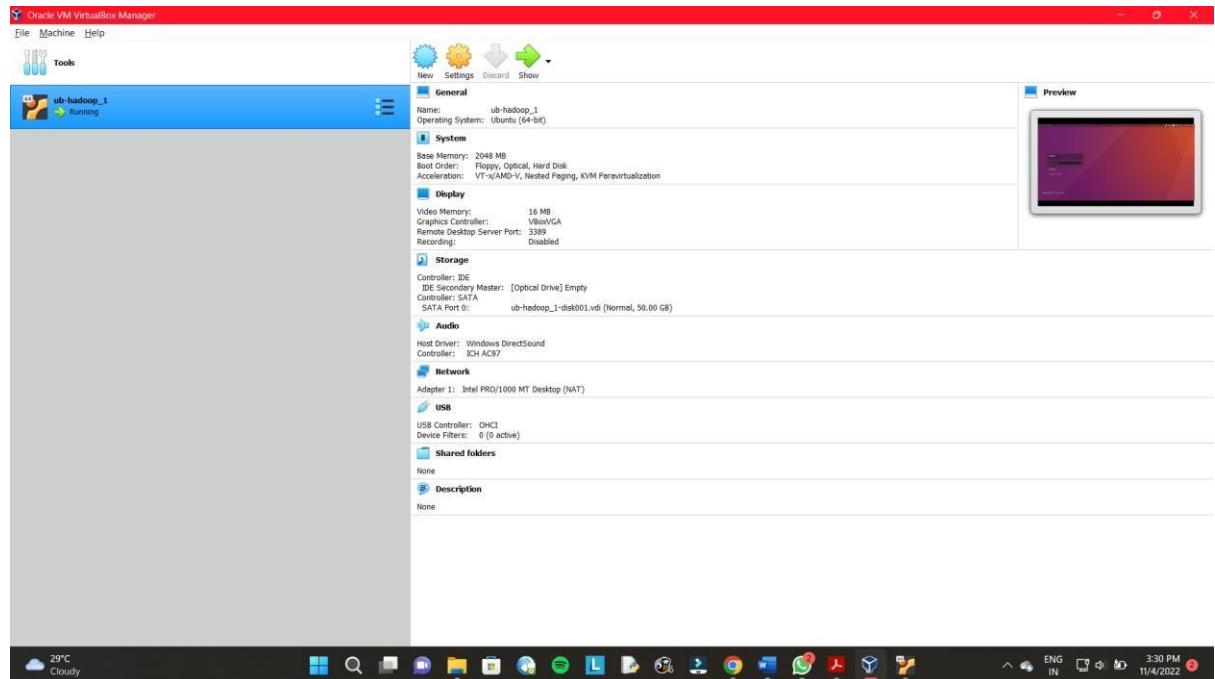


Ex. No. 01	Install Virtualbox/VMware Workstation with different flavours of linux
Date:	or windows OS on top of windows7 or 8.

Aim:

Procedure:

Output:



Result:

Ex. No. 02

Date:

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

Aim:

Procedure:

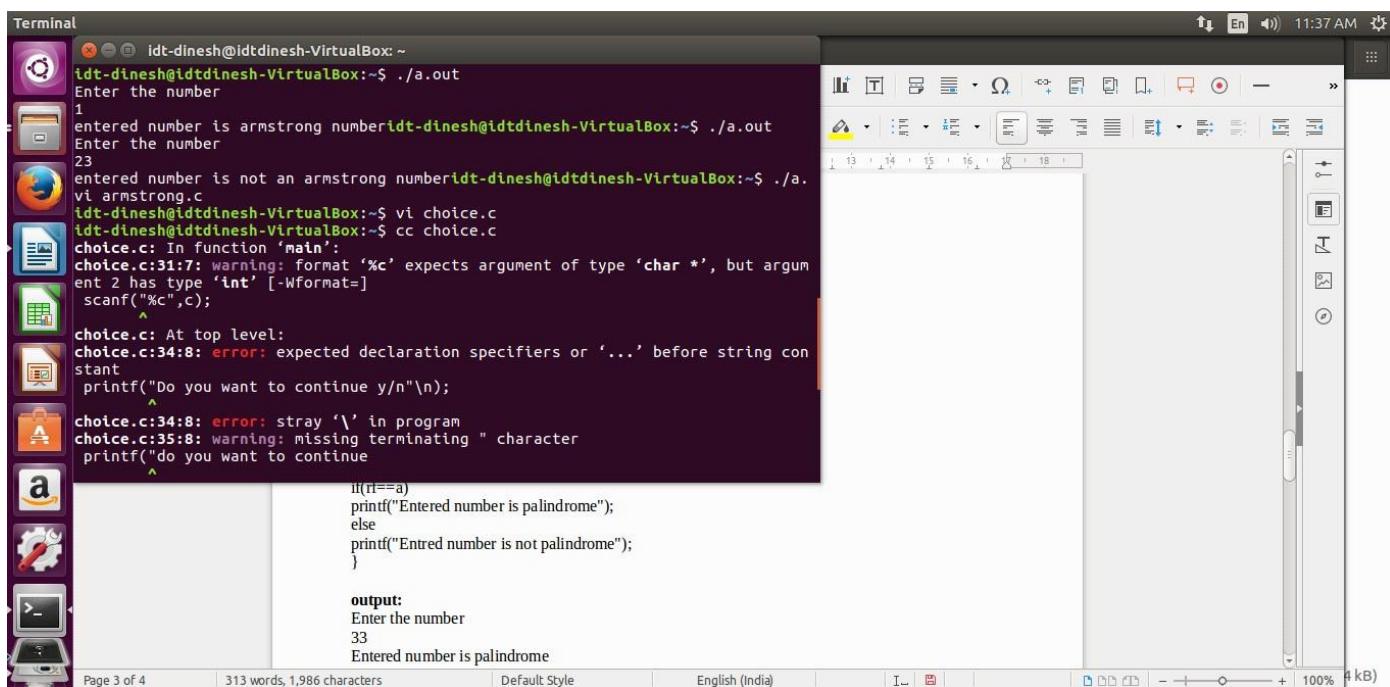
Programs:

Armstrong Number

```
#include<stdio.h>
void main()
{
int n,a=0,r=0,rf=0;
printf("Enter the number\n");
scanf("%d",&n);
a=n;
while(n!=0)
{
r=n%10;
rf+=r*r;
n=n/10;}if(rf==a)
printf("entered number is Armstrong number");
else
printf("entered number is not an Armstrong number");
}
```

Output:

Enter the number
153
entered number is Armstrong number



The screenshot shows a Linux desktop environment with a terminal window and a text editor window.

Terminal Window:

```
Terminal
idt-dinesh@idtdinesh-VirtualBox: ~
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
1
entered number is armstrong numberidt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
23
entered number is not an armstrong numberidt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
vi armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ vi choice.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc choice.c
choice.c: In function 'main':
choice.c:31:7: warning: format `"%c' expects argument of type 'char *', but argument 2 has type 'int' [-Wformat=]
    scanf("%c",c);
               ^
choice.c: At top level:
choice.c:34:8: error: expected declaration specifiers or '...' before string constant
    printf("Do you want to continue y/n"\n);
               ^
choice.c:34:8: error: stray '\`' in program
choice.c:35:8: warning: missing terminating " character
    printf("do you want to continue
               ^

```

Text Editor Window:

```
choice.c:34:8: error: stray '\`' in program
choice.c:35:8: warning: missing terminating " character
    printf("do you want to continue
               ^

```

The text editor shows the code for the Armstrong number program and the choice.c file. The choice.c file contains errors related to the use of single quotes and strings.

Factorial Number

```
#include<stdio.h>
void main()
{
int n,f=1,i;
printf("Enter the number\n");
scanf("%d",&n);
for(i=1;i<=n;i++)
f=f*i;
printf("factorial of %d is %d",n,f);}
```

Output:

Enter the number:4
factorial of 4 is 24

The screenshot shows a Linux desktop environment with a terminal window and a code editor. The terminal window on the left displays the execution of a C program to calculate factorials. It shows multiple runs of the program for different input values (3, 1, 2, 3) and also attempts to run other programs like 'armstrong' and 'choice'. The code editor on the right is showing the source code for these programs. The terminal window has a dark background with light-colored text, and the code editor has a light background with dark icons.

```
Terminal
idt-dinesh@idtdinesh-VirtualBox:~$ cc factorial.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
3
factorial of 3 is 6
idt-dinesh@idtdinesh-VirtualBox:~$ vi factorial.c
idt-dinesh@idtdinesh-VirtualBox:~$ vi armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
153
entered number is armstrong number
idt-dinesh@idtdinesh-VirtualBox:~$ 1
1: command not found
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
1
entered number is armstrong number
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
23
entered number is not an armstrong number
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.
vi armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ vi choice.c
choice.c: In function 'main':
choice.c:31:7: warning: format '%c' expects argument of type 'char *', but argument
      has type 'int' [-Wformat]
    if(n==a)
        printf("Entered number is palindrome");
    else
        printf("Entered number is not palindrome");
    }

output:
Enter the number
33
Entered number is palindrome
```

Palindrome

```
#include<stdio.h>
void main()
{
printf("Enter the number");

int n,a=0,r=0,rf=0;
scanf("%d",&n);
a=n;
while(n!=0)
{
r=n%10;
rf=rf*10+r;
```

```

n=n/10;
}
if(rf==a)
printf("Entered number is palindrome");
else
printf("Entered number is not palindrome");
}

```

Output:

Enter the number

33

Entered number is palindrome

The screenshot shows a Linux desktop environment with a terminal window open in the foreground and a LibreOffice Calc spreadsheet window in the background.

Terminal Window Content:

```

Terminal idt-dinesh@idtdinesh-VirtualBox: ~
idt-dinesh@idtdinesh-VirtualBox:~$ vi poly.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc poly.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number23
Entered number is not palindromeidt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number33
Entered number is palindromeidt-dinesh@idtdinesh-VirtualBox:~$ cat poly
cat: poly: No such file or directory
idt-dinesh@idtdinesh-VirtualBox:~$ vi poly.c
idt-dinesh@idtdinesh-VirtualBox:~$ vi sum.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc sum.c
sum.c: In function 'main':
sum.c:8:8: warning: too many arguments for format [-Wformat-extra-args]
    printf("Sum of a and b is : ",c);
    ^
idt-dinesh@idtdinesh-VirtualBox:~$ vi sum.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc sum.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
enter two numbers23
12
Sum of a and b is 35idt-dinesh@idtdinesh-VirtualBox:~$ vi table.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc table.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
    if(rf==a)
        printf("Entered number is palindrome");
    else
        printf("Entered number is not palindrome");
    }

output:
Enter the number
33
Entered number is palindrome

```

LibreOffice Calc Window:

A blank spreadsheet document is visible in the background, showing a standard grid of cells from A1 to B18.

Multiplication Table

```

#include<stdio.h>
void main()
{
int m,n,i;
printf("Enter which table you want");
scanf("%d",&m);
printf("Enter the range of the table");
scanf("%d",&n);
for(i=1;i<=n;i++)
printf("%d x %d = %d \n",i,m,i*m);
}

```

Output:

Enter which table you want

2

Enter the range of the table

3

$1 \times 2 = 2$

$2 \times 2 = 4$

$3 \times 2 = 6$

The screenshot shows a Linux desktop environment with a terminal window open in the foreground and a code editor window in the background.

Terminal Window Content:

```
idt-dinesh@idtdinesh-VirtualBox:~$ vi table.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc table.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter which table you want
Enter the range of the table
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
idt-dinesh@idtdinesh-VirtualBox:~$ vi factorial.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc factorial.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
3
factorial of 3 is 6
idt-dinesh@idtdinesh-VirtualBox:~$ vi armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ cc armstrong.c
idt-dinesh@idtdinesh-VirtualBox:~$ ./a.out
Enter the number
153
entered number is armstrong number
idt-dinesh@idtdinesh-VirtualBox:~$ 1
1: command not found
```

Code Editor Window Content:

```
if(n==a)
printf("Entered number is palindrome");
else
printf("Entered number is not palindrome");
```

Output:

```
Enter the number
33
Entered number is palindrome
```

Result:

Ex. No. 03

Install Google App Engine. Create hello world app and other simple web

Date:

Applications using python/java.

Aim:

Procedure:

Serverless computing > App Engine > Documentation



Google App Engine Documentation

[Send feedback](#)

App Engine is a fully managed, serverless platform for developing and hosting web applications at scale. You can choose from several popular languages, libraries, and frameworks to develop your apps, then let App Engine take care of provisioning servers and scaling your app instances based on demand.

Pick a language to learn more.

 Go View docs	 PHP View docs	 Java View docs
 Python View docs	 Node.js View docs	 .NET View docs
 Ruby View docs	 More languages View custom runtime docs	

1. Click View Docs Python 3.7

The screenshot shows the Google App Engine documentation page for Python 3.7. The main heading is "Choose your preferred environment". Below it, there are two sections: "Standard environment" and "Flexible environment". Each section has a list of bullet points and a "VIEW DOCS" button at the bottom.

Standard environment	Flexible environment
<ul style="list-style-type: none">The Python 3.7 runtime is capable of running any framework, library, or binary.Optimized to scale nearly instantaneously to handle huge traffic spikes.Free tier.The Python 2.7 runtime is not recommended because Python 2 is no longer supported by the community.	<ul style="list-style-type: none">Open source runtimes capable of running any framework, library, or binary.Greater CPU and memory instance types.Can access resources in the same Compute Engine network.Python 2.7 and 3.6No free tier. Application always has a minimum number of running instances. Most cost-effective for applications that serve traffic continuously.

2. Click Quick Start

The screenshot shows the "Google App Engine Python 3 Standard Environment documentation". The left sidebar has a "Documentation" section selected. The main content area displays the title "Google App Engine Python 3 Standard Environment documentation". Below the title, a paragraph explains that Python 3 apps run in secure sandboxes and scale automatically. There are six cards in a grid: "Quickstart" (Deploy your first app), "How-to guides" (Learn a task), "APIs and reference" (API, Configuration Files, App Engine Glossary), "Concepts" (Develop a deep understanding of Python 3 standard environment), "Tutorials" (Walkthroughs of common applications), and "Resources" (Pricing, quotas, and release notes).

