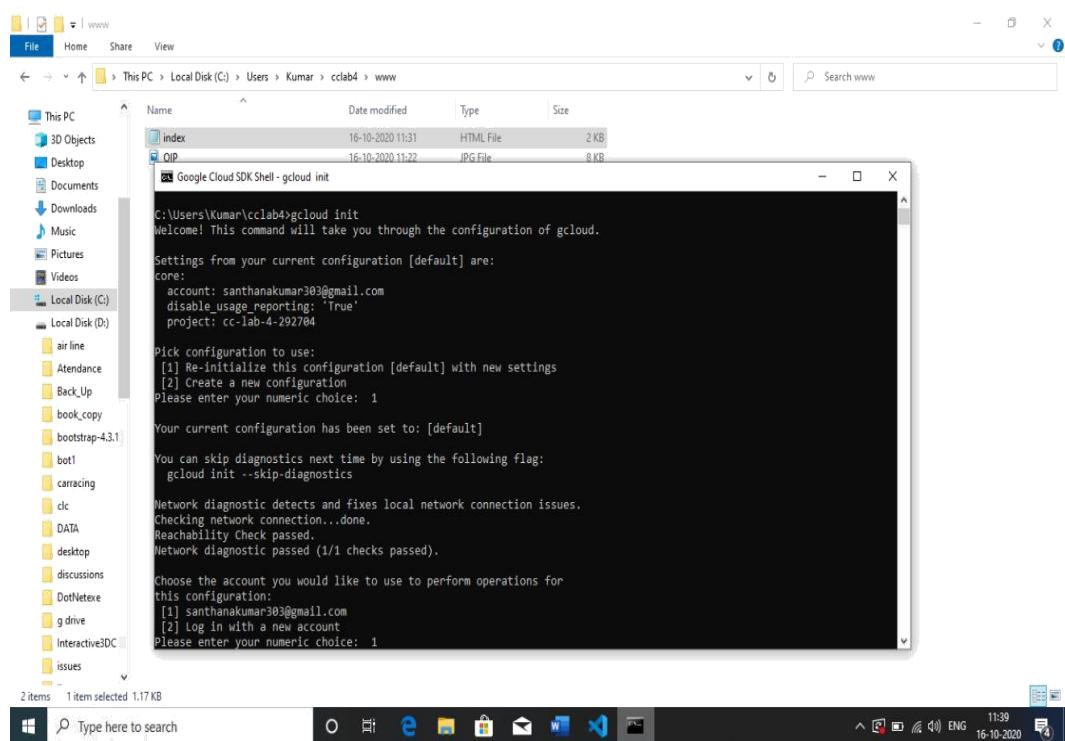
Windows taskbar: Type here to search, Start button, File Explorer, Task View, Microsoft Edge, File, Mail, Photos, Word, Powerpoint, Taskbar icons, Date/Time: 09:56, 16-10-2020, Battery icon.

The screenshot shows the Google Cloud Platform Dashboard for project 'CC LAB 4'. A prominent notification bar at the top left says 'You are now authenticated!' and 'Your free trial is waiting: activate now to get \$300 credit to explore Google Cloud products [Learn more](#)'. On the right, there are 'DISMISS' and 'ACTIVATE' buttons. The dashboard features several cards: 'Project info' (with a 'How Google Cloud is helping during COVID-19' message), 'Resources' (noting 'This project has no resources'), 'Trace' (noting 'No trace data from the last 7 days'), 'Getting started' (with a link to 'Get started with Trace'), 'App Engine', 'Compute Engine', 'SQL', and 'APIs' (noting 'Now viewing project 'CC LAB 4' in organisation 'No organisation''. A sidebar on the right lists 'Notifications' with three items: 'Create Project: CC LAB 4' (Just now, SELECT PROJECT), 'Create Project: cclab' (5 days ago, SELECT PROJECT), and 'Create Project: My First Project' (5 days ago, SELECT PROJECT). Below the notifications are links to 'SEE ALL ACTIVITIES', 'Set up alerting policies', 'Create uptime checks', and 'View all dashboards'. At the bottom of the dashboard are standard Windows taskbar icons.

This screenshot shows a Windows File Explorer window displaying the contents of the 'www' folder on the 'Local Disk (C:)'. Inside the folder, there are two files: 'index' (HTML File, 16-10-2020 11:31, 2 KB) and '.OIP' (JPG File, 16-10-2020 11:22, 8 KB). Overlaid on the file list is a terminal window titled 'Google Cloud Shell - gcloud init'. The terminal output shows the following steps:

```
Welcome to the Google Cloud SDK! Run "gcloud -h" to get the list of available commands.  
...  
C:\Users\Kumar\AppData\Local\Google\Cloud SDK>cd ..  
C:\Users\Kumar\AppData\Local>cd ..  
C:\Users\Kumar>cd cclab4  
C:\Users\Kumar\cclab4>gcloud init  
Welcome! This command will take you through the configuration of gcloud.  
Settings from your current configuration [default] are:  
core:  
  account: santhanakumar30@gmail.com  
  disable_usage_reporting: 'True'  
  project: cc-lab-4-292704  
Pick configuration to use:  
[1] Re-initialize this configuration [default] with new settings  
[2] Create a new configuration  
Please enter your numeric choice: 1
```

The Windows taskbar at the bottom includes icons for Start, Task View, File Explorer, Edge, File Explorer, Mail, and File Explorer. The system tray shows the date and time as 16-10-2020 11:38.



Screenshot 1: Google Cloud SDK Shell - gcloud app deploy

```

index 16-10-2020 11:31 HTML File 2 KB
OIP 16-10-2020 11:22 JPG File 8 KB

Run 'gcloud topic configurations' to learn more.

Some things to try next:
* Run 'gcloud --help' to see the Cloud Platform services you can interact with. And run 'gcloud help COMMAND' to get help on any gcloud command.
* Run 'gcloud topic --help' to learn about advanced features of the SDK like arg files and output formatting

C:\Users\Kumar\cclab4>gcloud app deploy

descriptor: [C:\Users\Kumar\cclab4\app.yaml]
source: [C:\Users\Kumar\cclab4]
target project: [cc-lab-4-292704]
target service: [default]
target version: [20201016t114033]
target url: [https://cc-lab-4-292704.firebaseioapp.com]

Do you want to continue (Y/n)? y

Beginning deployment of service [default]...
=====
#-----#
# Uploading 2 files to Google Cloud Storage =#
#-----#
File upload done.
WARNING: You will be unable to deploy applications after November 30, 2019 without adding a billing instrument to your project. Please add one at https://console.cloud.google.com/billing/linkedaccount?cc-lab-4-292704

Updating service [default]...

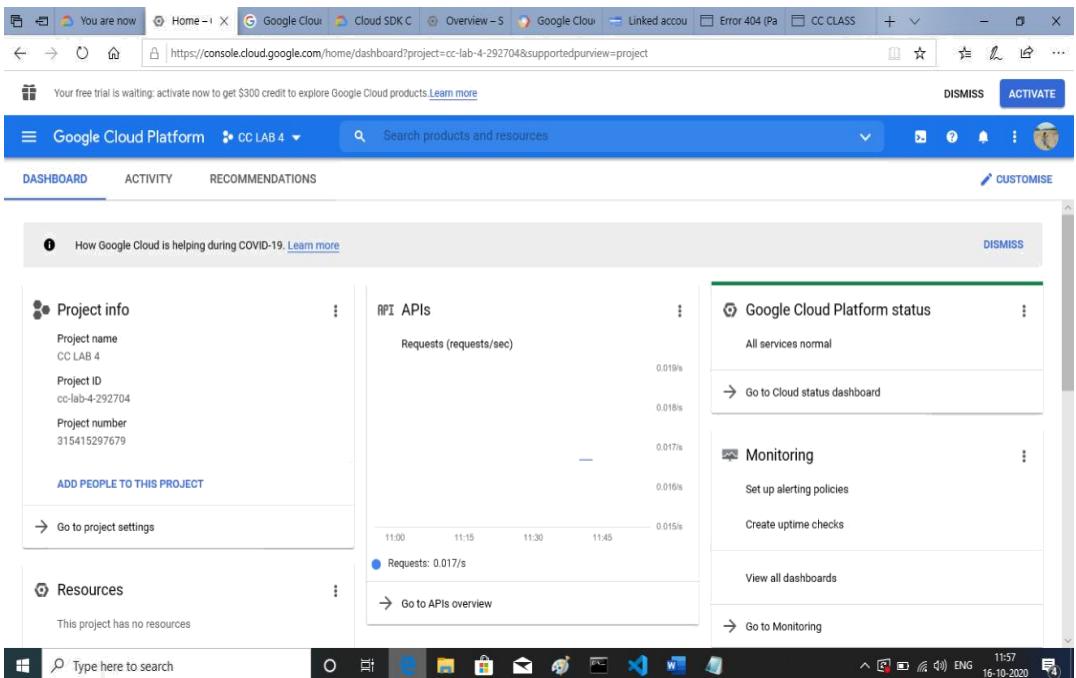
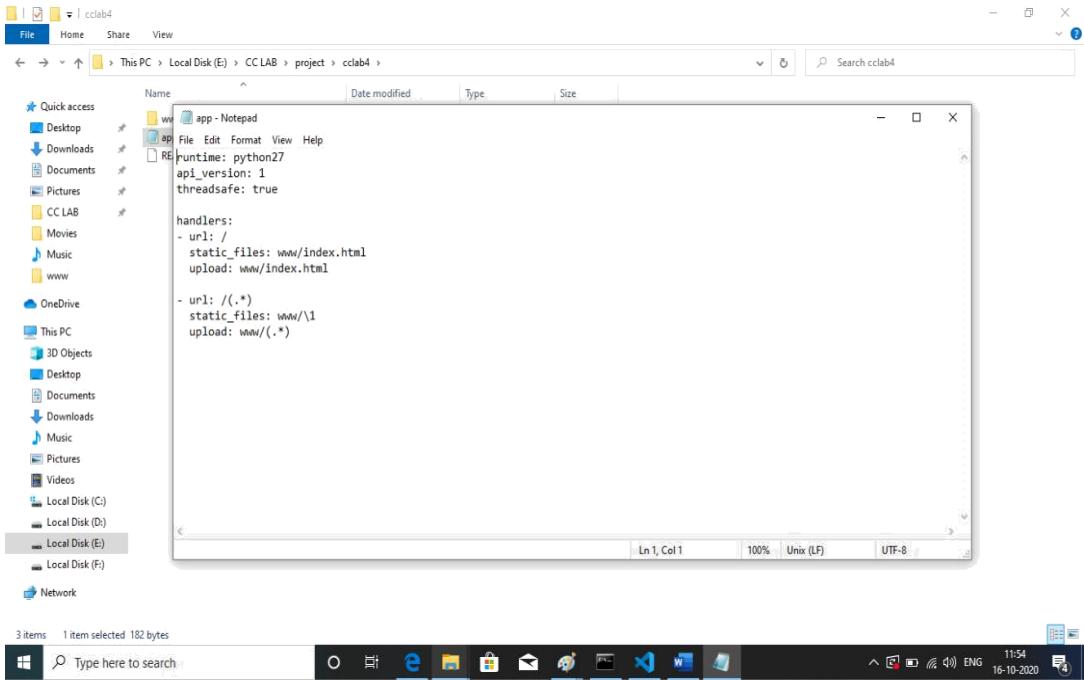
```

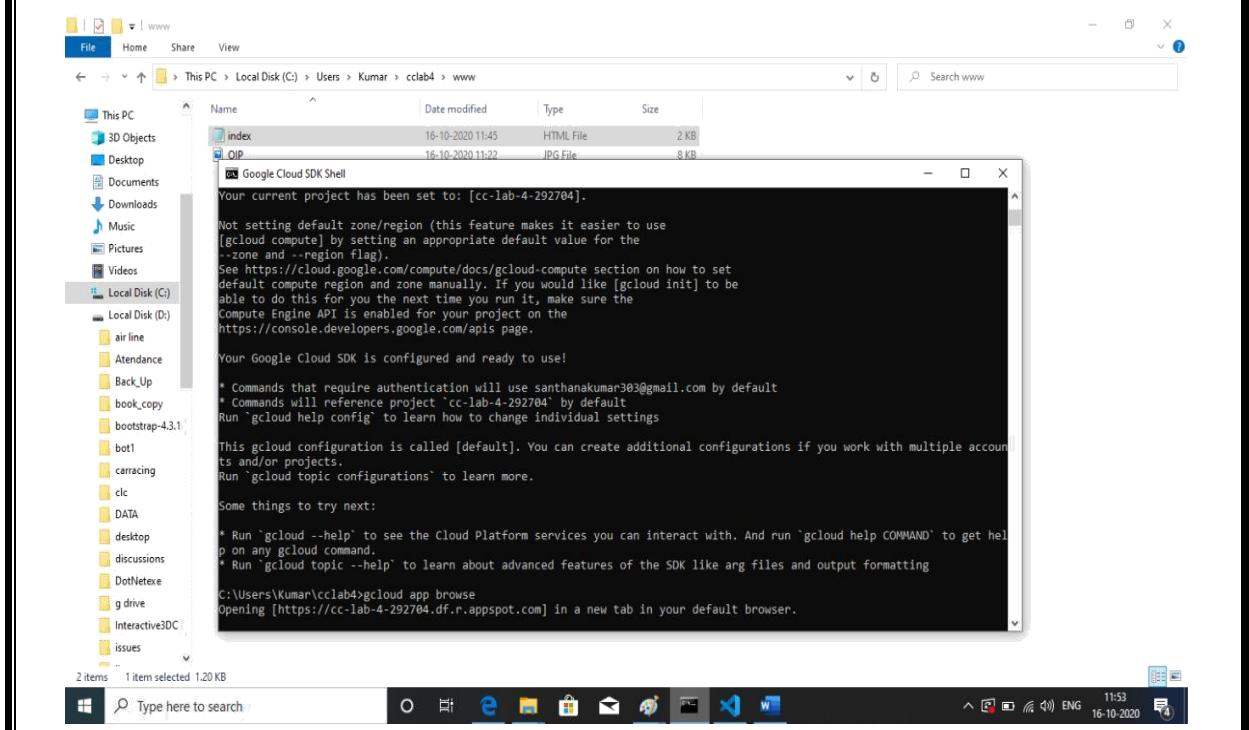
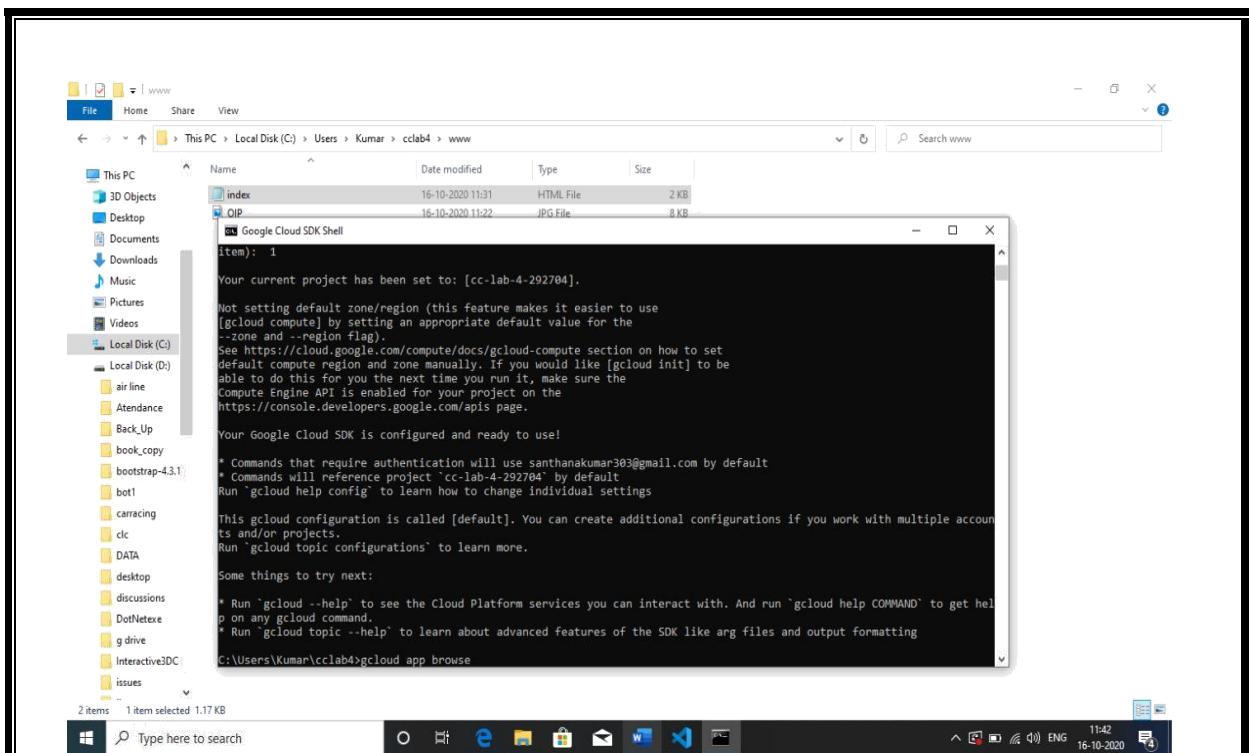
Screenshot 2: Visual Studio Code - index.html

```

button:hover, a:hover {
  opacity: 0.7;
}
</style>
</head>
<body>
<h2 style="text-align:center">ONLINE CLASS</h2>
<div class="card">
  
  <h3>CC CLASS</h3>
  <p>Time : 09-10 AM</p>
  <p>Join in TEAM</p>
  <div style="margin: 24px 0;">
    <a href="#"><i class="fa fa-dribbble"></i></a>
    <a href="#"><i class="fa fa-twitter"></i></a>
    <a href="#"><i class="fa fa-linkedin"></i></a>
    <a href="#"><i class="fa fa-facebook"></i></a>
  </div>
  <p><button>Contact</button></p>
</div>

```







CONCLUSION:

Thus the web application was launched successfully using the GAE launcher.

Ex No:5	SIMULATE A CLOUD SCENARIO USING CLOUD SIM AND RUN A SCHEDULING ALGORITHM THAT IS NOT PRESENT IN CLOUD SIM
Date:	

AIM:

To simulate a cloud scenario using cloud sim and run a scheduling algorithm that is not present in cloud sim

PROCEDURE:

STEP 1: Download the cloud sim installable files from

<http://code.google.com/p/cloudsim/downloads/list> and unzip

STEP 2: Open the IDE's such as Eclipse or NetBeans in your system

STEP 3: Create a new java project

File -> New

STEP 4: Import an unpacked Cloud Sim project into the new Java project

. **STEP 5:** The first step is to initialize the cloud sim package by initializing the cloud Sim Library ,as cloud Sim.init (num-user , calender , trace-flag)

STEP 6: Creation of data centers is the second step. To create datacenter, you need the Data center characteristics object that stores the properties of data center such as OS , list of machines,time zone etc

STEP 7: The third step is to create a broker.

 DataCenter Broker broker =CreateBroker();

STEP 8: The fourth step is to create one virtual machine Unique mips, ID of the VM , UserID ID the VM's owner,mips,amount of storage etc

 Vm vm = new Vm (VMid ,brokerID ,mips.ped Number,ram,bw,size,vmm,new CloudScheduler Time Shared())