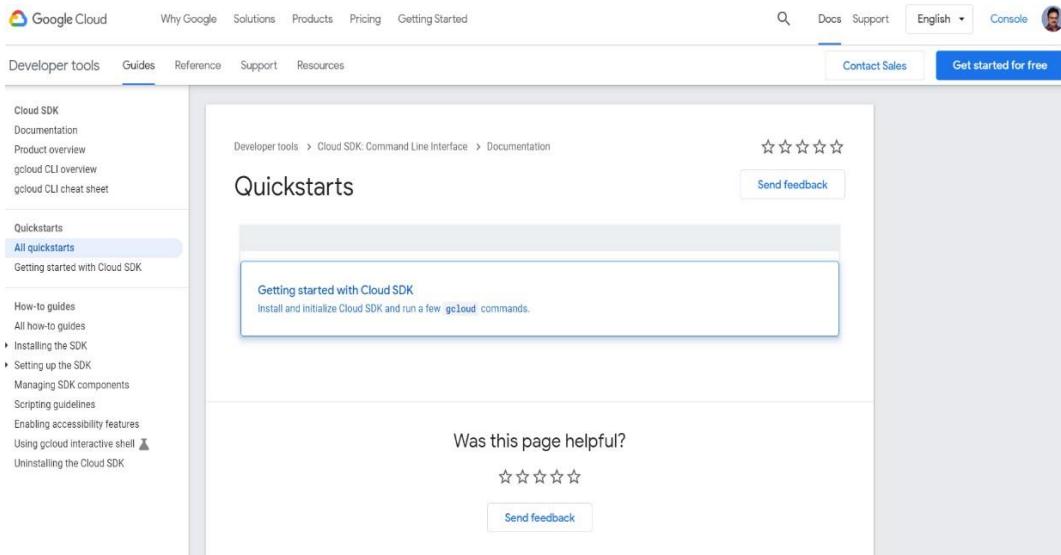
3. Download the Cloud SDK

The screenshot shows a web browser displaying the Google Cloud App Engine Python 3 Standard Environment Quickstart page at cloud.google.com/appengine/docs/standard/python3/quickstart. The page is titled "Serverless computing" and "Guides". The main content area discusses deploying a sample app on App Engine, mentioning no costs associated with running the guide. It includes a "Before you begin" section and a step-by-step process for downloading the Cloud SDK. A sidebar on the left lists various guides like "Google App Engine Documentation" and "Python 3 Standard Environment Quickstart". A sidebar on the right provides links for "Table of contents", "Costs", "Before you begin", "Download", "Run Hello world", "Deploy an app", "Clean up", and "What's next".

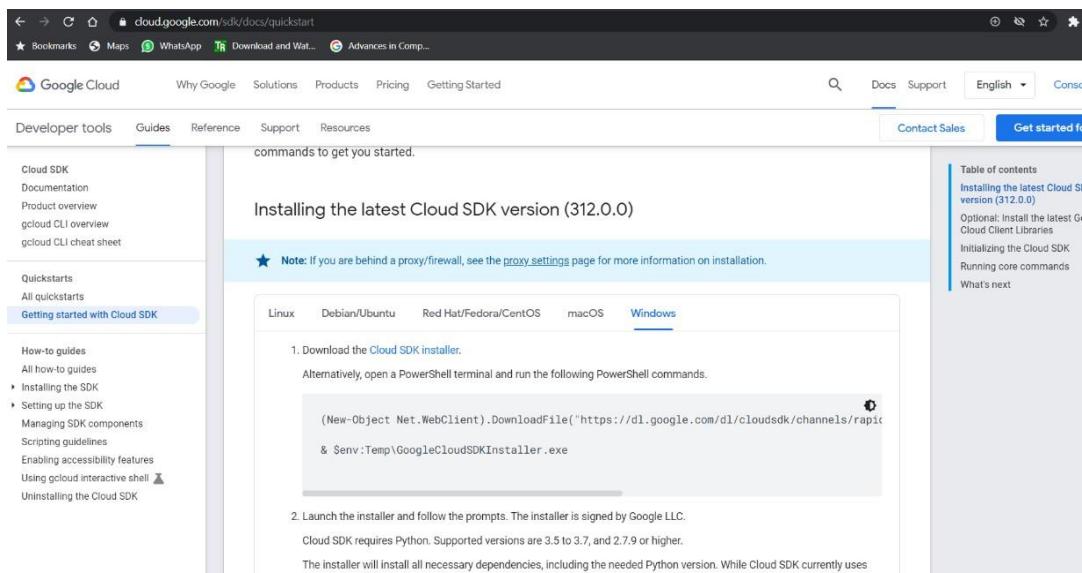
4. Click on Quickstarts (Install and initialize Cloud SDK)

The screenshot shows a web browser displaying the Google Cloud SDK documentation page at cloud.google.com/sdk/docs. The page is titled "Developer tools" and "Guides". The main content area is titled "Google Cloud SDK documentation" and describes the Cloud SDK as a set of tools for managing resources and applications on Google Cloud. It highlights the gcloud command-line tool and provides links to "Quickstarts", "How-to guides", "APIs and reference", "Concepts", "Support", "Resources", and "Solutions". The sidebar on the left lists "Cloud SDK Documentation", "Quickstarts", "How-to guides", and "Cloud SDK Components".

5. Click and install “Getting started with Cloud SDK”

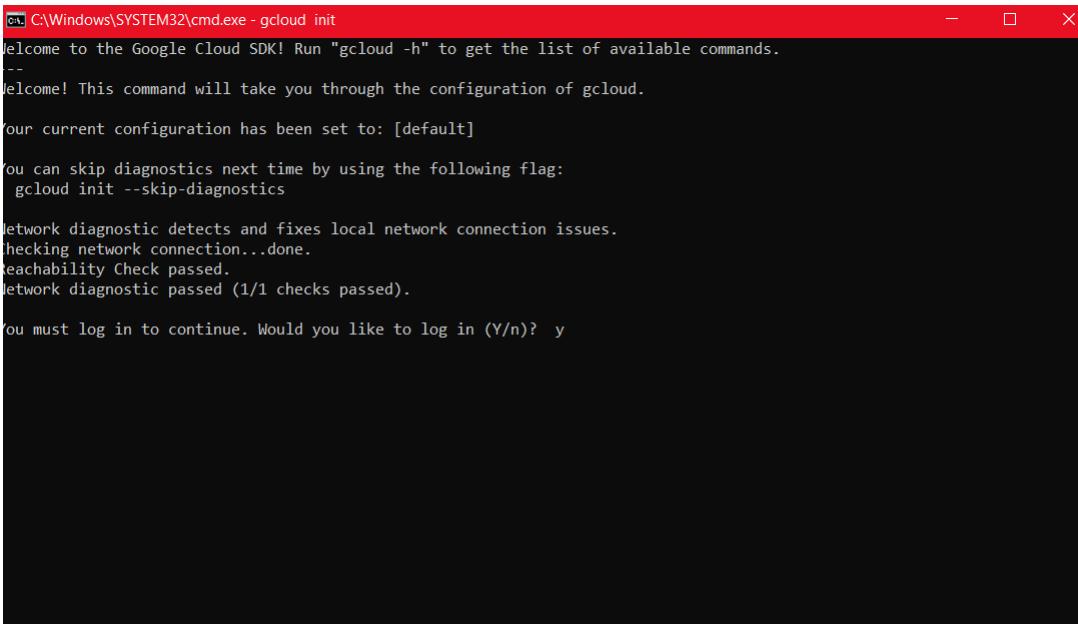


6. Click on Download the “Cloud SDK installer” for Windows



7. Install the Google Cloud SDK Setup

8. Google Cloud SDK will show up. Initialize for the first use by giving y (yes)



```
C:\Windows\SYSTEM32\cmd.exe - gcloud init
Welcome to the Google Cloud SDK! Run "gcloud -h" to get the list of available commands.
...
Welcome! This command will take you through the configuration of gcloud.

Your current configuration has been set to: [default]

You can skip diagnostics next time by using the following flag:
  gcloud init --skip-diagnostics

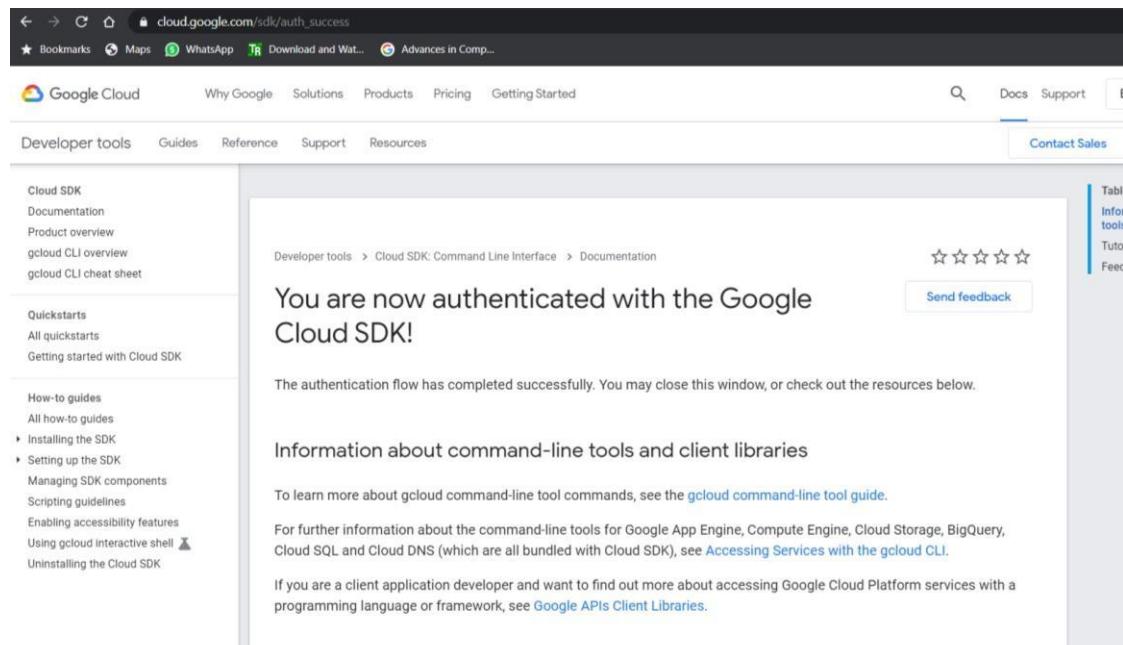
Network diagnostic detects and fixes local network connection issues.
  Checking network connection...done.
  Reachability Check passed.
  Network diagnostic passed (1/1 checks passed).

You must log in to continue. Would you like to log in (Y/n)? y
```

9. Login with your google account

10. Give access to Google Cloud SDK by clicking “Allow”

11. Now you are authenticated with the Google Cloud SDK (minimize this and proceed further)



The screenshot shows a web browser window with the URL `cloud.google.com/sdk/auth_success`. The page content is as follows:

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](#).

```
C:\Windows\SYSTEM32\cmd.exe - gcloud init
Your current configuration has been set to: [default]
You can skip diagnostics next time by using the following flag:
gcloud init --skip-diagnostics

Network diagnostic detects and fixes local network connection issues.
Checking network connection...done.
Reachability Check passed.
Network diagnostic passed (1/1 checks passed).

You must log in to continue. Would you like to log in (Y/n)? y

Your browser has been opened to visit:

https://accounts.google.com/o/oauth2/auth?client_id=32555940559.apps.googleusercontent.com&redirect_uri=http%3A%2F%2Flocalhost%3A8085%2F&scope=openid+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts.reauth&code_challenge=YHpyHgKjh-aalPx3ju6tJuctssCXETW8xHMEpb9o4w&code_challenge_method=S256&access_type=offline&response_type=code&prompt=select_account

You are logged in as: [nrajeshpandian@gmail.com].
Pick cloud project to use:
[1] mercurial-weft-291709
[2] savvy-surge-291709
[3] silent-fuze-291709
[4] Create a new project
Please enter numeric choice or text value (must exactly match list item):
```

12. Now start the python coding

Python file: (index.py)

```
import webapp2 # CGI standard - framework

class MainPage(webapp2.RequestHandler): # create a MainPage class
inherits from webapp module request handler

def get(self): # create a get function for get request for writes out hello
world

self.response.out.write('Hello World')

app = webapp2.WSGIApplication([('/', MainPage)],debug=True) # instance that routes incoming requests to handlers based on the URL

# Set the MainPage handler with its URL we put in the created handler
```

In app.yaml file:

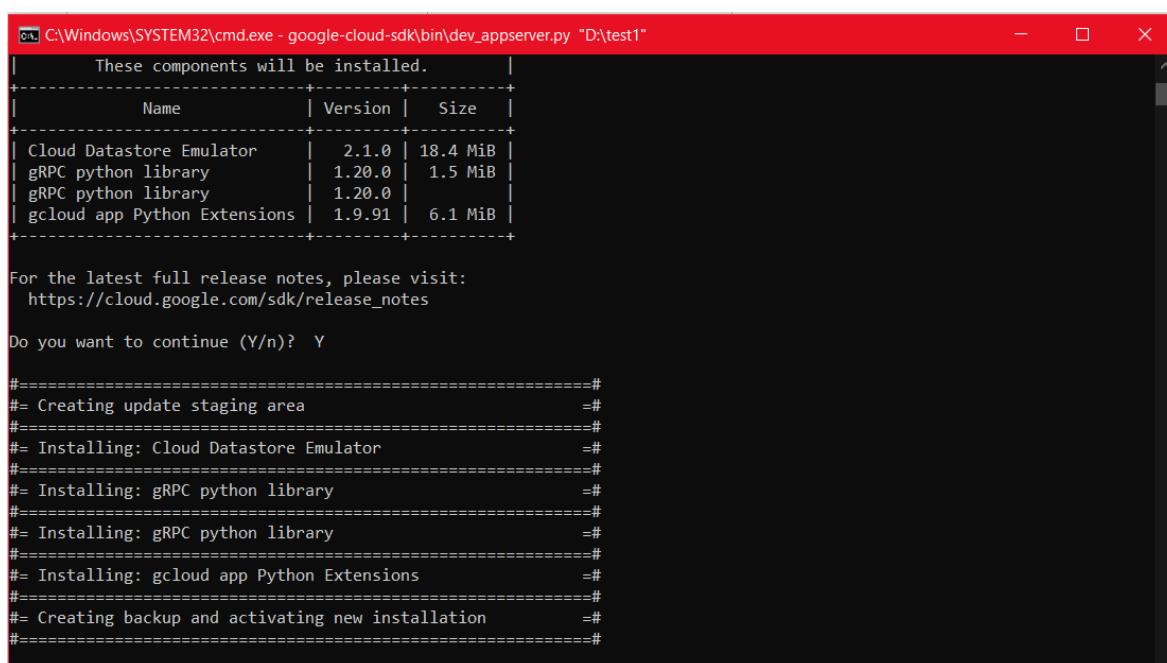
- Set python and its version as 27
- Next set the handlers we are going to use ('/')
- Set the name of the script as fname.app

13. Open the Google Cloud SDK Shell

14. Run the built-in development server located in the path google-cloud-sdk\bin\dev_appserver.py “file path”

15. Drag your folder with code files and press enter

16. Running for the first time means you have to install the python extension by giving y (yes)



The screenshot shows a Windows command prompt window titled 'C:\Windows\SYSTEM32\cmd.exe - google-cloud-sdk\bin\dev_appserver.py "D:\test1"'. The window displays the following output:

```
| These components will be installed. |
+-----+-----+
| Name | Version | Size |
+-----+-----+
| Cloud Datastore Emulator | 2.1.0 | 18.4 MiB |
| gRPC python library | 1.20.0 | 1.5 MiB |
| gRPC python library | 1.20.0 | |
| gcloud app Python Extensions | 1.9.91 | 6.1 MiB |
+-----+-----+

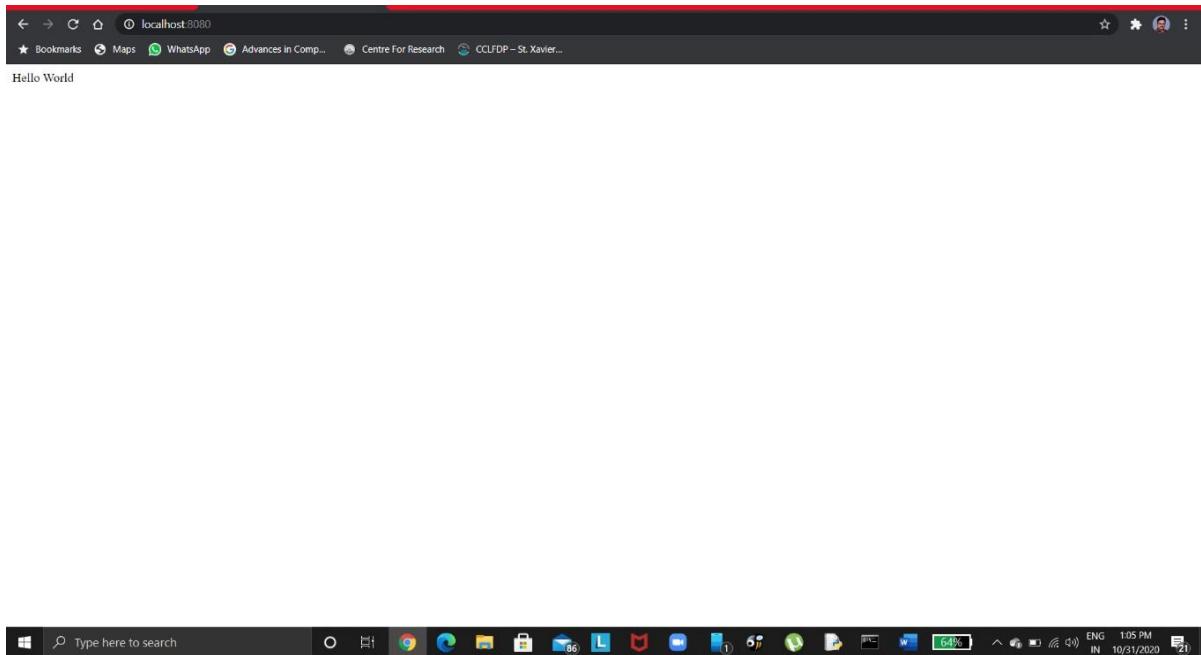
For the latest full release notes, please visit:
https://cloud.google.com/sdk/release_notes

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

#====#
#= Creating update staging area =#
#====#
#= Installing: Cloud Datastore Emulator =#
#====#
#= Installing: gRPC python library =#
#====#
#= Installing: gRPC python library =#
#====#
#= Installing: gcloud app Python Extensions =#
#====#
#= Creating backup and activating new installation =#
#====#
```

Output:

Open the browser: URL - Localhost:8080



Result:

Ex. No. 04

Date:

Use GAE launcher to launch the web applications.

Aim:

Procedure:

Python file: (todo.py)

```
from google.appengine.api import users
from google.appengine.ext import webapp
from google.appengine.ext.webapp.util import run_wsgi_app
from google.appengine.ext import db
from google.appengine.ext.webapp import template
from google.appengine.api import mail

# Todo defines the data model for the Todos
# as it extends db.model the content of the class will automatically
# stored

class TodoModel(db.Model):
    author      = db.UserProperty(required=True)
    shortDescription = db.StringProperty(required=True)

    longDescription = db.StringProperty(multiline=True)
    url          = db.StringProperty()
    created      = db.DateTimeProperty(auto_now_add=True)
    updated      = db.DateTimeProperty(auto_now=True)
    dueDate      = db.StringProperty(required=True)
    finished     = db.BooleanProperty()

# The main page where the user can login and logout
```

```
# MainPage is a subclass of webapp.RequestHandler and overwrites  
the get method
```

```
class MainPage(webapp.RequestHandler):  
    def get(self):  
        user = users.get_current_user()  
        url = users.create_login_url(self.request.uri)  
        url_linktext = 'Login'  
        if user:  
            url = users.create_logout_url(self.request.uri)  
            url_linktext = 'Logout'  
  
    # GQL is similar to SQL  
    todos = TodoModel.gql("WHERE author = :author and  
finished=false",  
                          author=users.get_current_user())  
  
    values = {  
        'todos': todos,  
        'numbertodos' : todos.count(),  
        'user': user,  
        'url': url,  
        'url_linktext': url_linktext,  
    }  
    self.response.out.write(template.render('index.html', values))  
  
    # This class creates a new Todo item  
    class New(webapp.RequestHandler):  
        def post(self):  
            user = users.get_current_user()  
            if user:  
                testurl = self.request.get('url')  
                if not testurl.startswith("http://") and testurl:  
                    testurl = "http://" + testurl  
                todo = TodoModel(  
                    author = users.get_current_user(),  
                    shortDescription = self.request.get('shortDescription'),  
                    longDescription = self.request.get('longDescription'),  
                    dueDate = self.request.get('dueDate'),  
                    url = testurl,  
                    finished = False)  
                todo.put();
```

```

        self.redirect('/')

# This class deletes the selected Todo
class Done(webapp.RequestHandler):
    def get(self):
        user = users.get_current_user()
        if user:
            raw_id = self.request.get('id')
            id = int(raw_id)
            todo = TodoModel.get_by_id(id)
            todo.delete()
            self.redirect('/')

#This class emails the task to yourself

class Email(webapp.RequestHandler):
    def get(self):
        user = users.get_current_user()
        if user:
            raw_id = self.request.get('id')
            id = int(raw_id)
            todo = TodoModel.get_by_id(id)
            message      =      mail.EmailMessage(sender=user.email(),
subject=todo.shortDescription)
            message.to = user.email()
            message.body = todo.longDescription
            message.send()
            self.redirect('/')

# Register the URL with the responsible classes

application = webapp.WSGIApplication(
    [('/', MainPage),
     ('/new', New),
     ('/done', Done),
     ('/email', Email)],
    debug=True)

# Register the wsgi application to run

def main():
    run_wsgi_app(application)

```

```
if __name__ == "__main__":
    main()
```

In app.yaml file

```
application: GAE launcher
```

```
version: 1
```

```
runtime: python
```

```
api_version: 1
```

```
handlers:
```

```
- url: /css
```

```
  static_dir: css
```

```
- url: /images
```

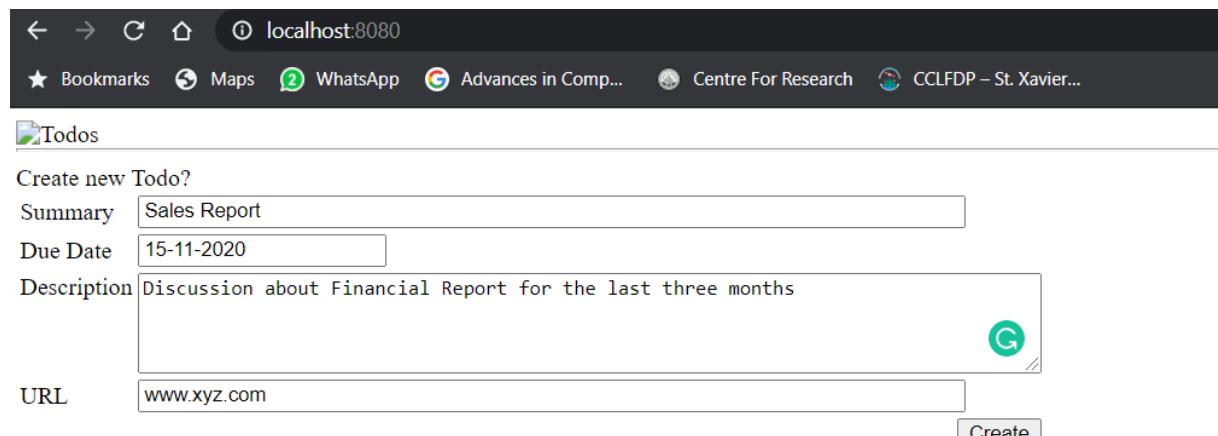
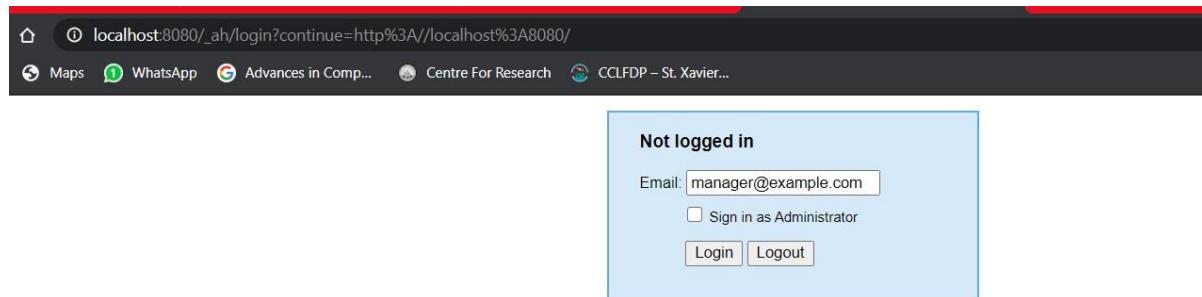
```
  static_dir: images
```

```
- url: /.*
```

```
  script: todo.py
```

Output:

Open the browser: URL - Localhost:8080



localhost:8080

Bookmarks Maps WhatsApp Advances in Comp... Centre For Research CCLFDP – St. Xavier...

Todos

You have a total number of 1 Todos.

Short description	Due Date	Long Description	URL	Created	Updated	Done	Send Email reminder
Sales Report	15-11-2020	Discussion about Financial Report for the last three months	http://www.xyz.com	10.11.2020	10.11.2020	Done	Email

Create new Todo?

Summary

Due Date

Description

URL

Result:

Ex. No. 05

Date:

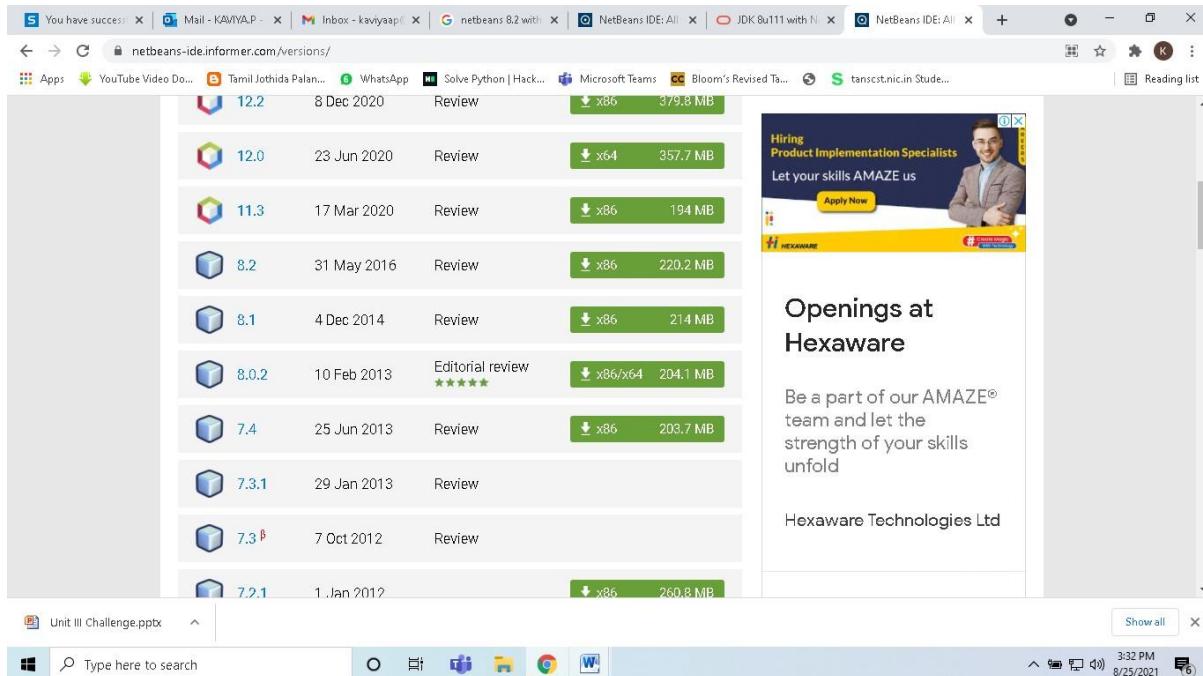
Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

Aim:

Procedure:

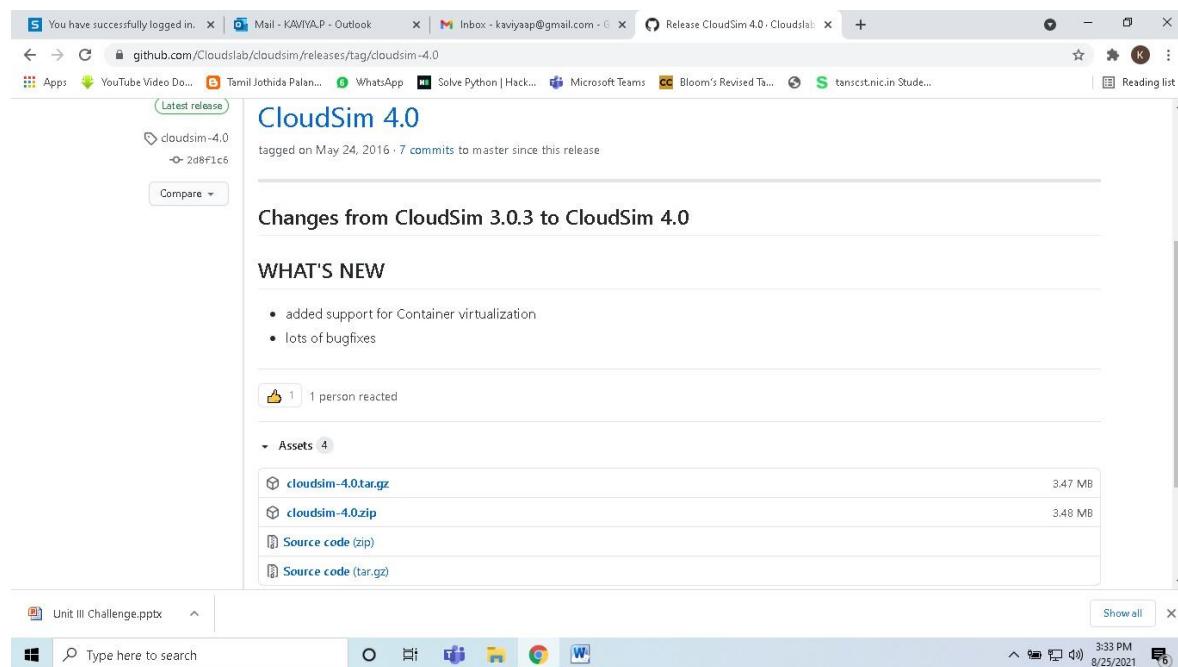
1. Download NetBeans and intall it.