STEP 9: Submit the VM list of the broker

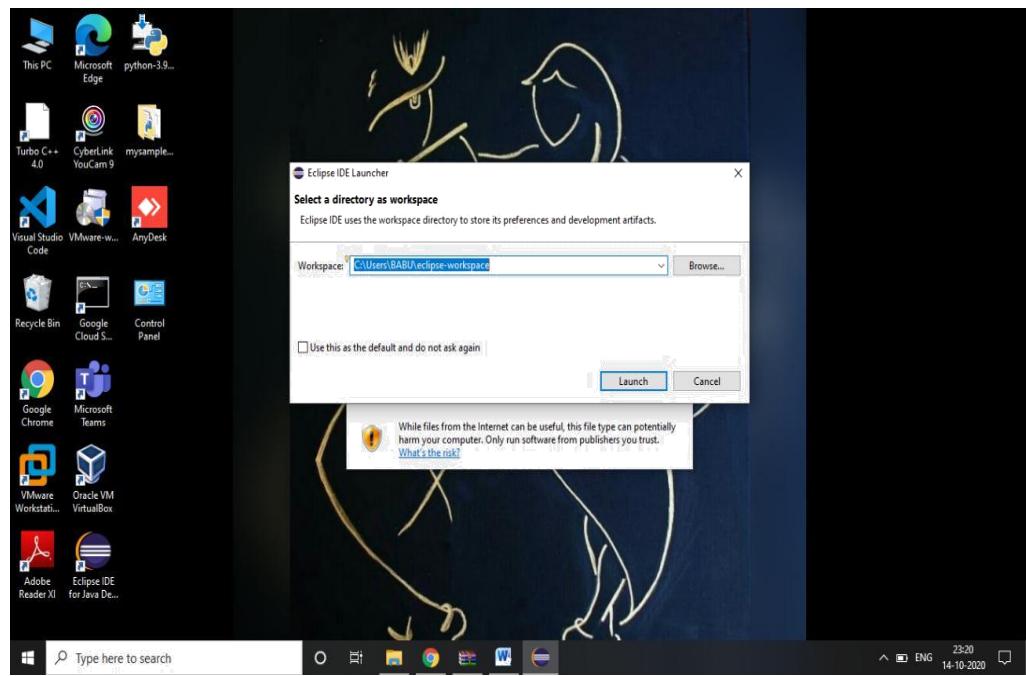
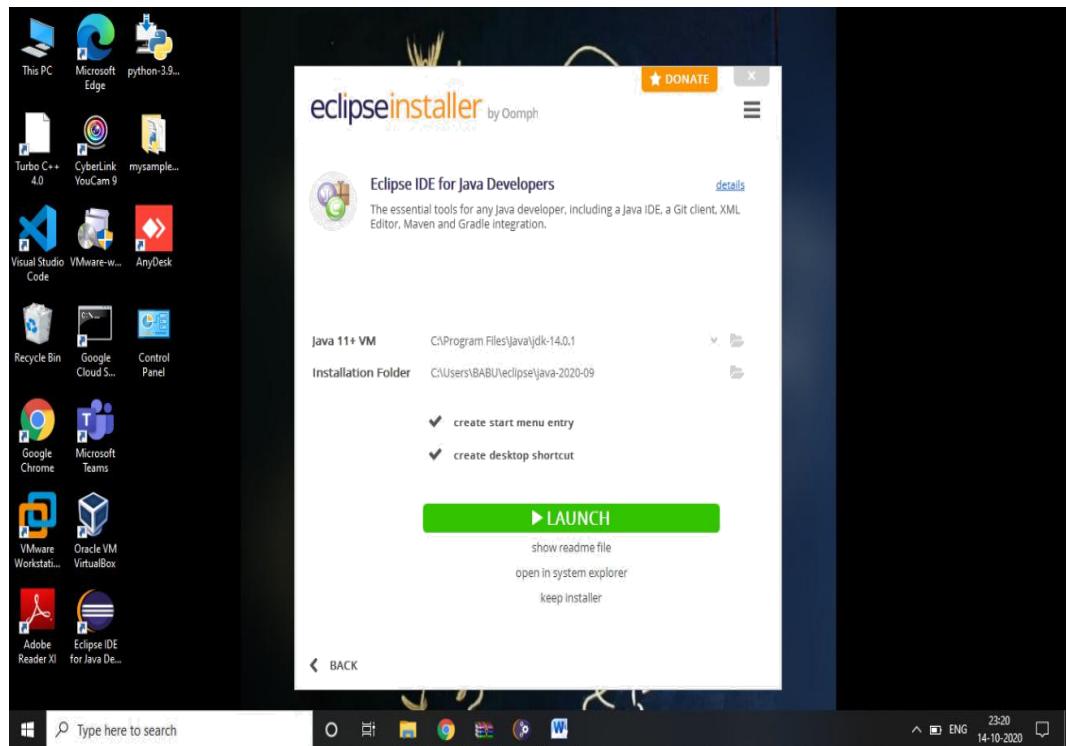
 Broker.submit VmList (VmList)

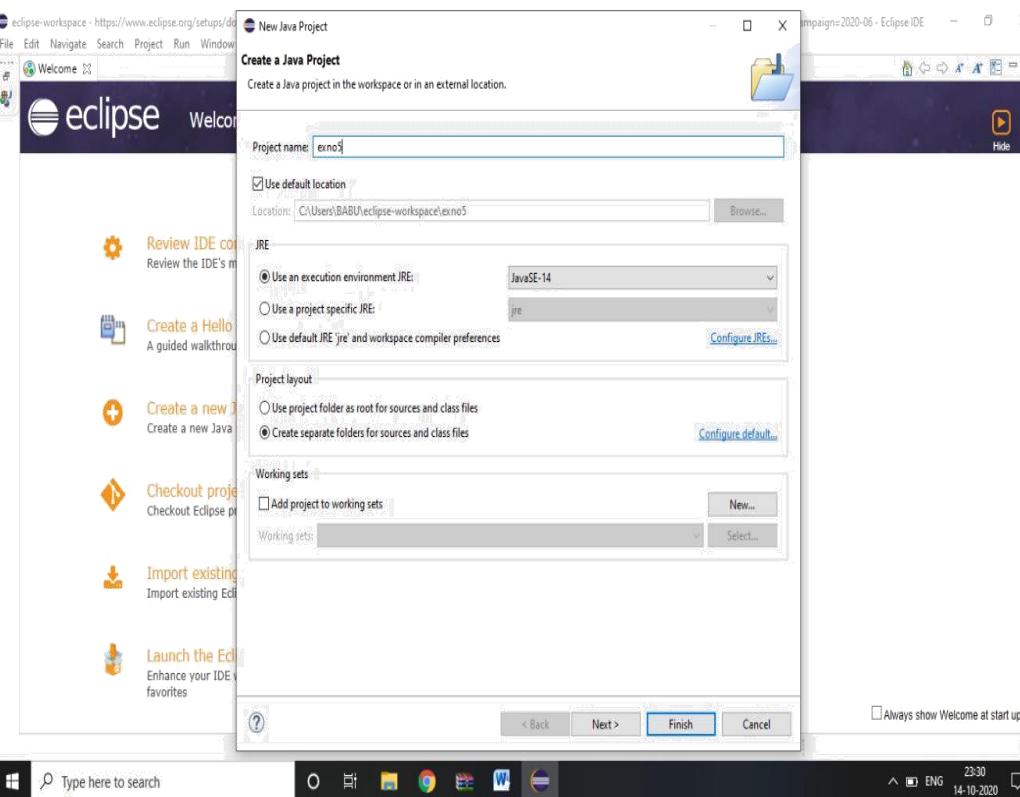
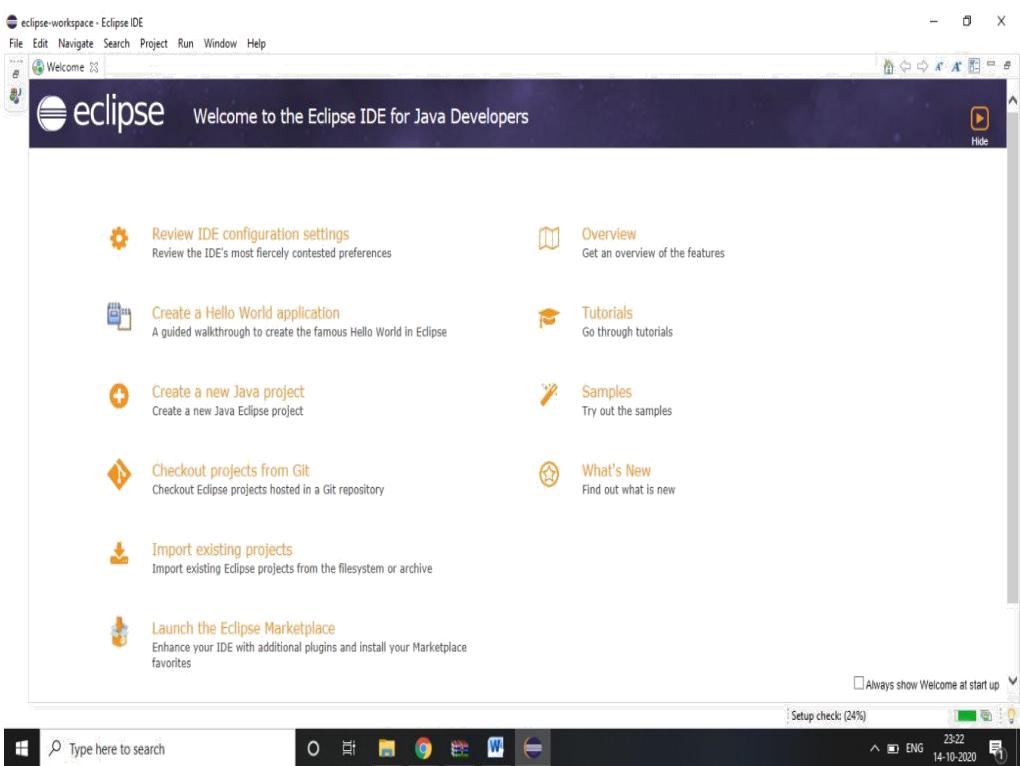
STEP 10: Create a cloud let with length , file size , output and utilization model.
Cloudlet.Cloudlet =new
cloudlet(id,length,pesNumber,filesize,outputsize,utilization,Model,Utilization mode)