Link: <https://netbeans-ide.informer.com/versions/>

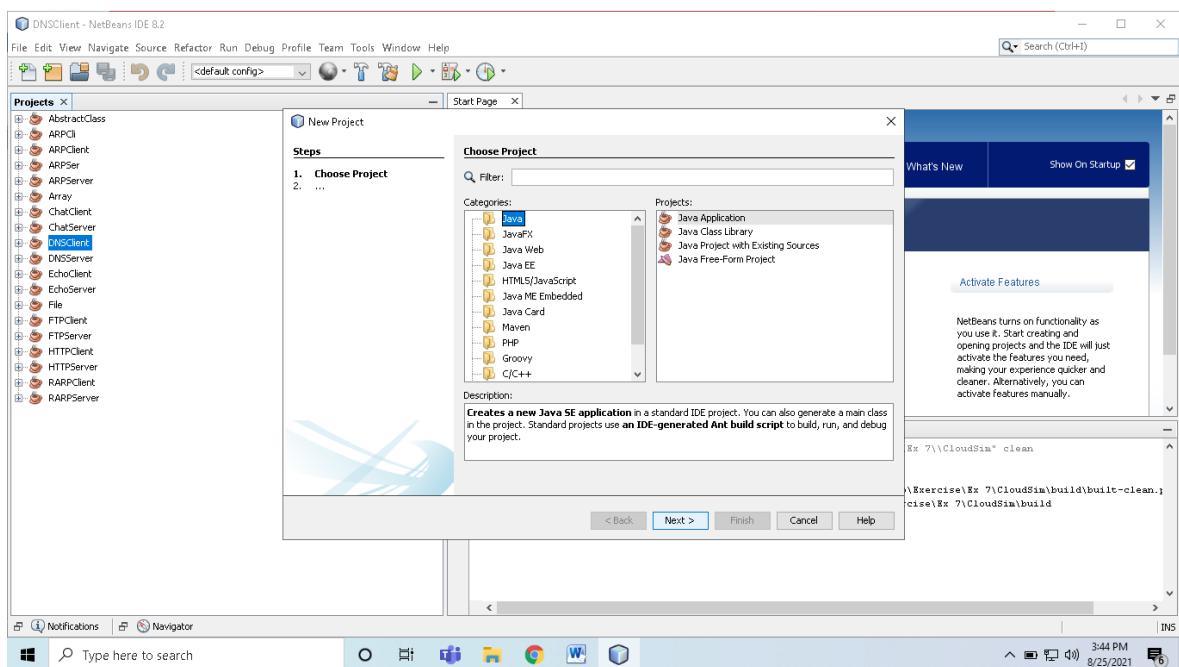
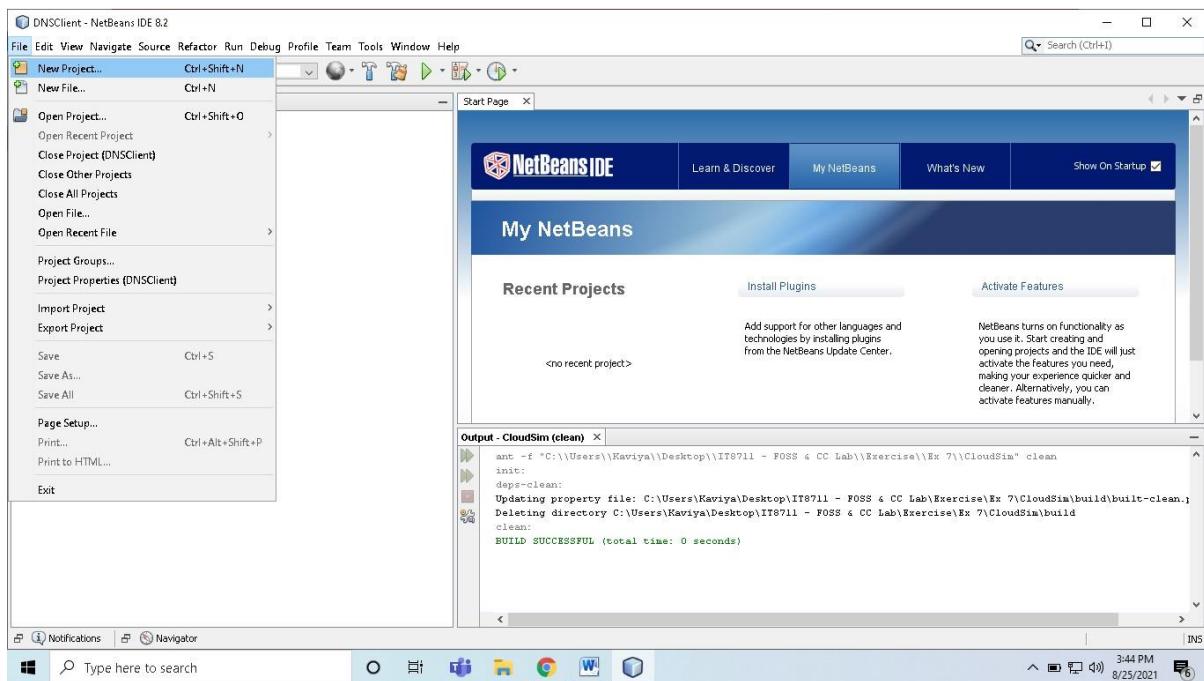


2. Download CloudSim 4.0 (jar and source code), extract and place it in respective folders.

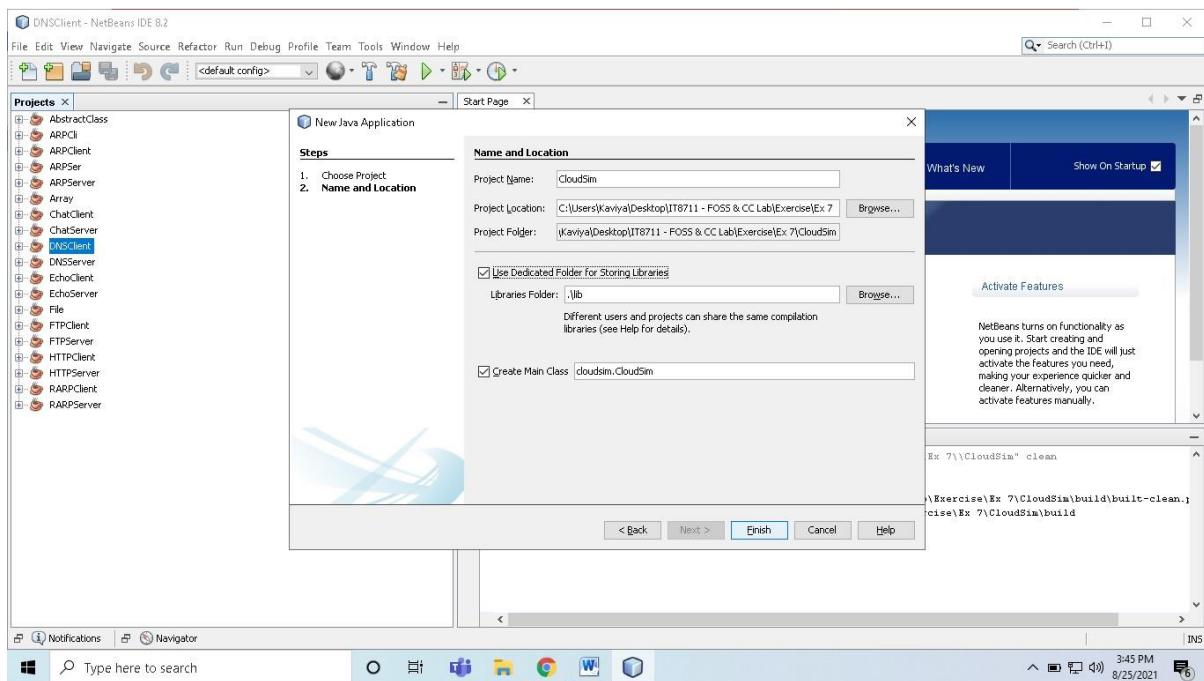
Link: <https://github.com/Cloudslab/cloudsim/releases/tag/cloudsim-4.0>



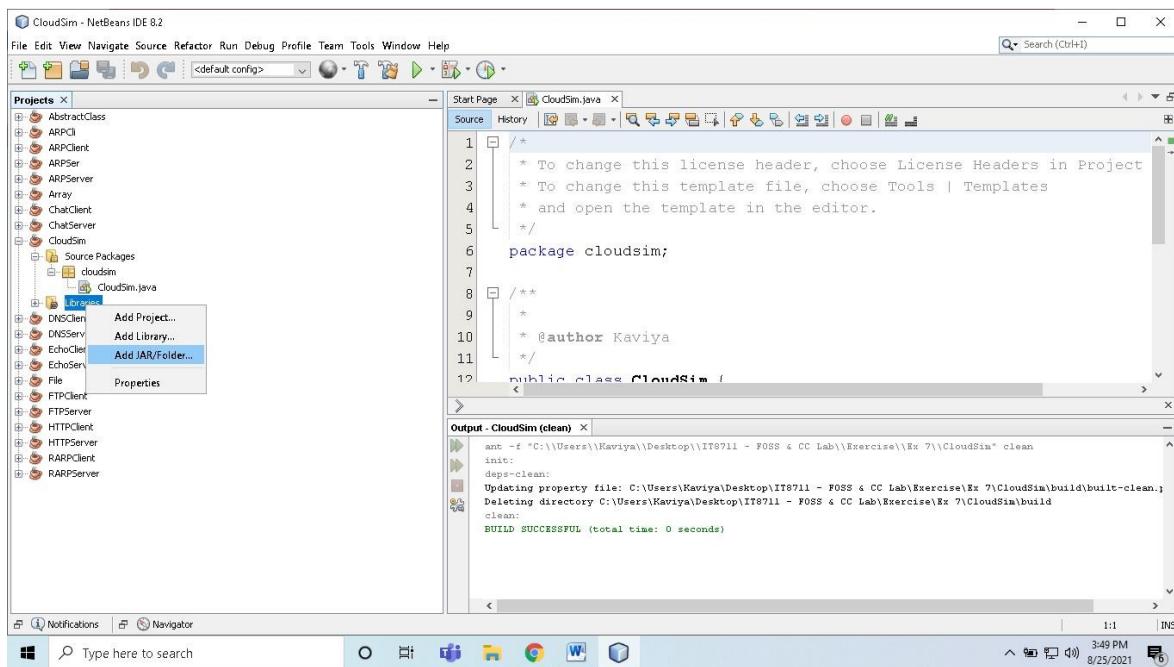
3. Open NetBeans & Create a project “CloudSim”