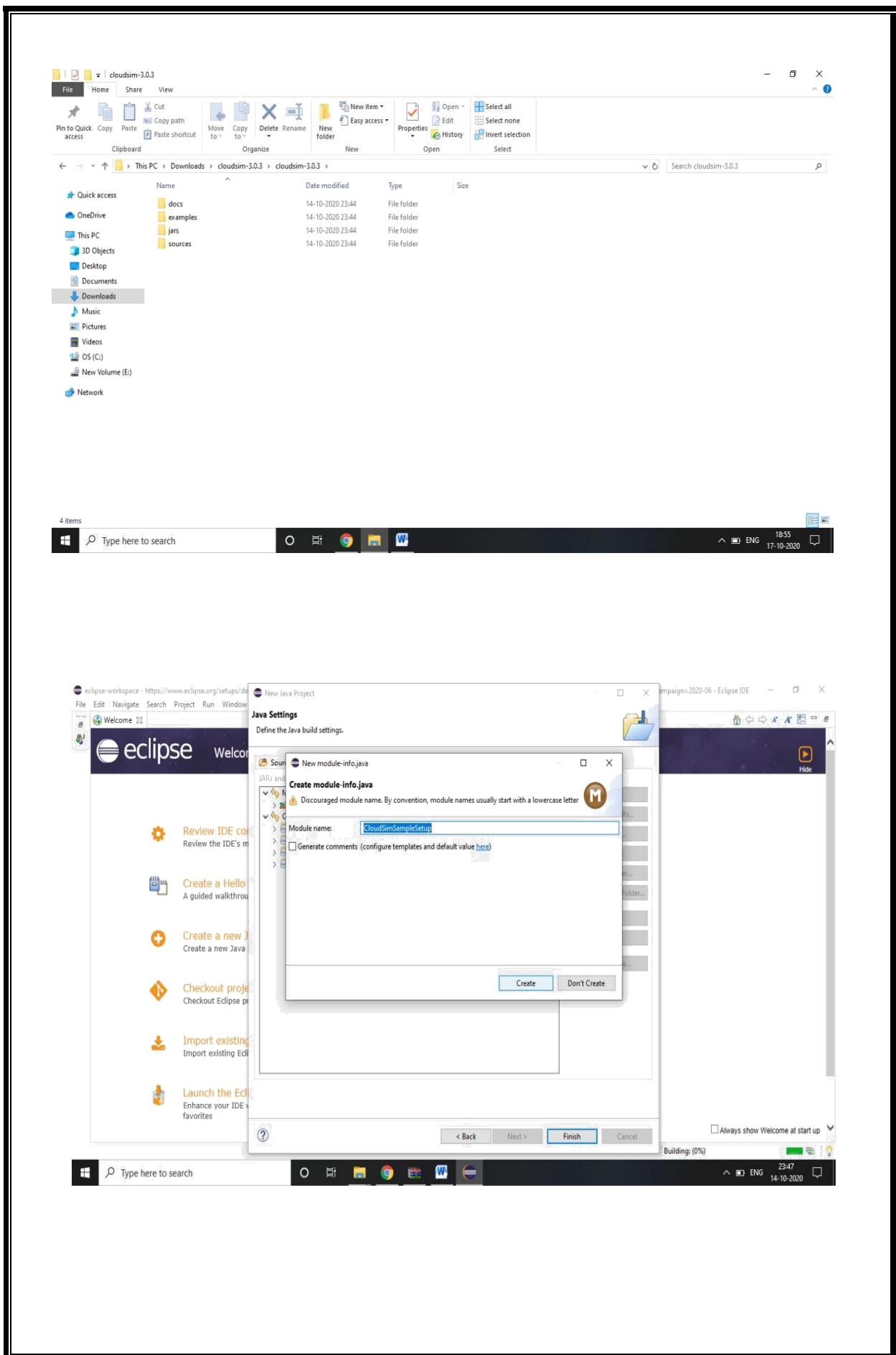
STEP 11: Submit the cloud let list to the broker broker.submit Cloudlet list(Cloudlet List)

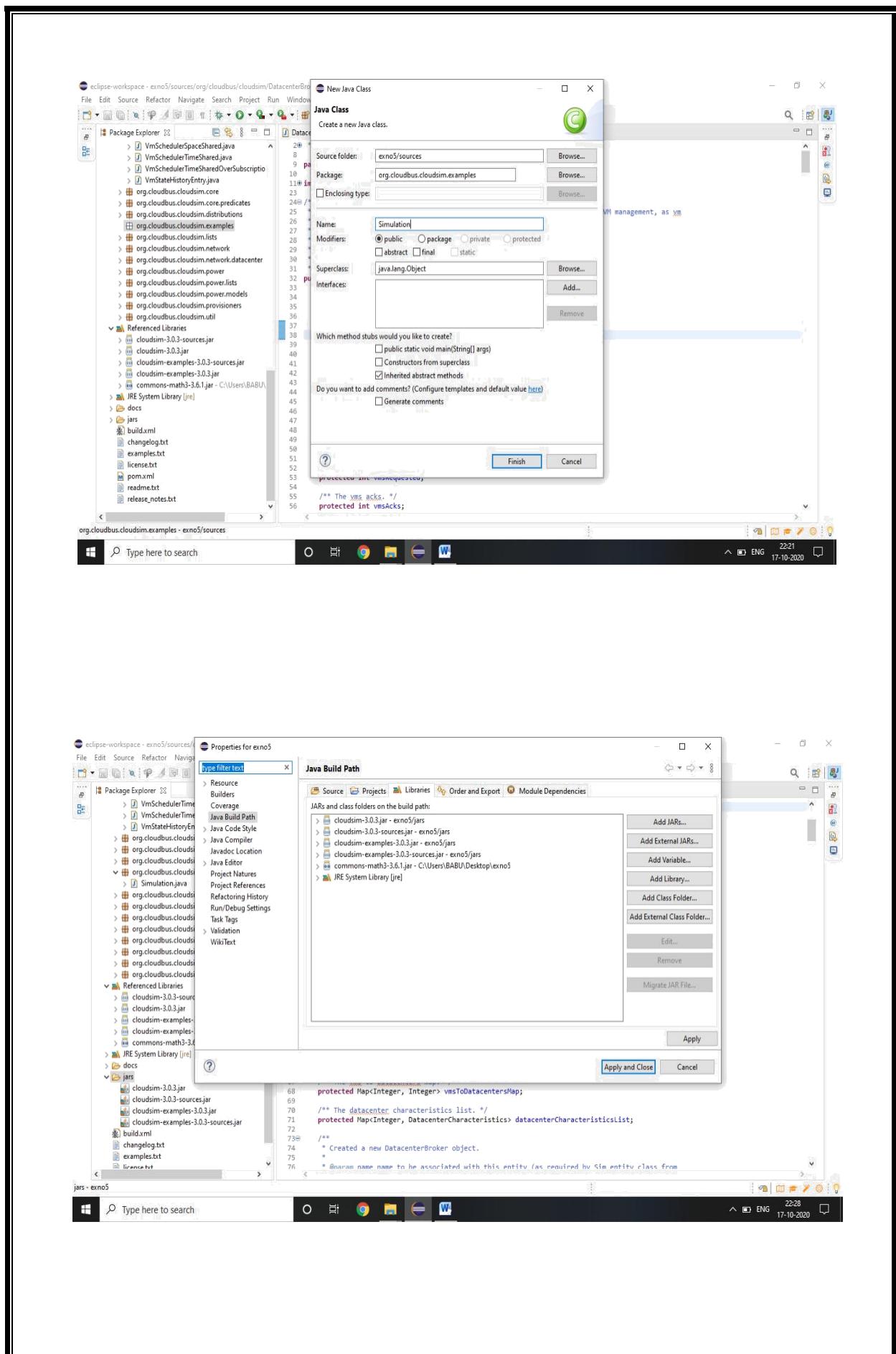
STEP 12: Start the simulation

Cloudsim , startSimulation()









Eclipse IDE - DatacenterBroker.java

```

eclipse-workspace - exno5/sources/org/cloudbus/cloudsim/DatacenterBroker.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Simulation.java DatacenterBroker.java
examples
  examples (default package)
  org.cloudbus.cloudsim.examples
    CloudSimExample1.java
    CloudSimExample2.java
    CloudSimExample3.java
    CloudSimExample4.java
    CloudSimExample5.java
    CloudSimExample6.java
    CloudSimExample7.java
    CloudSimExample8.java
org.cloudbus.cloudsim.examples.network
org.cloudbus.cloudsim.examples.network.d
org.cloudbus.cloudsim.examples.power
org.cloudbus.cloudsim.examples.power.pla
org.cloudbus.cloudsim.examples.power.ran
workload.planetlab
sources
  org.cloudbus.cloudsim
    Cloudlet.java
    CloudletScheduler.java
    CloudletScheduledDynamicWorkload.java
    CloudletSchedulerSpaceShared.java
    CloudletSchedulerTimeShared.java
    Consts.java
    Datacenter.java
    DatacenterBroker.java
    DatacenterCharacteristics.java
    DataCloudTags.java
    File.java
    FileAttribute.java
    HarddriveStorage.java
org.cloudbus.cloudsim.DatacenterBroker.java - exno5/sources

```

Eclipse IDE - CloudSimExample.java

```

eclipse-workspace - exno5/examples/org/cloudbus/cloudsim/examples/CloudSimExample.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Problems Declaration Console
terminated: CloudSimExample [Java Application] C:\Users\BABAU\AppData\Local\Temp\eo48D1\plugins\org.eclipse.jdt.core\openjdk.hotspot.jre.minimal.stripped.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (15-Oct-2020, 10:50:47 p
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Initialising...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: Sending VM #0 to VM #0
400.0: Broker: Cloudlet 0 received
400.1: Broker: All Cloudlets executed. Finishing...
400.1: Broker: Destroying VM #0
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

***** OUTPUT *****
Cloudlet ID STATUS Data center ID VM ID Time Start Time Finish Time
  0  SUCCESS      2        0     400     0.1    400.1
CloudSimExample1 finished!

```

eclipse-workspace - exno5/examples/org/cloudbus/cloudsim/examples/CloudSimExample6.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

<terminated> CloudSimExample6 [Java Application] C:\Users\BABA\AppData\Local\Temp\ec40D1.tmp\plugins\org.eclipse.jst.jdt.openjdk.hotspot.jre.minimal.stripped.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (17-Oct-2020, 10:11:26)

***** OUTPUT *****

Cloudlet ID	Status	Data center ID	VM ID	Time	Start Time	Finish Time
4	SUCCESS	2	4	3	0.2	3.2
16	SUCCESS	2	4	3	0.2	3.2
28	SUCCESS	2	4	3	0.2	3.2
5	SUCCESS	2	5	3	0.2	3.2
17	SUCCESS	2	5	3	0.2	3.2
29	SUCCESS	2	5	3	0.2	3.2
6	SUCCESS	3	6	3	0.2	3.2
18	SUCCESS	3	6	3	0.2	3.2
30	SUCCESS	3	6	3	0.2	3.2
7	SUCCESS	3	7	3	0.2	3.2
19	SUCCESS	3	7	3	0.2	3.2
31	SUCCESS	3	7	3	0.2	3.2
8	SUCCESS	3	8	3	0.2	3.2
20	SUCCESS	3	8	3	0.2	3.2
32	SUCCESS	3	8	3	0.2	3.2
10	SUCCESS	3	10	3	0.2	3.2
22	SUCCESS	3	10	3	0.2	3.2
34	SUCCESS	3	10	3	0.2	3.2
9	SUCCESS	3	9	3	0.2	3.2
21	SUCCESS	3	9	3	0.2	3.2
33	SUCCESS	3	9	3	0.2	3.2
11	SUCCESS	3	11	3	0.2	3.2
23	SUCCESS	3	11	3	0.2	3.2
35	SUCCESS	3	11	3	0.2	3.2
0	SUCCESS	2	0	4	0.2	4.2
12	SUCCESS	2	0	4	0.2	4.2
24	SUCCESS	2	0	4	0.2	4.2
36	SUCCESS	2	0	4	0.2	4.2
1	SUCCESS	2	1	4	0.2	4.2
13	SUCCESS	2	1	4	0.2	4.2
25	SUCCESS	2	1	4	0.2	4.2
37	SUCCESS	2	1	4	0.2	4.2
2	SUCCESS	2	2	4	0.2	4.2
14	SUCCESS	2	2	4	0.2	4.2
26	SUCCESS	2	2	4	0.2	4.2
38	SUCCESS	2	2	4	0.2	4.2