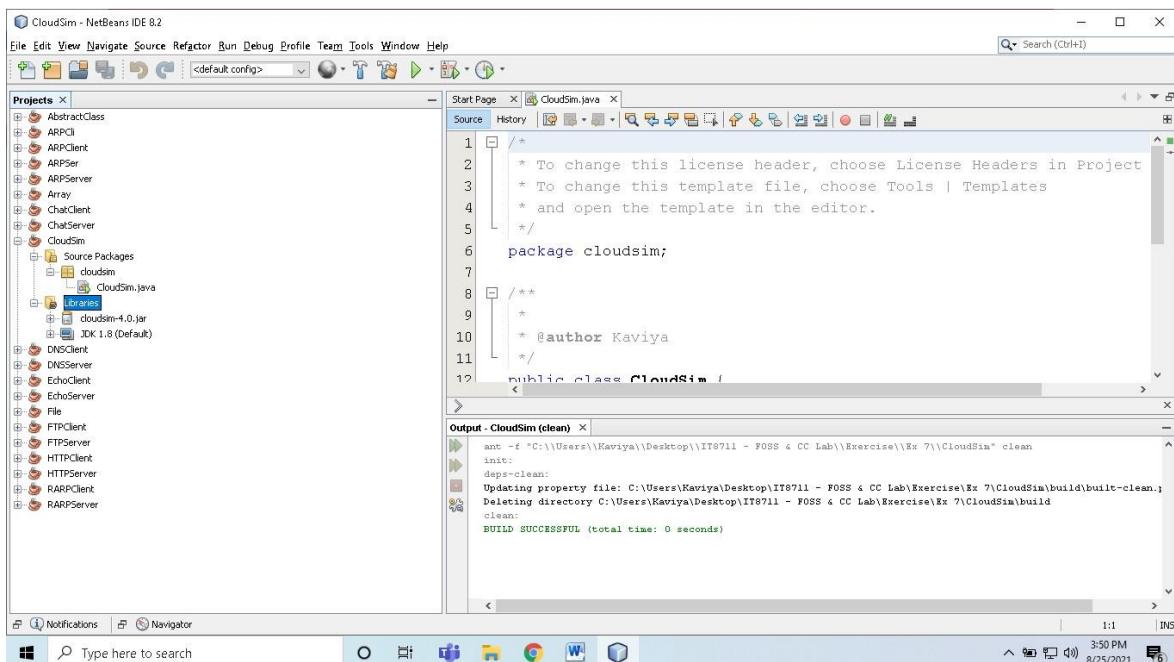
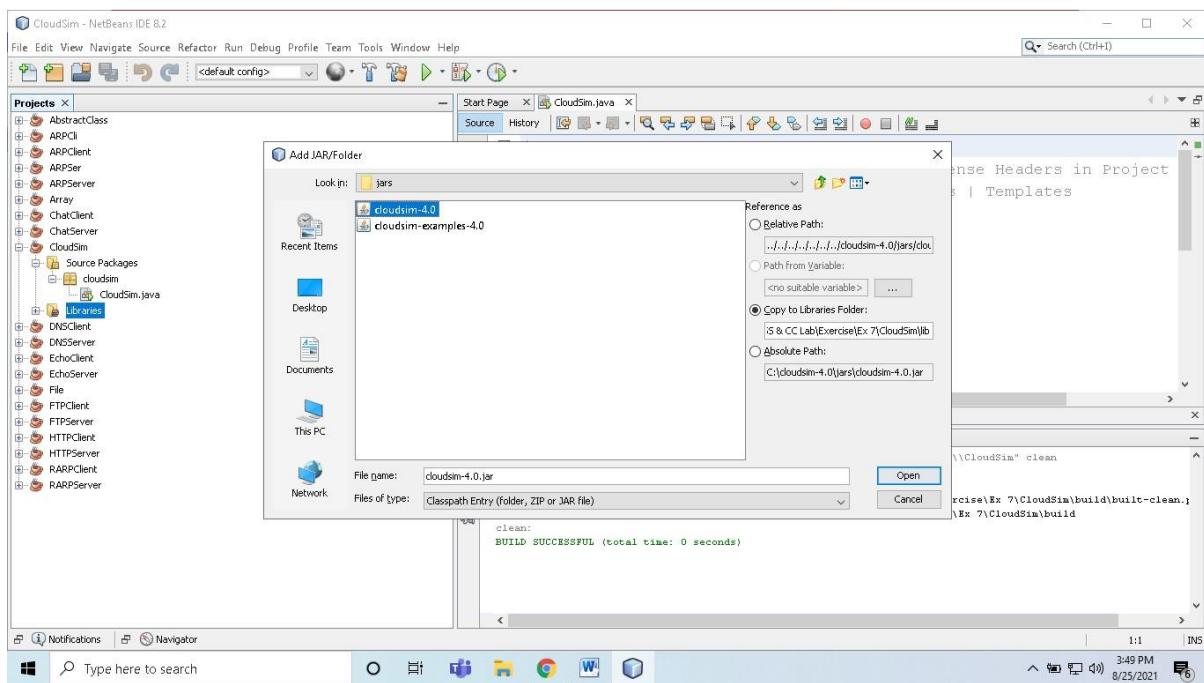


4. Check “Use dedicated folder for Storing Libraries” → Click “Finish”

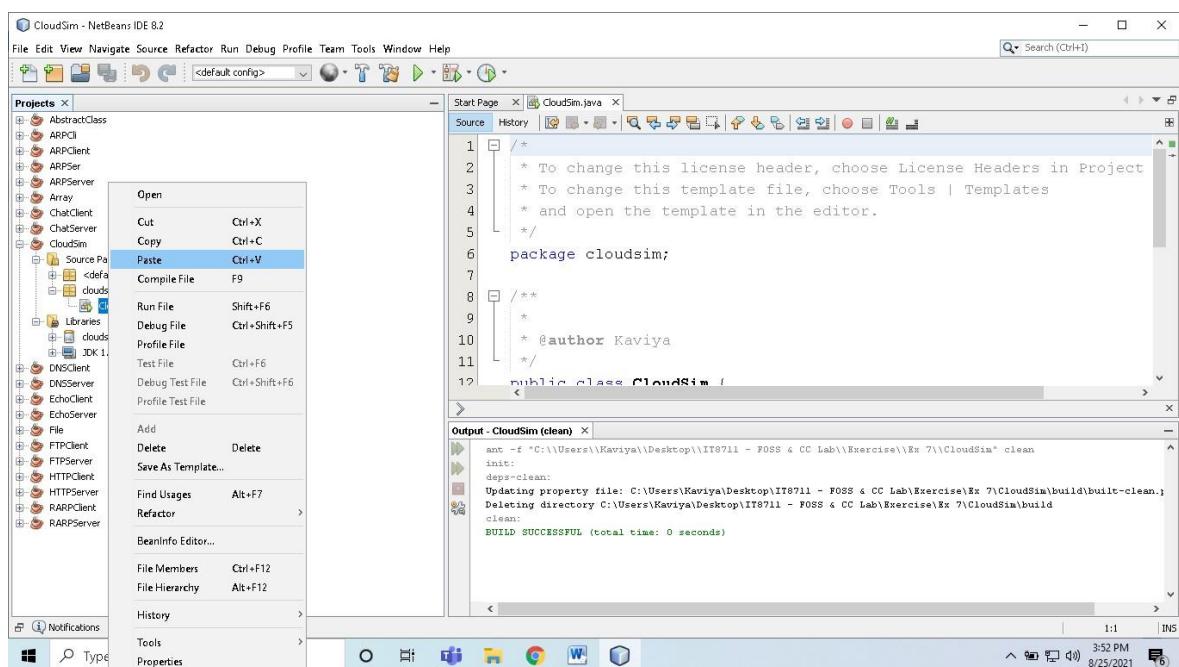
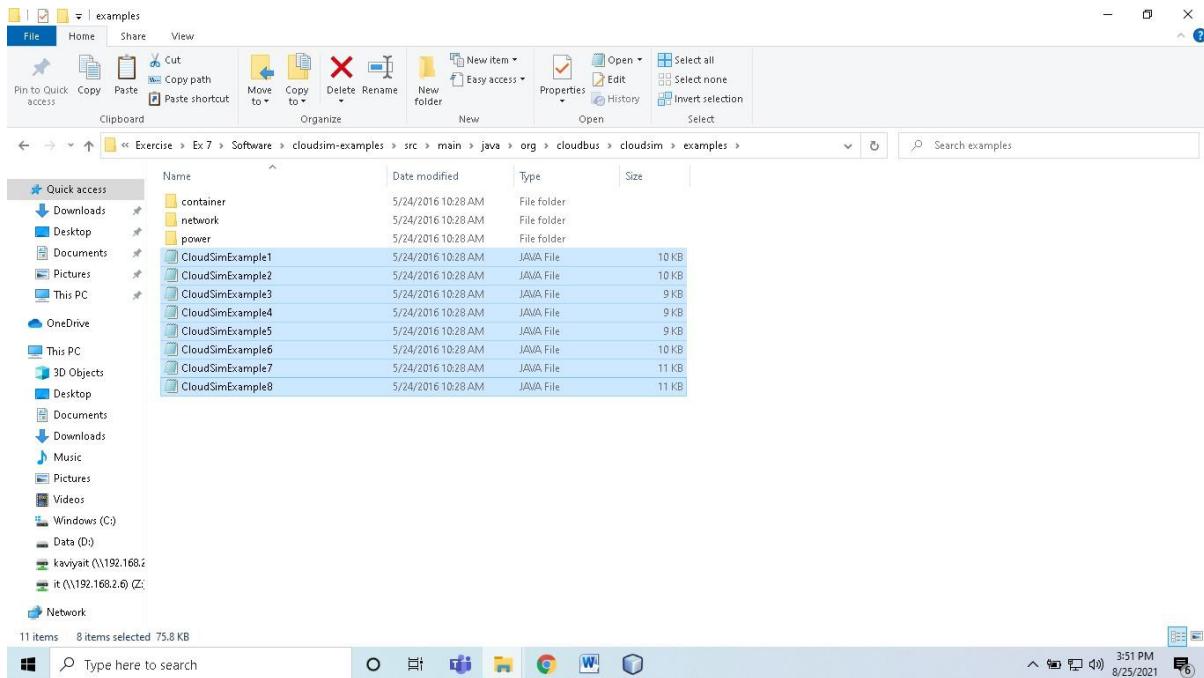


5. Add jar file “cloudsim-4.0”





6. Add “cloudsim examples” in CloudSim Project



CloudSim - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Start Page CloudSim.java CloudSimExample1.java CloudSimExample2.java

Projects

- AbstractClass
- APPCl
- APPClient
- APPSer
- APPSer
- Array
- ChatClient
- ChatServer
- CloudSim
- Source Packages cloudsim
 - CloudSim.java
 - CloudSimExample1.java
 - CloudSimExample2.java
 - CloudSimExample3.java
 - CloudSimExample4.java
 - CloudSimExample5.java
 - CloudSimExample6.java
 - CloudSimExample7.java
 - CloudSimExample8.java
- Libraries cloudsim-4.0.jar Jdk 1.8 (Default)
- DNSClient
- DNSServer
- EchoClient
- EchoServer
- File
- FTPClient
- FTPServer
- HTTPClient
- HTTPServer
- random

Output - CloudSim(clean) x

```
ant -f "C:\\\\Users\\\\Kaviya\\\\Desktop\\\\IT8711 - FOSS & CC Lab\\\\Exercise\\\\Ex 7\\\\CloudSim" clean
init:
deps-clean:
Created dir: C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build
Updating property file: C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build\\built-clean.
Deleting directory C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

Notifications Navigator

Type here to search

10:1 IN5 3:58 PM 8/25/2021

```
/*
 * Title: CloudSim Toolkit
 * Description: CloudSim (Cloud Simulation) Toolkit for Modeling a
 * of Clouds
 * Licence: GPL - http://www.gnu.org/copyleft/gpl.html
 *
 * Copyright (c) 2009, The University of Melbourne, Australia
 */
package cloudsim;
```

```
ant -f "C:\\\\Users\\\\Kaviya\\\\Desktop\\\\IT8711 - FOSS & CC Lab\\\\Exercise\\\\Ex 7\\\\CloudSim" clean
init:
deps-clean:
Created dir: C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build
Updating property file: C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build\\built-clean.
Deleting directory C:\\Users\\Kaviya\\Desktop\\IT8711 - FOSS & CC Lab\\Exercise\\Ex 7\\CloudSim\\build
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

CloudSim - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Start Page CloudSim.java CloudSimExample1.java CloudSimExample2.java

Projects

- AbstractClass
- APPCl
- APPClient
- APPSer
- APPSer
- Array
- ChatClient
- ChatServer
- CloudSim
- Source Packages cloudsim
 - CloudSim.java
 - CloudSimExample1.java
 - CloudSimExample2.java
 - CloudSimExample3.java
 - CloudSimExample4.java
 - CloudSimExample5.java
 - CloudSimExample6.java
 - CloudSimExample7.java
 - CloudSimExample8.java
- Libraries cloudsim-4.0.jar Jdk 1.8 (Default)
- DNSClient
- DNSServer
- EchoClient
- EchoServer
- File
- FTPClient
- FTPServer
- HTTPClient
- HTTPServer
- random

Output - CloudSim(run) x

```
>>> run:
```

Type here to search

10:1 IN5 4:01 PM 8/25/2021

```
51 : vmlist; */
52 : static List<Vm> vmlist;
53 :
54 :
55 : states main() to run this example
56 :
57 : static void main(String[] args) {
58 :
59 : Log.printLine("Starting CloudSimExample2...");
60 :
61 : try {
62 :     // First step: Initialize the CloudSim package. It should be called
63 :     // before creating any entities.
64 :     int num_user = 1; // number of cloud users
65 :     Calendar calendar = Calendar.getInstance();
66 :     boolean trace_flag = false; // mean trace events
67 :
68 :     // Initialize the CloudSim library
69 :     CloudSim.init(num_user, calendar, trace_flag);
70 :
71 :     // Second step: Create Datacenters
72 :     //Datacenters are the resource providers in CloudSim. We need at least one of them to
73 :     @SuppressWarnings("unused")
```

CloudSim - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Start Page CloudSim.java CloudSimExample1.java CloudSimExample2.java

Projects x

- AbstractClass
- APPCl
- APPClient
- APPSe
- APPSever
- Arr
- ChatClient
- ChatServer
- CloudSim
 - Source Packages
 - cloudsim
 - CloudSim.java
 - CloudSimExample1.java
 - CloudSimExample2.java
 - CloudSimExample3.java
 - CloudSimExample4.java
 - CloudSimExample5.java
 - CloudSimExample6.java
 - CloudSimExample7.java
 - CloudSimExample8.java
 - Libraries
 - cloudsim-4.0.jar
 - JDK 1.8 (Default)
- DNSClient
- DNSSever
- EchoClient
- EchoSever
- File
- FTPClient
- FTPServer
- HTTPClient
- HTTPSever
- nnclient

Notifications Navigator

Type here to search

Output - CloudSim (run) > run:

```

132    //bind the cloudlets to the vms. This way, the broker
133    // will submit the bound cloudlets only to the specific VM
134    broker.bindCloudletToVm(cloudlet1.getId(),vm1.getId());
135    broker.bindCloudletToVm(cloudlet2.getId(),vm2.getId());
136
137
138    // Sixth step: Starts the simulation
139    CloudSim.startSimulation();
140
141
142    // Final step: Print results when simulation is over
143    List<Cloudlet> newList = broker.getCloudletReceivedList();
144
145    CloudSim.stopSimulation();
146
147    printCloudletList(newList);
148
149    Log.println("CloudSimExample2 finished!");
150
151
152    catch (Exception e) {
153        e.printStackTrace();
154        Log.println("The simulation has been terminated due to an unexpected error");
155    }

```

10:1 IN5 4:02 PM 8/25/2021

7. Run the CloudSim examples

CloudSim - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Start Page CloudSim.java CloudSimExample1.java CloudSimExample2.java