22:16 ENG 17-10-2020

eclipse-workspace - exno5/examples/org/cloudbus/cloudsim/examples/CloudSimExample6.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

<terminated> CloudSimExample6 [Java Application] C:\Users\BABA\AppData\Local\Temp\ec40D1.tmp\plugins\org.eclipse.jst.jdt.openjdk.hotspot.jre.minimal.stripped.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (17-Oct-2020, 10:11:26)

***** OUTPUT *****

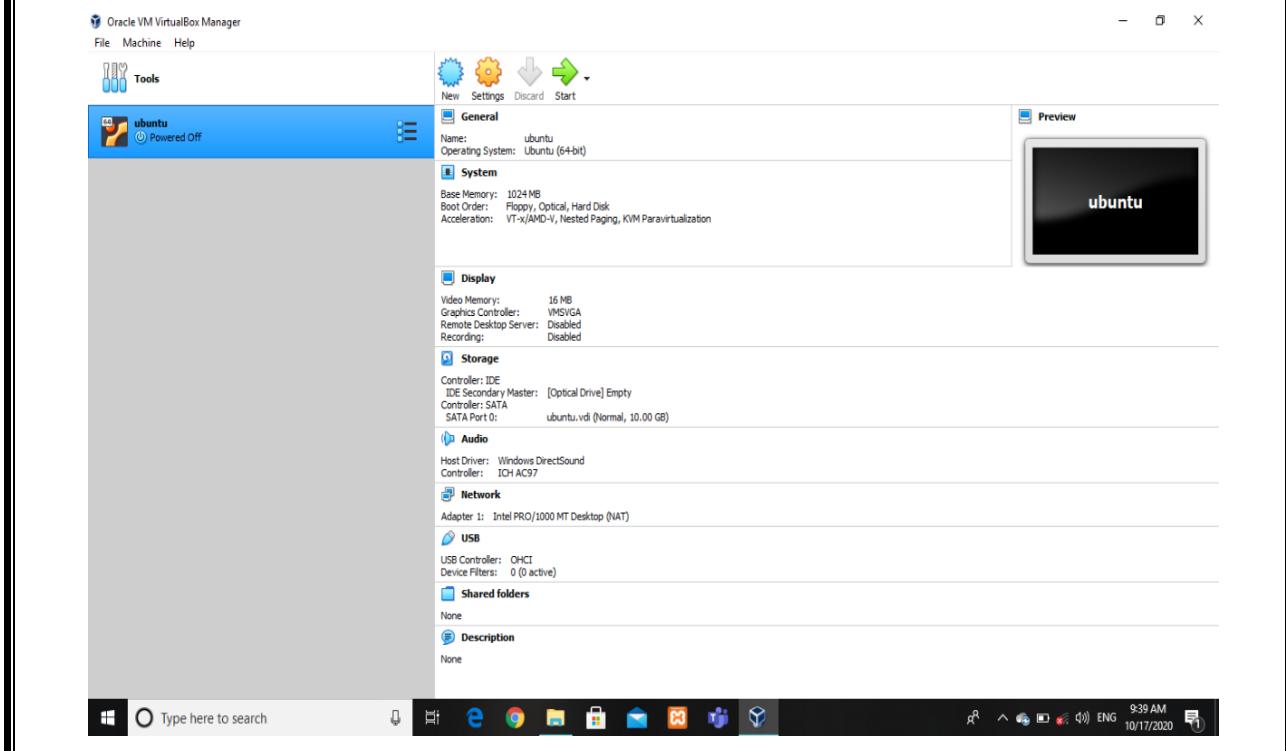
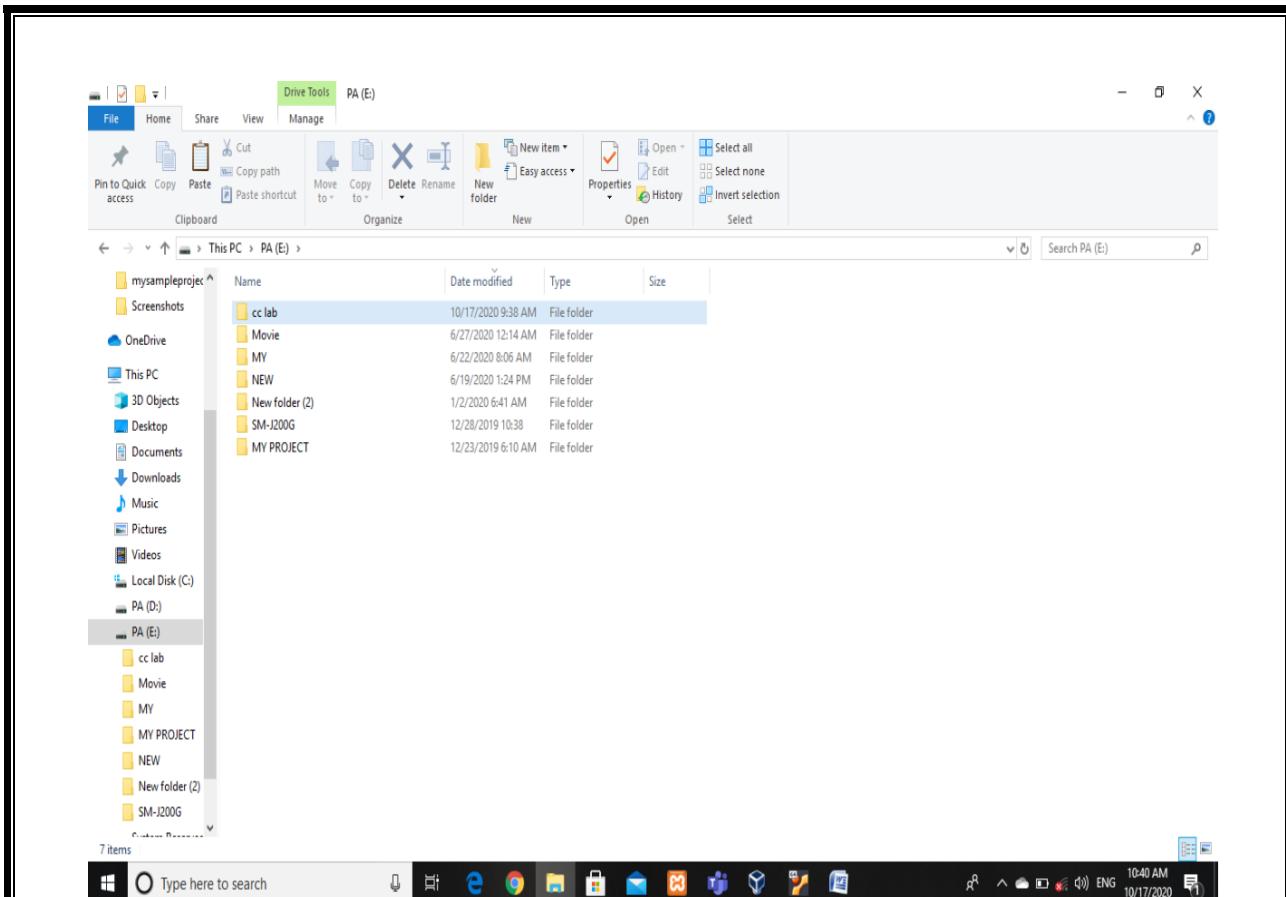
Cloudlet ID	Status	Data center ID	VM ID	Time	Start Time	Finish Time
17	SUCCESS	2	5	3	0.2	3.2
29	SUCCESS	2	5	3	0.2	3.2
5	SUCCESS	3	6	3	0.2	3.2
18	SUCCESS	3	6	3	0.2	3.2
30	SUCCESS	3	6	3	0.2	3.2
7	SUCCESS	3	7	3	0.2	3.2
19	SUCCESS	3	7	3	0.2	3.2
31	SUCCESS	3	7	3	0.2	3.2
8	SUCCESS	3	8	3	0.2	3.2
20	SUCCESS	3	8	3	0.2	3.2
32	SUCCESS	3	8	3	0.2	3.2
10	SUCCESS	3	10	3	0.2	3.2
22	SUCCESS	3	10	3	0.2	3.2
34	SUCCESS	3	10	3	0.2	3.2
9	SUCCESS	3	9	3	0.2	3.2
21	SUCCESS	3	9	3	0.2	3.2
33	SUCCESS	3	9	3	0.2	3.2
11	SUCCESS	3	11	3	0.2	3.2
23	SUCCESS	3	11	3	0.2	3.2
35	SUCCESS	3	11	3	0.2	3.2
0	SUCCESS	2	0	4	0.2	4.2
12	SUCCESS	2	0	4	0.2	4.2
24	SUCCESS	2	0	4	0.2	4.2
36	SUCCESS	2	0	4	0.2	4.2
1	SUCCESS	2	1	4	0.2	4.2
13	SUCCESS	2	1	4	0.2	4.2
25	SUCCESS	2	1	4	0.2	4.2
37	SUCCESS	2	1	4	0.2	4.2
2	SUCCESS	2	2	4	0.2	4.2
14	SUCCESS	2	2	4	0.2	4.2
26	SUCCESS	2	2	4	0.2	4.2
38	SUCCESS	2	2	4	0.2	4.2
3	SUCCESS	2	3	4	0.2	4.2
15	SUCCESS	2	3	4	0.2	4.2
27	SUCCESS	2	3	4	0.2	4.2
39	SUCCESS	2	3	4	0.2	4.2