Projects x

- AbstractClass
- APPCl
- APPClient
- APPSe
- APPSever
- Arr
- ChatClient
- ChatServer
- CloudSim
 - Source Packages
 - cloudsim
 - CloudSim.java
 - CloudSimExample1.java
 - CloudSimExample2.java
 - CloudSimExample3.java
 - CloudSimExample4.java
 - CloudSimExample5.java
 - CloudSimExample6.java
 - CloudSimExample7.java
 - CloudSimExample8.java
 - Libraries
 - cloudsim-4.0.jar
 - JDK 1.8 (Default)
- DNSClient
- DNSSever
- EchoClient
- EchoSever
- File
- FTPClient
- FTPServer
- HTTPClient
- HTTPSever
- nnclient

Notifications Navigator

Type here to see

Run File Shift+F6

Output - CloudSim (clean) >

```

1  /*
2   * Title:      CloudSim Toolkit
3   * Description: CloudSim (Cloud Simulation) Toolkit for Modeling a
4   *              of Clouds
5   * Licence:    GPL - http://www.gnu.org/copyleft/gpl.html
6
7   * Copyright (c) 2009, The University of Melbourne, Australia
8   */
9
10 package cloudsim;
11
12

```

```

ant -f "C:\Users\Kaviya\Desktop\IT8711 - FOSS & CC Lab\Exercise\Ex 7\CloudSim" clean
init:
deps-clean:
Created dir: C:\Users\Kaviya\Desktop\IT8711 - FOSS & CC Lab\Exercise\Ex 7\CloudSim\build
Updating property file: C:\Users\Kaviya\Desktop\IT8711 - FOSS & CC Lab\Exercise\Ex 7\CloudSim\build\built-clean.
Deleting directory C:\Users\Kaviya\Desktop\IT8711 - FOSS & CC Lab\Exercise\Ex 7\CloudSim\build
clean:
BUILD SUCCESSFUL (total time: 0 seconds)

```

10:1 IN5 3:59 PM 8/25/2021

8. Output for CloudSimExample2.java

The screenshot shows the NetBeans IDE 8.2 interface. The left pane displays the 'Projects' view with the 'CloudSim' project selected, showing various source packages like 'AbstractClass', 'APPCl', 'APPClient', etc., and Java files such as 'CloudSim.java', 'CloudSimExample1.java', 'CloudSimExample2.java', and 'CloudSimExample8.java'. The right pane shows the 'Output - CloudSim (run)' window with the following log output:

```
Starting CloudSimExample2...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.0: Broker: Trying to Create VM #1 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: VM #1 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
0.1: Broker: Sending cloudlet 1 to VM #1
1000.1: Broker: Cloudlet 0 received
1000.1: Broker: Cloudlet 1 received
1000.1: Broker: All Cloudlets executed. Finishing...
1000.1: Broker: Destroying VM #0
1000.1: Broker: Destroying VM #1
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
CloudSimExample2 finished!
```

The status bar at the bottom indicates the time as 4:00 PM and the date as 8/25/2021.

===== OUTPUT =====

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start	Time
Finish Time						
0	SUCCESS	2	0	400		0.1
400.1						

*****Datacenter: Datacenter_0*****

User id	Debt
3	35.6

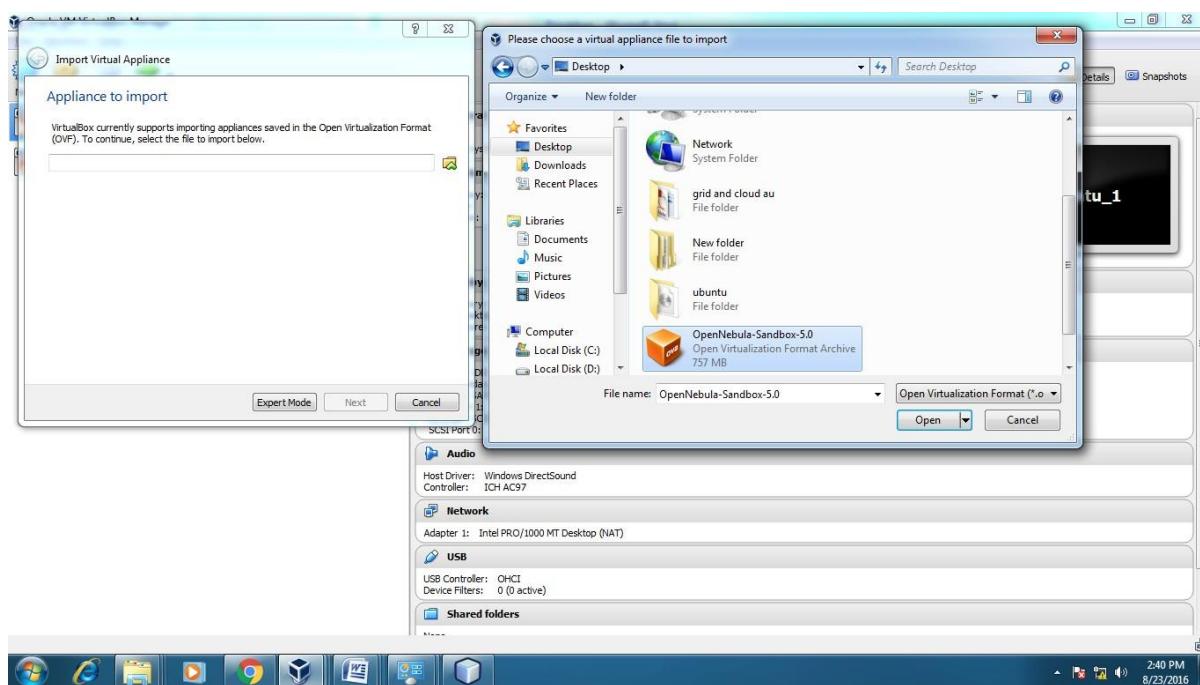
CloudSimExample1 finished!

Result:

Ex. No. 06	Find a procedure to transfer the files from one virtual machine to another virtual machine.
Date:	

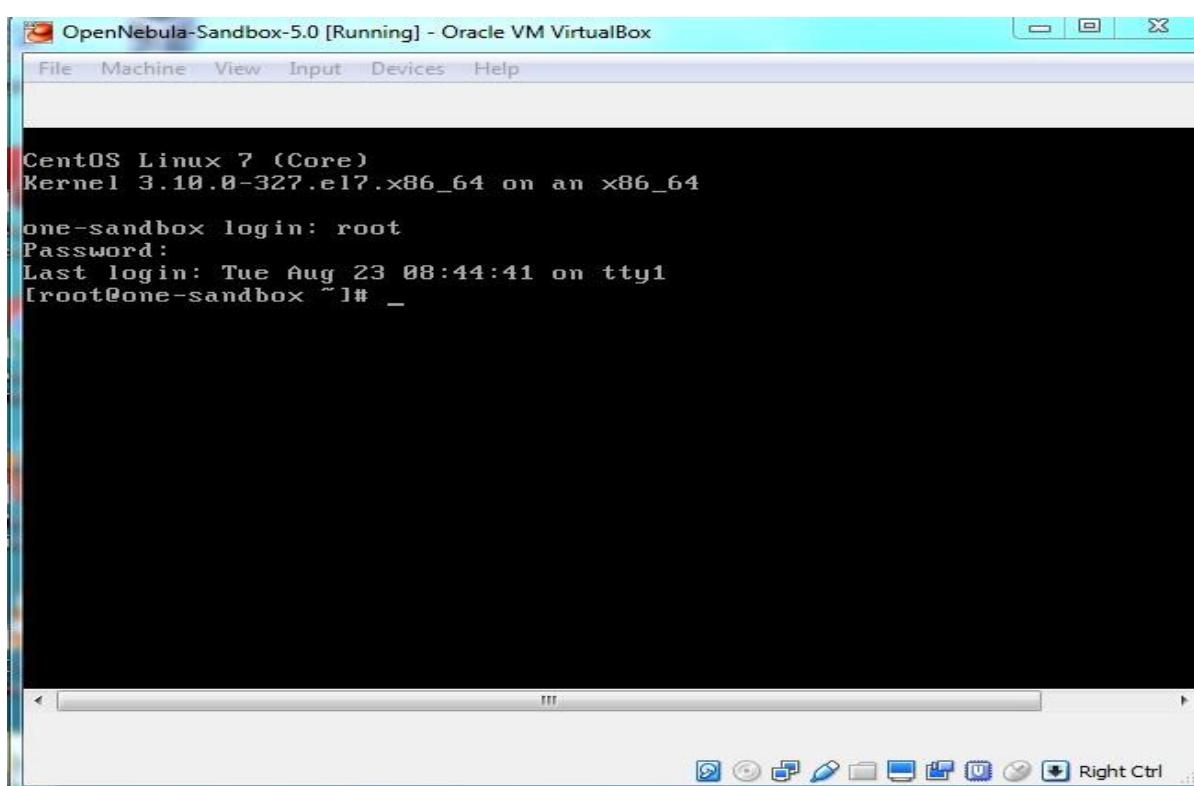
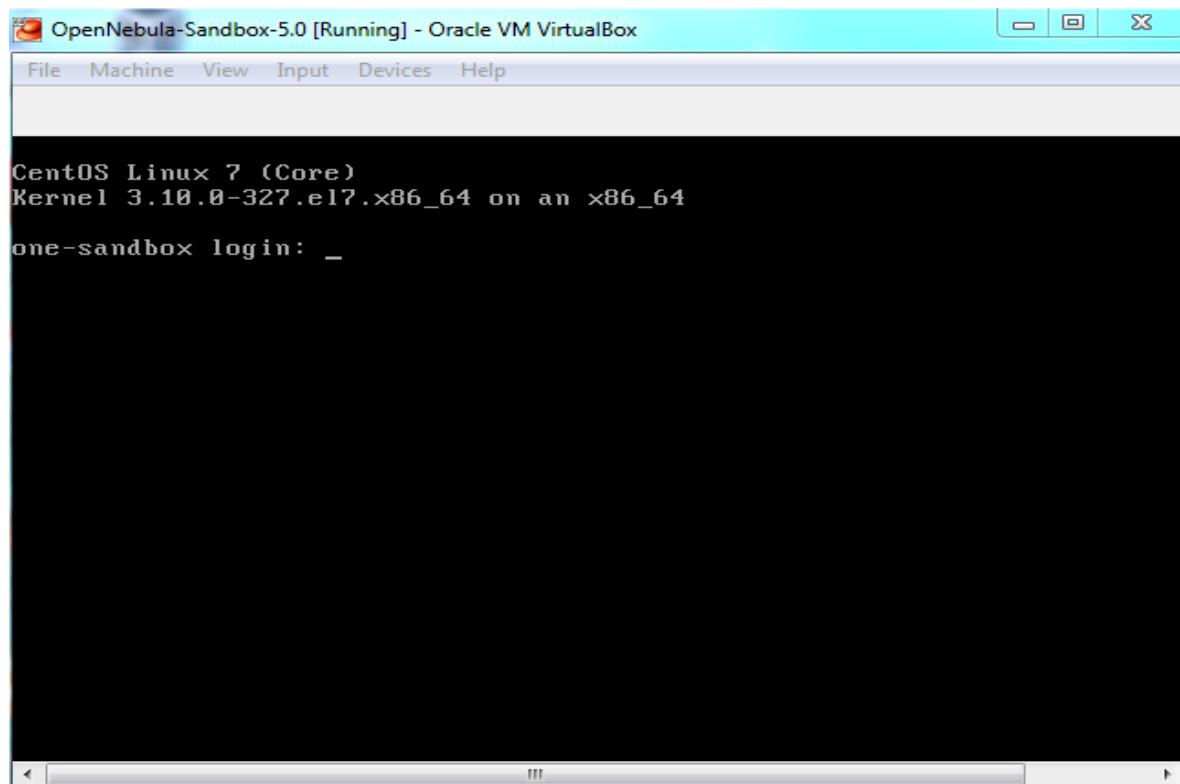
Aim:

Procedure:

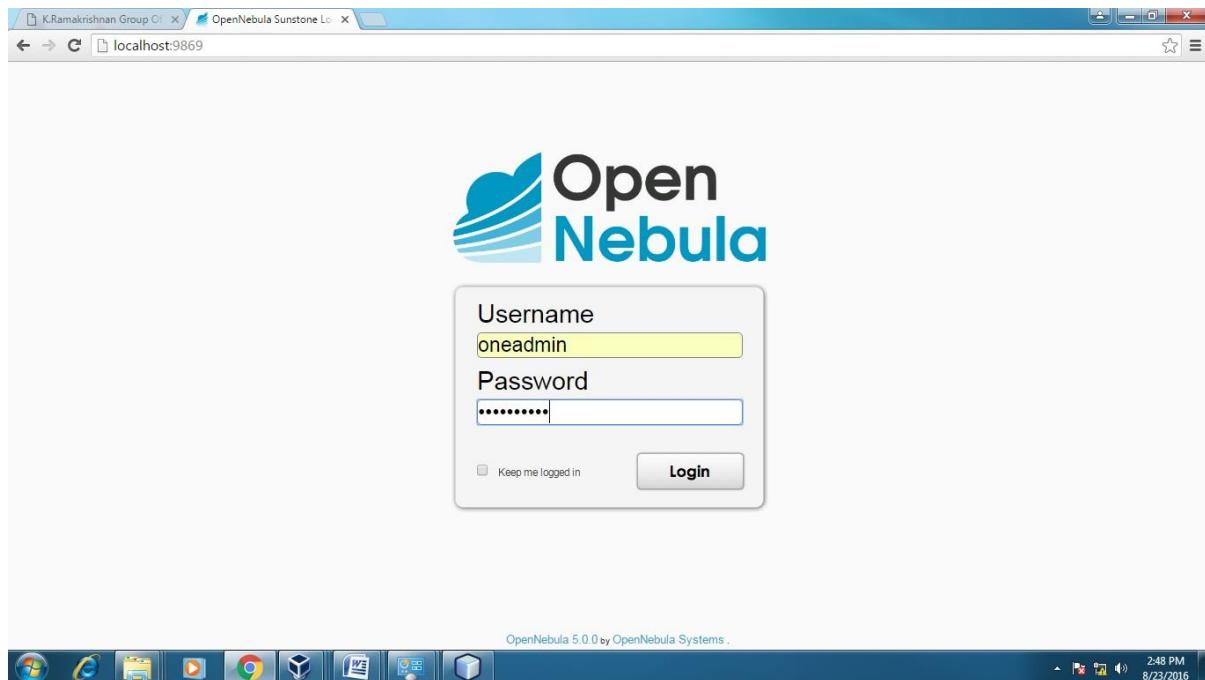


1. Select the OS image and click import.
2. Allocate the default resources to the Guest OS from the host machine and click next.
3. Then go to setting, select USB option in the left pane and select USB 1.1.
4. Click Start to Power ON the VM.