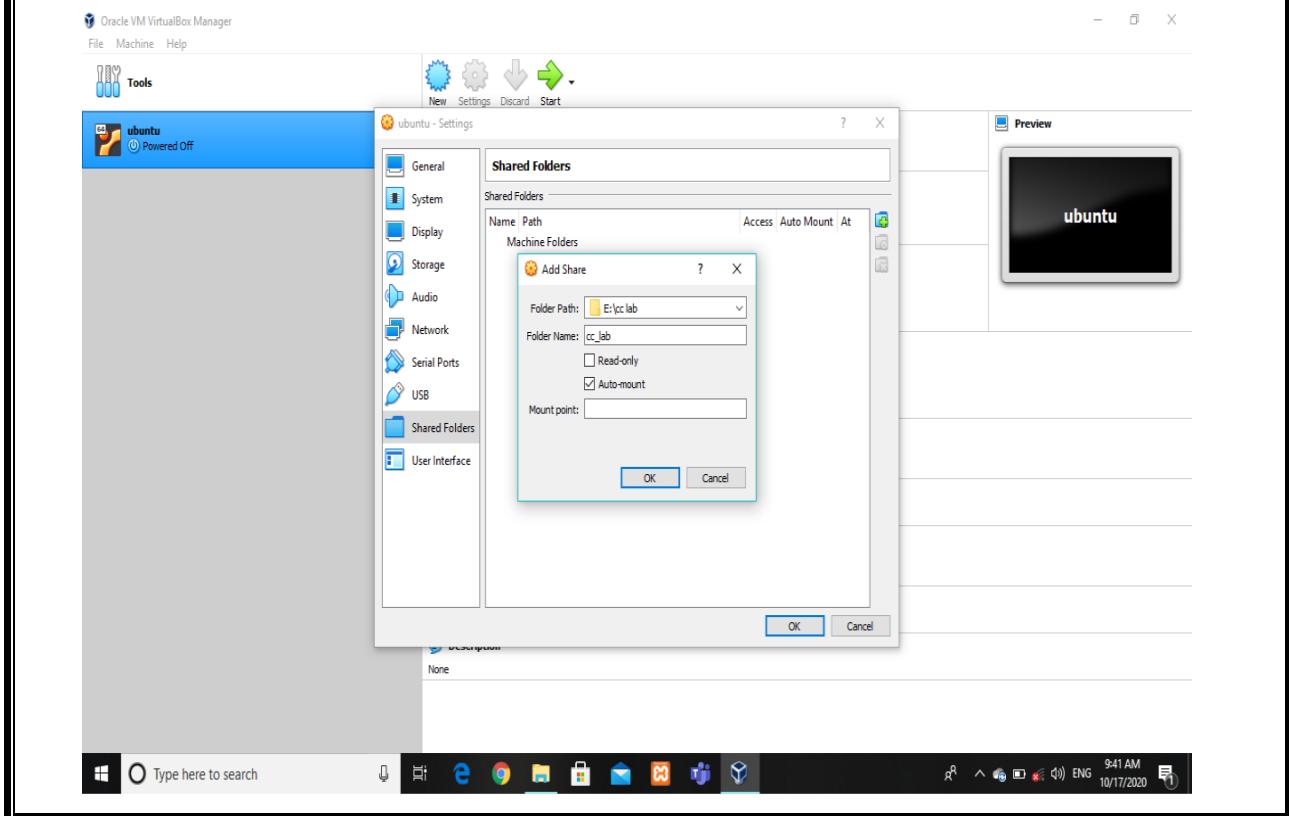
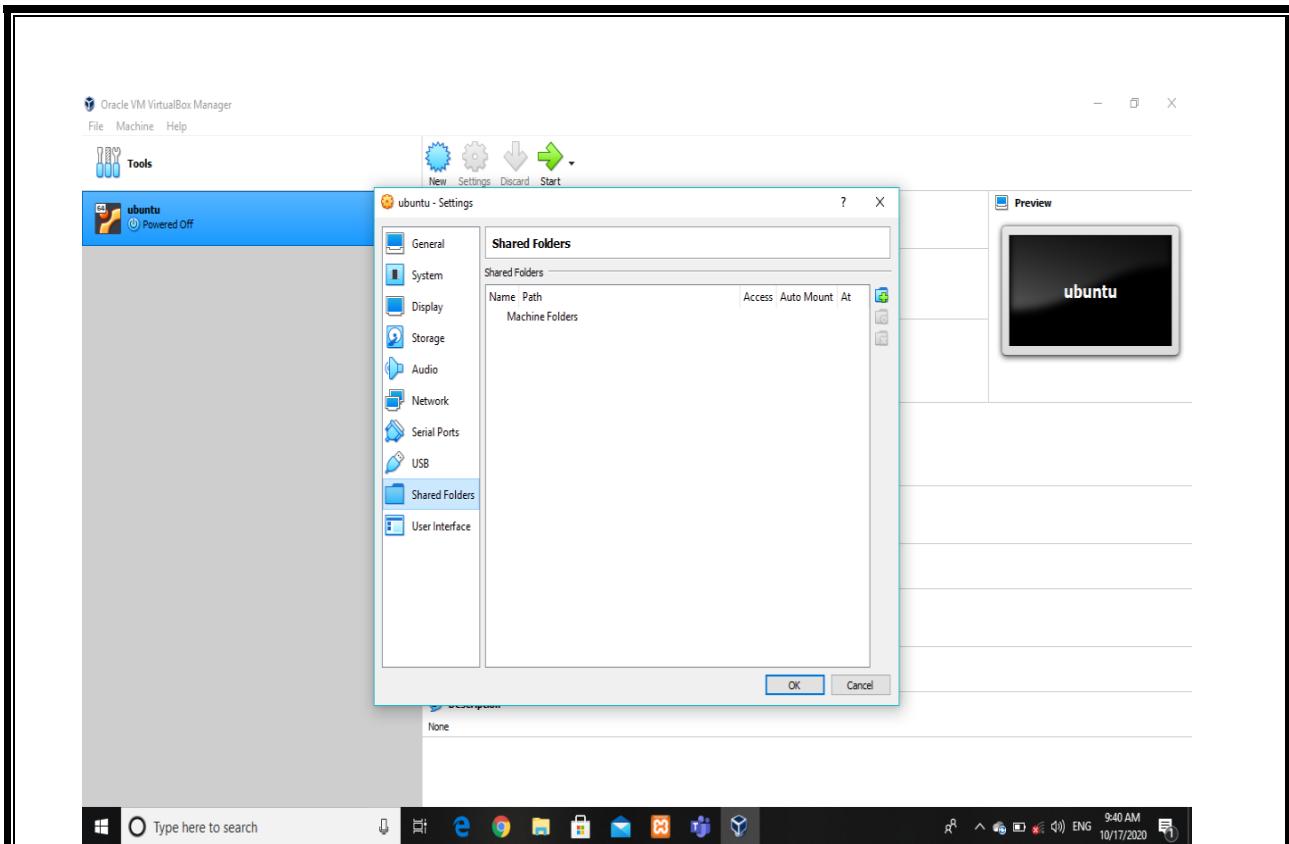
22:16 ENG 17-10-2020

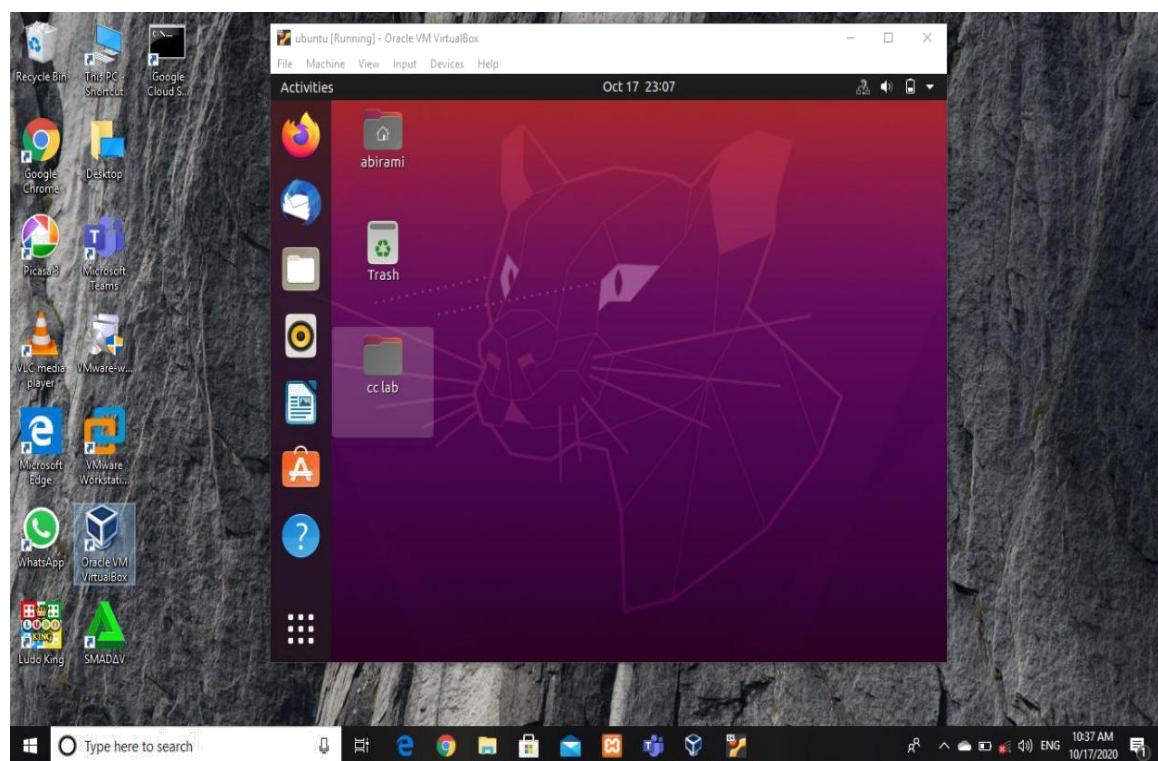
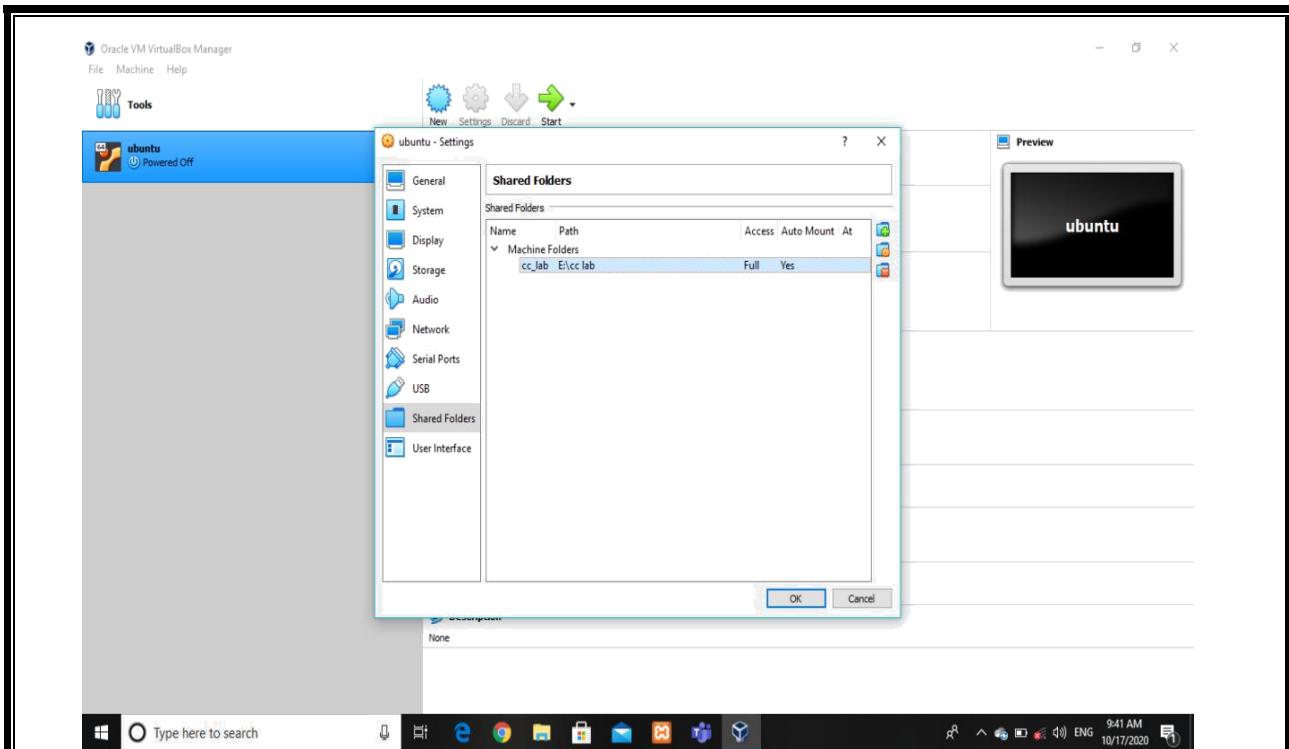
CONCLUSION:

Thus a cloud scenario using Cloud Sim was simulated and a scheduling algorithm that is not present in cloud Sim was executed and the output is verified successfully.

Ex no: 6	FIND A PROCEDURE TO TRANSFER THE FILES FROM ONE VIRTUAL MACHINE TO ANOTHER VIRTUAL MACHINE
Date:	
Aim:	
To find a procedure to transfer the files from one virtual machine to another virtual machine.	
Procedure:	
<p>STEP 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.</p> <p>STEP 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)</p> <p>STEP 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.</p> <p>STEP 4: You can use usual method to copy file between two different computer with client server application (eg: scp with sshd active for linux)</p> <p>STEP 5: You can mount part of the file system on a virtual machine via NFS or 3S HFS (or) you can also share file and directory with samba.</p>	







CONCLUSION:

Thus the procedure to transfer the files from one virtual machine to another virtual machine was explained and the output is verified successfully.

Ex No:7	INSTALL HADOOP SINGLE NODE CLUSTER AND RUN APPLICATIONS LIKE WORDCOUNT
Date:	

Aim:

To install Hadoop single node cluster and run simple applications like word count.

Procedure:

STEP 1: Download the Java & package from <https://drive.google.com/file/d/0BwIBPXSoIx-FIKRN21/mu9ZMm8/> view and save this file in your home directory.

STEP 2: Extract the Java Jar File using the command, tar-xuf jdk-swar-linux. I586. tar.gz

STEP 3: Download the Hadopp 3.7.3 package using the command wget <https://archive.apache.org/disk1/hadoop/core/hadoop-2.7.3/hadoop-2.7.3.tar.gz>

STEP 4: Extract the Hadopp tur file using command bar. xvf hadoop-2.7.3.tar.gz.

STEP 5: Add the Hadoop and Java path in the bash file, save the bash tile and close it.

STEP 6: Edit the Hadoop Configuration files using the command cd hadoop-2.7.3 / etc / Hadoop

STEP 7: Open the core site.xml and edit the property mentioned below using Vi core site.xml

STEP 8: Edit the half site.xml and edit the property using Vi hdfs.site.xml

STEP 9: Edit the mapred site.xml file and edit the property using cp mapred-site.xml template mapred-site.xml and then use the command vi mapred-site. xml

STEP 10: Edit yarn site.xml and edit the property using vi yarn- site. xml.

STEP 11: Edit hadoop.env.sh and add the Java Path using vi hadoop- env. sh

STEP 12: Go to hadoop home directory and format the NameNode

STEP 13 : Once the NameNode is formatted, go to hadoop-2.7.3 /sbin directory and start all the demons using /startallsh

STEP 14: Once the hadoop single node cluster is installed, we can now run a simple application like word count in it.

STEP 15: Create a new Java project and add JAR files to it.

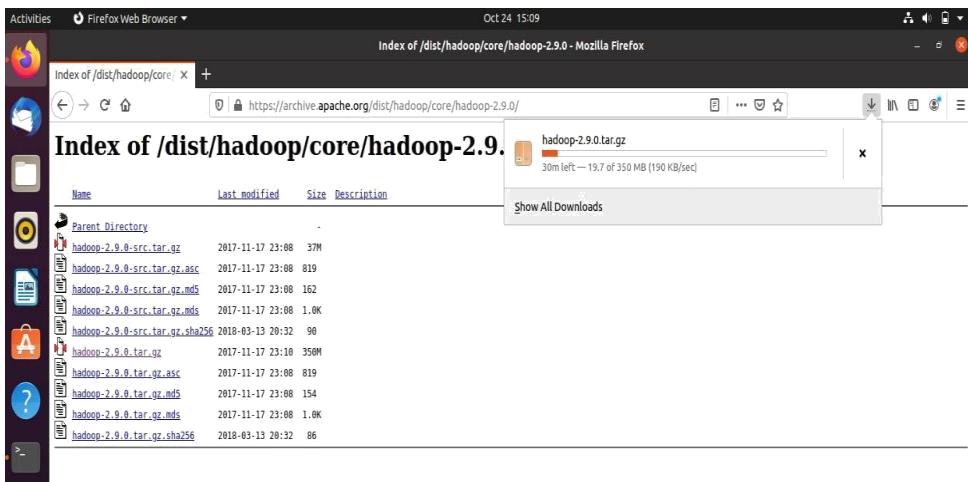
STEP 16: Create a new package within the project with the name com. code, dezyre.

STEP 17: Now implement the word count example program and create a Mapper class within the word count class which extends Map Reduce Base class to implement mapper interface.

STEP 18: Finally, execute the Hadoop Map Reduce word count program.

```
boomi@boomi-VirtualBox:~$ sudo apt-get install openjdk-8-jdk
[sudo] password for boomi:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev libx11-6
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc
  openjdk-8-demo openjdk-8-source visualvm icedtea-8-plugin
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
  fonts-wqy-zenhei
The following NEW packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
The following packages will be upgraded:
  libx11-6
```

```
boomi@boomi-VirtualBox:~$ source .bashrc
boomi@boomi-VirtualBox:~$ java -version
openjdk version "1.8.0_265"
OpenJDK Runtime Environment (build 1.8.0_265-8u265-b01-0ubuntu2-20.04-b01)
OpenJDK 64-Bit Server VM (build 25.265-b01, mixed mode)
boomi@boomi-VirtualBox:~$
```



```
[sudo] password for boomi:  
Adding group `hadoop` (GID 1001) ...  
Done.  
boomi@boomi-VirtualBox:~$ sudo adduser --ingroup hadoop hadoopusr  
Adding user `hadoopusr' ...  
Adding new user `hadoopusr' (1001) with group `hadoop' ...  
Creating home directory `/home/hadoopusr' ...  
Copying files from `/etc/skel' ...  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for hadoopusr  
Enter the new value, or press ENTER for the default  
    Full Name []: Boominathan  
    Room Number []: 25  
    Work Phone []: no  
    Home Phone []: Yes  
    Other []:  
Is the information correct? [Y/n] Y  
boomi@boomi-VirtualBox:~$ sudo adduser hadoopusr sudo  
Adding user `hadoopusr' to group `sudo' ...  
Adding user hadoopusr to group sudo  
Done.  
boomi@boomi-VirtualBox:~$
```

```
hadoopusr@boomi-VirtualBox:~$ ssh-keygen -t rsa -P ""  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/hadoopusr/.ssh/id_rsa):  
Created directory '/home/hadoopusr/.ssh'.  
Your identification has been saved in /home/hadoopusr/.ssh/id_rsa  
Your public key has been saved in /home/hadoopusr/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:jqCB0Pi7L/GK0K7bEBfb7p856g1FzzBN1f20B2aguw hadoopusr@boomi-VirtualBox  
The key's randomart image is:  
+---[RSA 3072]---+  
|          +. o.|  
|  o . . . o 0+ .|  
|o o  o oo.. o .|  
|.o + . 0.0 . |  
|o = o + SE |  
| +.= o = |  
|o +o= o . |  
|.*.+0+. |  
|+oXBoo. |  
+---[SHA256]---+  
hadoopusr@boomi-VirtualBox:~$
```

```
hadoopusr@boomi-VirtualBox:/home/boomi/Desktop$ ls  
hadoop-2.9.0  hadoop-2.9.0.tar.gz  
hadoopusr@boomi-VirtualBox:/home/boomi/Desktop$ sudo mv hadoop-2.9.0/usr/local/hadoop  
mv: missing destination file operand after 'hadoop-2.9.0/usr/local/hadoop'  
Try 'mv --help' for more information.  
hadoopusr@boomi-VirtualBox:/home/boomi/Desktop$ sudo mv hadoop-2.9.0 /usr/local/hadoop  
hadoopusr@boomi-VirtualBox:/home/boomi/Desktop$ sudo chown -R hadoopusr /usr/local/hadoop
```