5. Look for the Command Line Interface (CLI) of the Virtual Machine.
6. Login: root, Password: opennebula

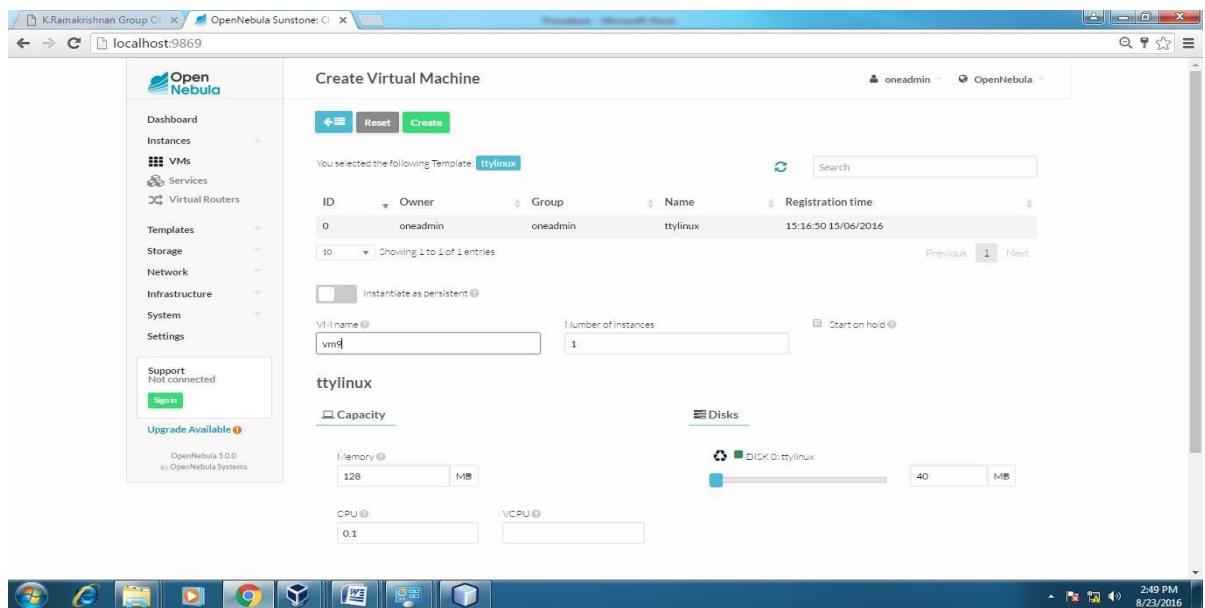


7. Now open the browser and type the URL – Localhost:9869
8. In the Open Nebula login page use the following credentials
9. Username: oneadmin, Password: opennebula



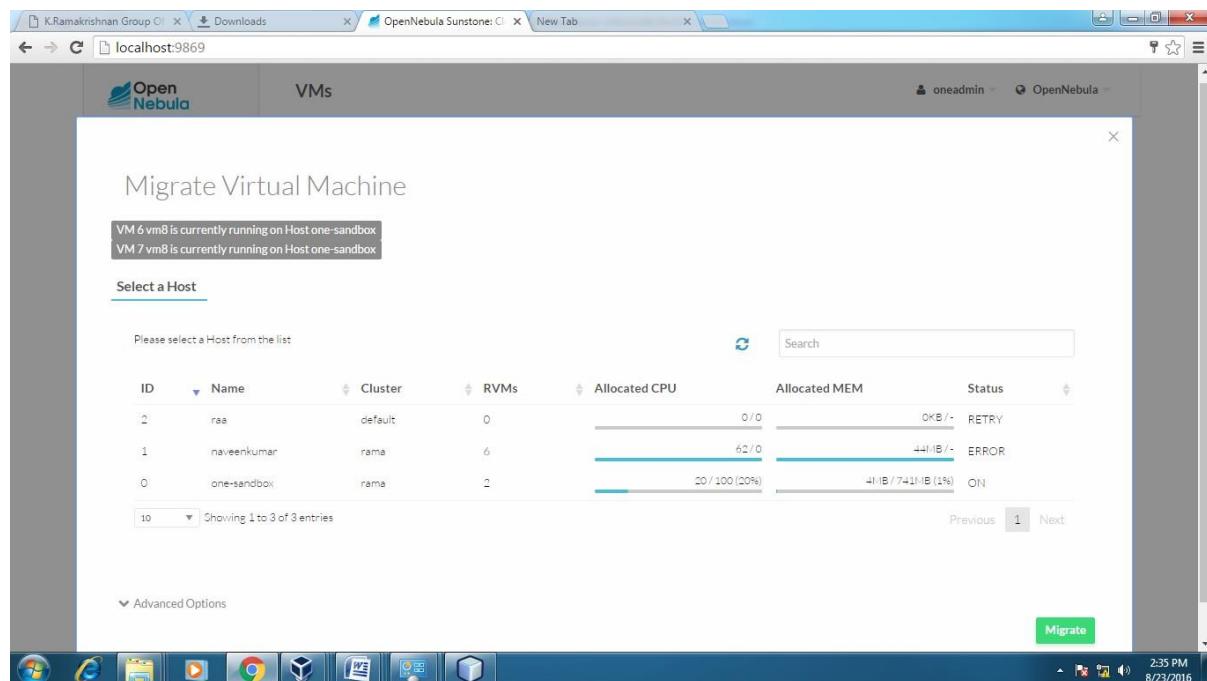
10. Then follow the steps to Create VMs

- a. Click on instances, select VMs then follow the steps to create VM
- b. Expand the + symbol
- c. Select user oneadmin
- d. Then enter the VM Name, No.of instance, and Memory size else use the default allocation.
- e. Then click on create button.
- f. Repeat the steps the a to e for creating more than one VMs.



Then follow the steps for VM Migration

- g. Click on infrastructure
 - h. Select clusters and enter the cluster name
 - i. Then select host tab, and select all host
 - j. Then select Virtual networkss tab, and select all VNets
 - k. Then select datastores tab, and select all datastores
 - l. And then choose host under infrastructure tab
 - m. Click on + symbol to add new host, name the host then click on create.
11. On instances, select VMs to migrate then follow the stpes
- a. Click on 8th icon, the drop-down list display
 - b. Select migrate on that, the popup window display
 - c. On that select the target host to migrate then click on migrate.



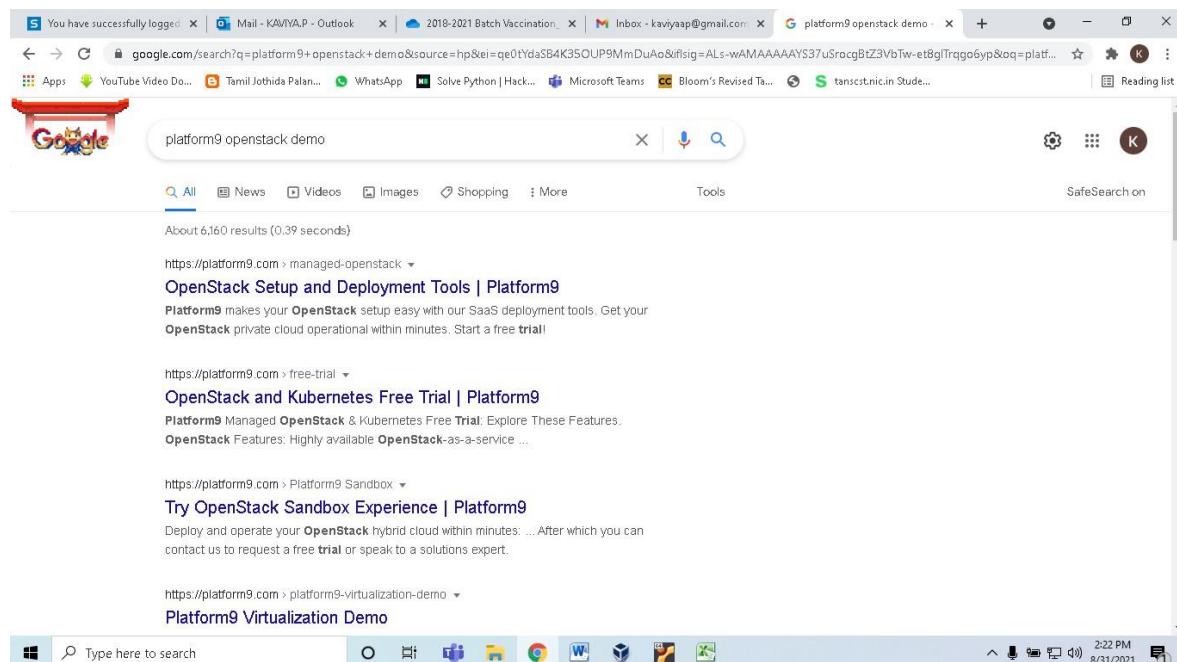
Result:

Ex. No. 07	Find a procedure to launch virtual machine using trystack (Online)
Date:	Openstack Demo Version

Aim:

Procedure:

1. Create an account in Platform9



The screenshot shows the Platform9 Managed OpenStack landing page. At the top, there's a navigation bar with links for Products, Solutions, Pricing, Resources, Docs, About, Login, Contact Us, and Get Started. Below the navigation is a large banner with the title "Managed OpenStack" and the subtitle "Deploy and operate OpenStack Hybrid Cloud with a 100% SaaS based service". Two buttons, "Explore Managed Openstack" and "Compare OpenStack", are visible. The background of the page is dark.

How it works



Click “Try SandBox”

The screenshot shows the same Platform9 Managed OpenStack landing page as before, but with the "Try Sandbox" button highlighted in blue. Below the main banner, there's a section titled "The power and flexibility of your own OpenStack cloud, delivered as a service" with a brief description of the service. At the bottom of the page, there's a "Recommended Resources" section. The taskbar at the bottom of the screen is identical to the one shown above.

Explore Managed OpenStack → Give all the details → Click “Submit”

Deploy and operate your OpenStack hybrid cloud within minutes:

- 100% SaaS-based managed service.
- No hardware required.
- Create virtual machines, set available customizations.
- Create tenant networks (for VM to VM communication).
- View available images, volumes and performance metrics.
- Manage all your infrastructure with a single pane of glass.
- See how you can manage Kubernetes clusters alongside OpenStack VMs for a unified cloud experience across any workload.

Business Email*

kavyait@kamarajengg.edu.in

First name*

KAVIYA

Last name*

P

Job Title*

Assistant Professor

Submit

Thank you!

Begin your Kubernetes or Openstack journey by following the steps below.

Welcome Chris

Clusters: 1 (1 healthy, 0 unhealthy, 0 pending) | Nodes: 1 (1 healthy, 0 unhealthy, 0 pending) | Pods: 22 (22 healthy, 0 unhealthy, 0 pending) | Cloud Accounts: 0 (0 active, 0 inactive)

Users: 2 (1 active, 1 inactive) | Tenants: 0 (0 active, 0 inactive) | Deployments: 0 (0 active, 0 inactive) | Services: 0 (0 active, 0 inactive)

Take Platform9 for a spin!

Platform9 Managed Kubernetes

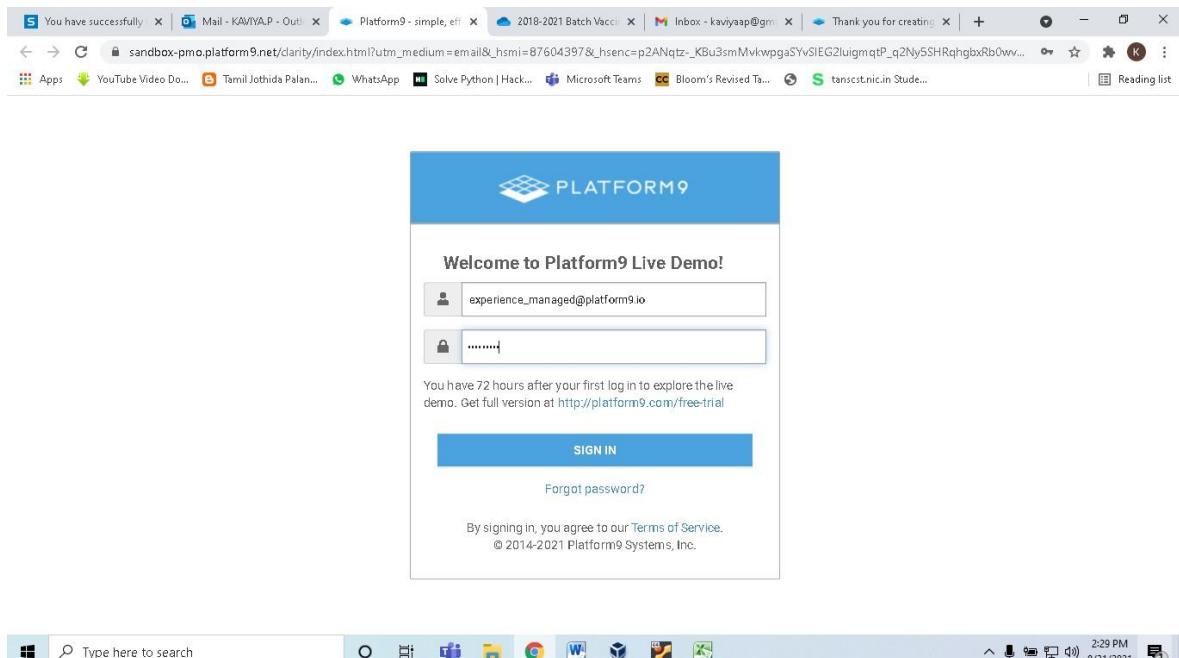
Watch later Share

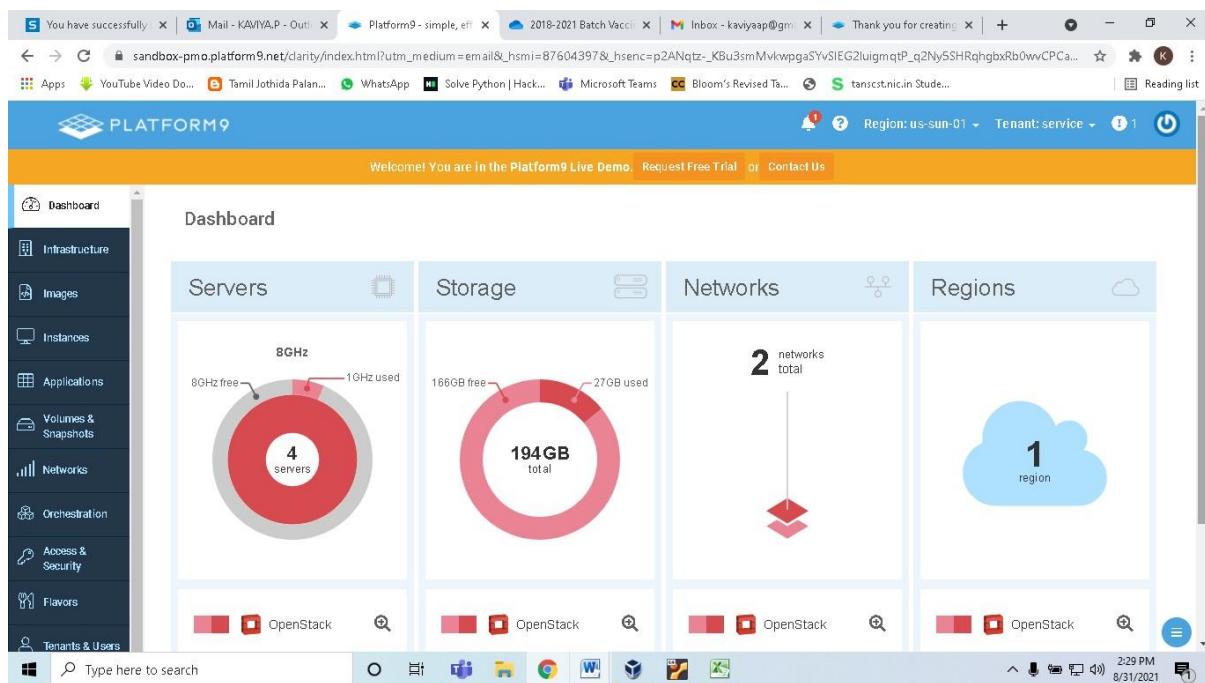
WRITE ONCE

Check the registered mail to get “User Name” and “Password”