Activities Terminal Oct 26 15:50

```
hadoopusr@boomi-VirtualBox: ~
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
hadoopusr@localhost's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

201 updates can be installed immediately.
85 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

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

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

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

hadoopusr@boomi-VirtualBox:~$ exit
logout
Connection to localhost closed.
hadoopusr@boomi-VirtualBox:~$
```

```
hdfs-site.xml (/usr/local/Cellar/hadoop/3.3.0/libexec/etc/hadoop) - VIM
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xmlstylesheet type="text/xsl" href="configuration.xsl"?>
3 <!--
4 Licensed under the Apache License, Version 2.0 (the "License");
5 you may not use this file except in compliance with the License.
6 You may obtain a copy of the License at
7
8 http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License. See accompanying LICENSE file.
15 -->
16
17 <!-- Put site-specific property overrides in this file. -->
18
19 <configuration>
20   <property>
21     <name>dfs.replication</name>
22     <value>4</value>
23   </property>
24 </configuration>
-
-
-
-
-
-
-
-
-
```

```
core-site.xml (/usr/local/Cellar/hadoop/3.3.0/libexec/etc/hadoop) - VIM

1 <?xml version="1.0" encoding="UTF-8"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3 <!--
4   Licensed under the Apache License, Version 2.0 (the "License");
5   you may not use this file except in compliance with the License.
6   You may obtain a copy of the License at
7
8     http://www.apache.org/licenses/LICENSE-2.0
9
10    Unless required by applicable law or agreed to in writing, software
11    distributed under the License is distributed on an "AS IS" BASIS,
12    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13    See the License for the specific language governing permissions and
14    limitations under the License. See accompanying LICENSE file.
15
16  -->
17 <!-- Put site-specific property overrides in this file. -->
18
19 <configuration>
20   <property>
21     <name>hadoop.tmp.dir</name>
22     <value>/usr/local/Cellar/hadoop/hdfs/tmp</value>
23     <description>A base for other temporary directories.</description>
24   </property>
25   <property>
26     <name>fs.default.name</name>
27     <value>hdfs://localhost:9000</value>
28   </property>
29 </configuration>
-
```

```
1 <?xml version="1.0"?>
2 <xsl:stylesheet type="text/xsl" href="configuration.xsl"?>
3 <!--
4 Licensed under the Apache License, Version 2.0 (the "License");
5 you may not use this file except in compliance with the License.
6 You may obtain a copy of the License at
7
8 http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License. See accompanying LICENSE file.
15 -->
16
17 <!-- Put site-specific property overrides in this file. -->
18
19 <configuration>
20   <property>
21     <name>mapred.job.tracker</name>
22     <value>localhost:9010</value>
23   </property>
24 </configuration>
```

Overview	
Started:	Sat Apr 30 12:26:12 PDT 2016
Version:	2.7.2, rb165c4fe8a74265c792ce23f546c64604acf0e41
Compiled:	2016-01-26T00:08Z by jenkins from (detached from b165c4f)
Cluster ID:	CID-c0a4b56e-26ad-4a3b-91cd-a25caa6b8945
Block Pool ID:	BP-1339268805-127.0.1.1-1462043709581

The image shows two windows side-by-side. The left window is a web-based Hadoop file browser titled 'Browse Directory' at the URL `localhost:50070/explorer.html#/WordCountTutorial/Input`. It has a green header bar with links for 'Hadoop', 'Overview', 'Datanodes', 'Snapshot', 'Startup Progress', and 'Utilities'. Below the header is a search bar and a table with columns: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. The right window is a desktop environment for an 'ubuntu 20.4 [Running] - Oracle VM VirtualBox' machine. The desktop interface includes a dock with various icons at the bottom.

Browse Directory

/WordCountTutorial/Input

Go!

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
------------	-------	-------	------	---------------	-------------	------------	------

ubuntu 20.4 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Files Oct 26 17:52

Recent Starred Home Desktop Documents Downloads Music Pictures Videos Trash Other Locations

Name	Size	Modified	Star
input	51 bytes	17:52	☆
wordcount.java	2.5 kB	17:50	☆

ubuntu 20.4 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Terminal ▾ Oct 26 17:38

```
boomi@boomi-VirtualBox:~/Desktop$ 
boomi@boomi-VirtualBox:~/Desktop$ javac -classpath ${HADOOP_CLASSPATH} -d '/home/sheeha/Desktop/WordCountTutorial/tutorial_classes' '/home/sheeha/Desktop/WordCountTutorial/WordCount.java'
boomi@boomi-VirtualBox:~/Desktop$ jar -cvf firstTutorial.jar -C tutorial_classes/ .
added manifest
adding: WordCount$TokenizerMapper.class(in = 1736) (out= 754)(deflated 56%)
adding: WordCount$IntSumReducer.class(in = 1739) (out= 739)(deflated 57%)
adding: WordCount.class(in = 1491) (out= 814)(deflated 45%)
boomi@boomi-VirtualBox:~/Desktop$ hadoop jar '/home/sheeha/Desktop/WordCountTutorial/firstTutorial.jar' WordCount /WordCountTutorial/Input /WordCountTutorial/Output
16/04/30 16:28:09 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

```
16/04/30 16:29:02 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Bahcesehir      1
Istanbul        2
Jerusalem       2
Mohammed        4
Omar            1
Palestine        4
Sheeha          2
Shiha           1
```

CONCLUSION:

Thus the program for find procedure to set up the Hadoop single node cluster and run simple application like word count was executed successfully and the output is verified

E.x.No: 8	
Date:	

Creating and Executing First Container using Docker

AIM:

To create and execute the first docker container.

PROCEDURE:

Step 1:

Clone the repository at <https://github.com/docker/welcome-to-docker>.

```
git clone https://github.com/docker/welcome-to-docker
```

The rest of this guide requires you to run commands in the new project directory.
Run the following command before moving on.

```
cd welcome-to-docker
```

Step 2:

Verify the docker file

Open the sample application in your IDE. Note that it already has a **Dockerfile**.
For your own projects you need to create this yourself.

Step 3:

Build Your Image

You can build an image using the following docker build command via a CLI in your project folder.

```
docker build -t welcome-to-docker .
```

Breaking down this command

The **-t** flag tags your image with a name. (welcome-to-docker in this case).
And the **.** lets Docker know where it can find the Docker file.

Step 4 :

Run your Container

Once the build is complete, an image will appear in the **Images** tab. Select the image name to see its details. Select **Run** to run it as a container. In the **Optional settings** remember to specify a port number (something like **8089**).

Images Give feedback						
Local		Hub	Artifactory	EARLY ACCESS		
347.61 MB / 347.59 MB in use 3 images						Last refresh: 3 hours ago
Search	Filter	Actions				
Name	Tag	Status	Created	Size	Actions	
welcome-to-docker 613906070843	latest	in use	3 days ago	269.77 MB		

Run a new container
`welcome-to-docker:latest`

Optional settings

Container name

A random name is generated if you do not provide one.

Ports

Enter "0" to assign randomly generated host ports.

Host port Container port

Volumes

Host path Container path

Environment variables

Variable Value

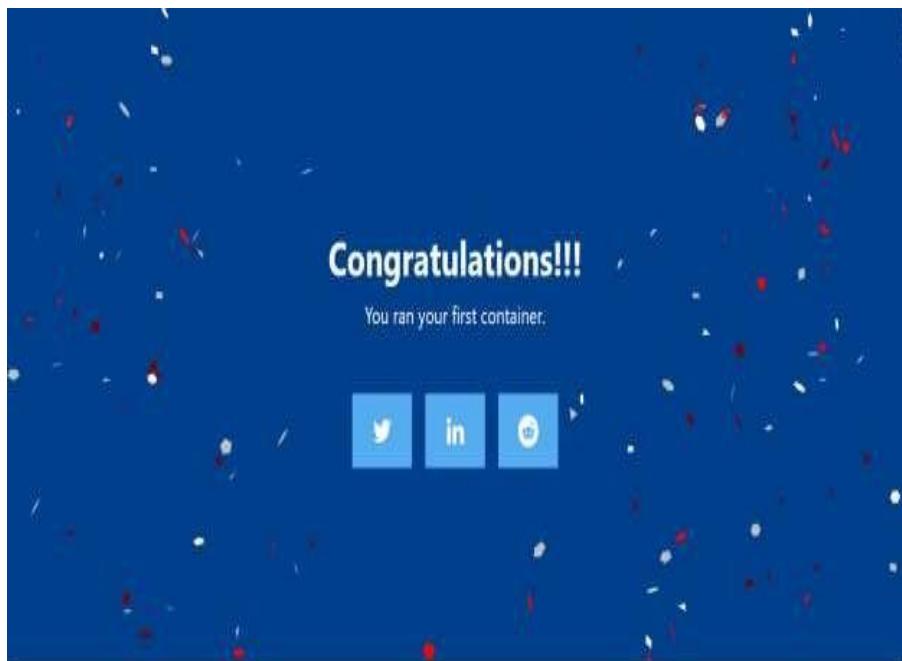
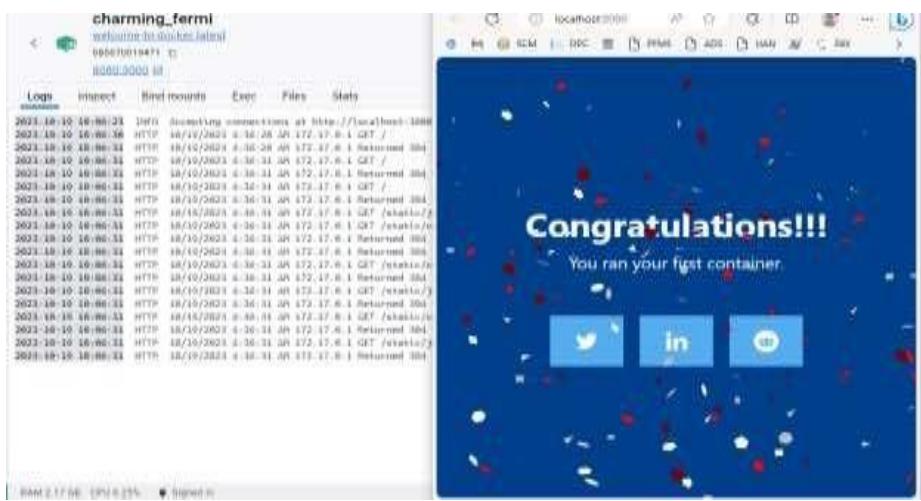
You can also run using

```
docker run welcome-to-docker.
```

Step 5:

View the output in the localhost:8080

OUTPUT / SCREENSHOTS:



CONCLUSION:

Thus, the first docker container has been created and executed successfully.

E.x. No: 9	
Date:	Run a Container from Docker Hub

AIM:

To run a container from docker hub.

PROCEDURE:**Step 1:**

Sign to the docker hub. Using,

```
docker login
```

```
C:\Users\Windows>docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub.
You can log in with your password or a Personal Access Token (PAT).
For more information, see https://docs.docker.com/go/access-tokens/
Username: harris.502126@sxccce.edu.in
Password:
Login Succeeded
```

Step 2:

Pull the docker image.

```
docker pull hello-world
```

Step 3:

Run your image.

Once the build is complete, an image will appear in the **Images** tab. Select the image name to see its details. Select **Run** to run it as a container.

Also, you can run an image using the following docker run command via a CLI in your project folder.

```
docker run hello-world
```

Step 4:

View the container. If the container worked correctly, it shows the following message

```
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.
```

OUTPUT / SCREENSHOTS:

```
C:\Users\Windows>docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

CONCLUSION:

Thus, the first docker image has been pulled from docker hub and ran a container successfully.