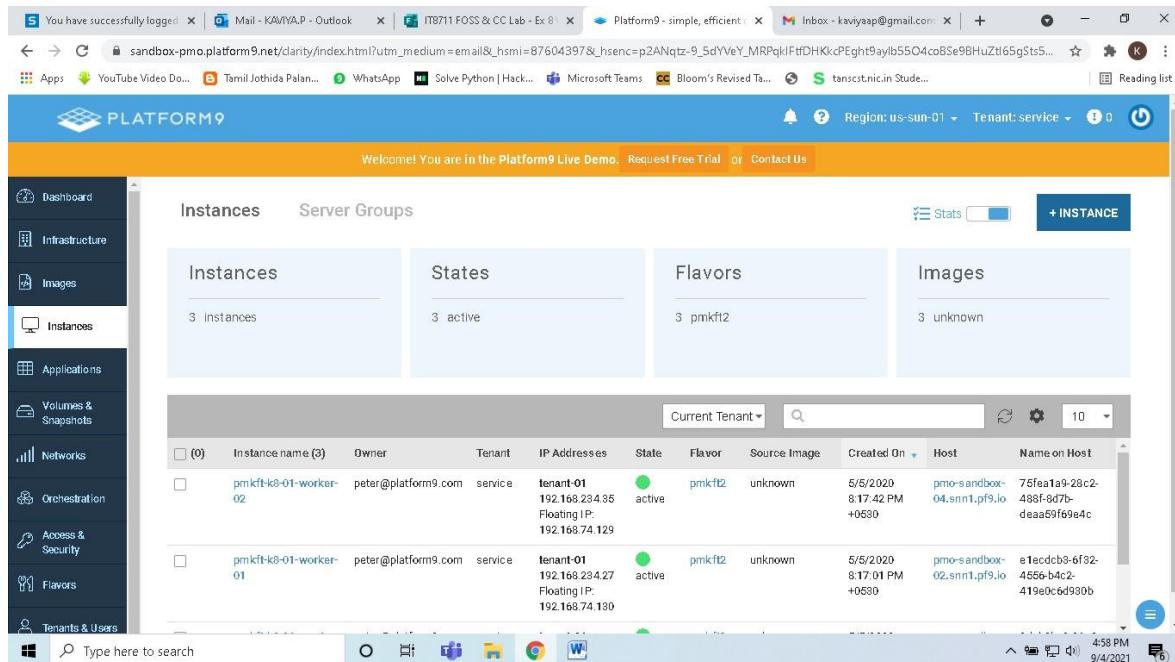
To Login “Platform9”.

Click URL → Give “User Name” and “Password”





1.6 Instances



2. Instance Creation

Move to Instances → Click “+INSTANCE”

The screenshot shows the Platform9 Instances management interface. On the left, a sidebar lists various navigation options: Dashboard, Infrastructure, Images, Instances (which is selected), Applications, Volumes & Snapshots, Networks, Orchestration, Access & Security, and Flavors. The main content area has tabs for Instances, Server Groups, Stats, and + INSTANCE. Below these are four cards: Instances (3 instances), States (3 active), Flavors (pmkft2), and Images (3 unknown). A large table lists the details of the two existing instances:

Instance name	Owner	Tenant	IP Addresses	State	Flavor	Source Image	Created On	Host	Name on Host
pmkft-k8-01-worker-02	peter@platform9.com	service	tenant-01 192.168.234.35 Floating IP: 192.168.74.129	active	pmkft2	unknown	5/5/2020 8:17:42 PM +0530	pmo-sandbox-04.snn1.pf9.io	75fe1a9-28c2-488f-8d7b-daaa59f69e4c
pmkft-k8-01-worker-01	peter@platform9.com	service	tenant-01 192.168.234.27 Floating IP: 192.168.74.130	active	pmkft2	unknown	5/5/2020 8:17:01 PM +0530	pmo-sandbox-02.snn1.pf9.io	a1ecdcbb-6f32-4556-b4c2-419e0cd980b

Source: Boot Instance from → Image, Image Selection → Choose an “OS image” → Click “Next”

Flavor:

Availability Zone → Choose “Nova”

Flavor Selection → Choose “m1.small” → Click “Next”

Network Selection: Select “tenant-01” → Click “Next”

Configure:

1. Instance Name (Eg: Test)
2. SSH Key (Optional)
3. Server Group (Optional)

Enter the details → Click “Next”

(Optional) Customize → Click “Finish”

New VM Instance Review → Click “Create Instance”

Instance is Created

3. To Create new SSH Key

Move “Access & Security” → Click “+ NEW SSH KEY” → Give
“Keypair name” & “Public key content”
To generate SSH public key, In Linux (CentOS), Open

TerminalSwitch to “root” user [Password: newitlab]

To install SSH → #yum -y install openssh-server openssh-clients

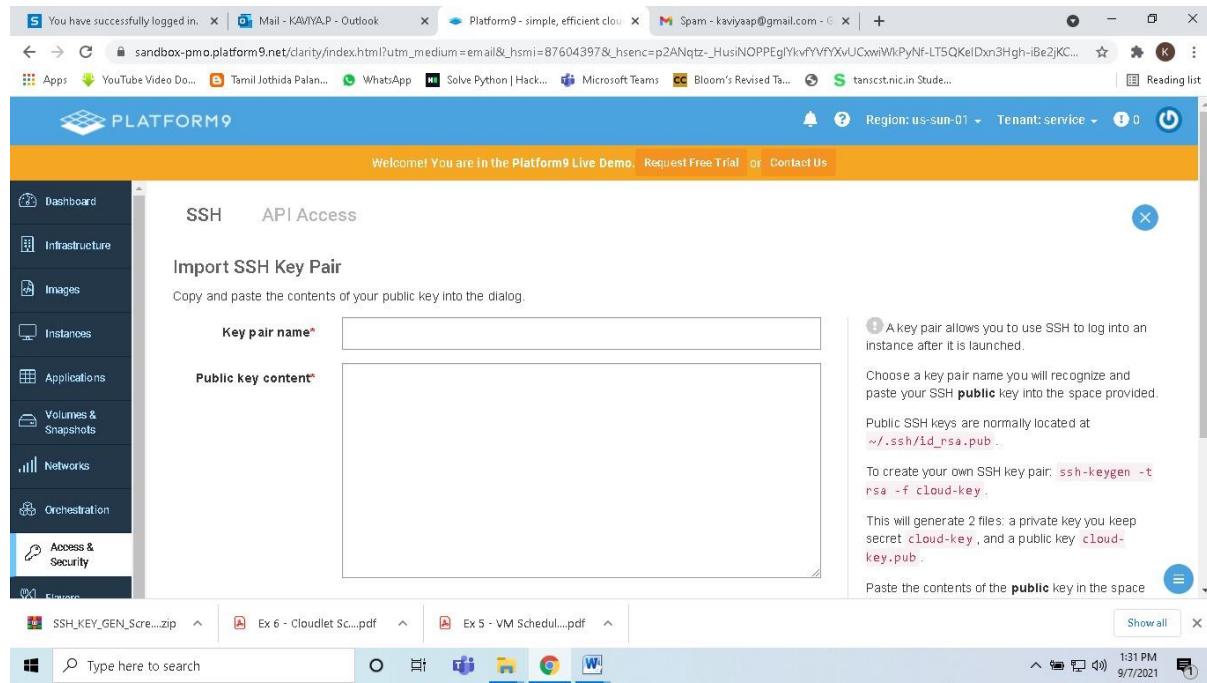
Set the service to start with the machine → #chkconfig sshd on

To generate SSH Key → #ssh-keygen -t rsa -f cloud-key

To view “Private Key” of cloud-key

To view “Public Key” of cloud-key

4. Copy “cloud-key.pub” key → Paste in platform9 “Import SSH Key Pair → Public key content” → Click “IMPORT”



Result:

Ex. No. 08	Install Hadoop single node cluster and run simple applications like word count.
Date:	

Aim:

Procedures:

/home/dinesh/Downloads/hadoop-2.7.0

- a. To format the HDFS:

bin/hadoop namenode -format

- b. To start NameNode daemon and DataNode daemon:

sbin/start-dfs.sh

- c. Start ResourceManager daemon and NodeManager daemon:

sbin/start-yarn.sh

- d. Make the HDFS directories required to execute MapReduce jobs:

bin/hdfs dfs -mkdir /Annie

- e. Copy the input files into the distributed filesystem:

bin/hdfs dfs -put <input-path>/* /Annie

f. Execute the wordcount jar file as follows

```
bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0.jar wordcount /Cloud/file1.txt /op1
```

Output:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	itadmin	supergroup	0 B	5/28/2016, 2:07:16 PM	1	128 MB	_SUCCESS
-rw-r--r--	itadmin	supergroup	142 B	5/28/2016, 2:07:16 PM	1	128 MB	part-r-00000

Hadoop, 2014.

```
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
```

2. Steps to be followed to Shut down the Hadoop
 - sbin/stop-dfs.sh → to stop the nodes
 - sbin/stop-yarn.sh → to stop the node managers

Result:

CONTENT BEYOND SYLLABUS

Ex. No. 01	
Date:	Develop a new Web Service for Calculator

Aim

To develop a new Web service for Calculator applications.

Procedure:

When you start Globus toolkit container, there will be number of services starts up. The service for this task will be a simple Math service that can perform basic arithmetic for a client.

The Math service will access a resource with two properties:

An integer value that can be operated upon by the service

A string values that holds string describing the last operation

The service itself will have three remotely accessible operations that operate upon value:

add, that adds a to the resource property value.

subtract that subtracts a from the resource property value.

getValueRP that returns the current value of value.

Usually, the best way for any programming task is to begin with an overall description of what you want the code to do, which in this case is the service interface. The service interface describes how what the service provides in terms of names of operations, their arguments and return values. A Java interface for our service is:

```
public interface Math { public void add(int a); public void subtract(int a); public int  
getValueRP();  
}
```

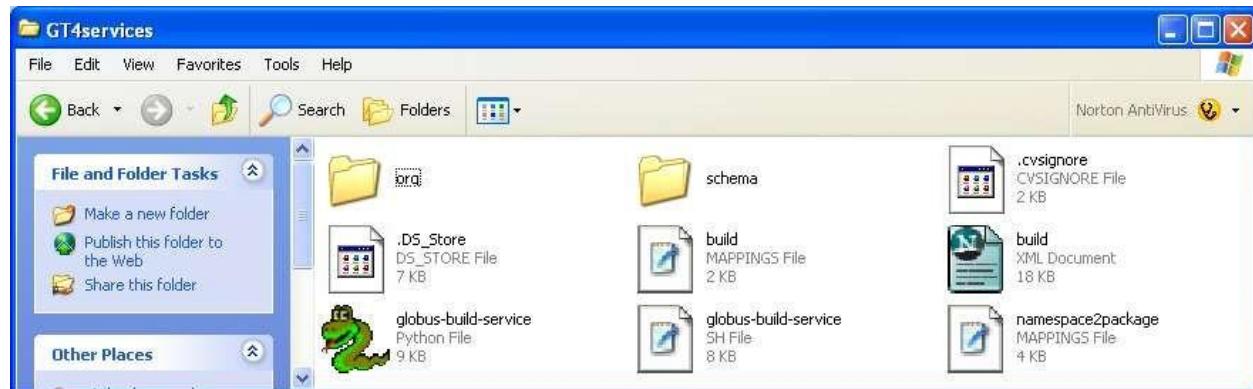
It is possible to start with this interface and create the necessary WSDL file using the standard Web service tool called Java2WSDL. However, the WSDL file for GT 4 has to include details of resource properties that are not given explicitly in the interface above. Hence, we will provide the WSDL file.

Step 1 Getting the Files

All the required files are provided and comes directly from [1]. The MathService source code files can be found from <http://www.gt4book.com> (<http://www.gt4book.com/downloads/gt4book-examples.tar.gz>)

A Windows zip compressed version can be found at <http://www.cs.uncc.edu/~abw/ITCS4146S07/gt4book-examples.zip>.

Download and uncompress the file into a directory called GT4services. Everything is included (the java source WSDL and deployment files, etc.):



WSDL service interface description file -- The WSDL service interface description file is provided within the GT4services folder at: GT4Services\schema\examples\MathService_instance\Math.wsdl

This file, and discussion of its contents, can be found in Appendix A. Later on we will need to modify this file, but first we will use the existing contents that describe the Math service above. Service code in Java -- For this assignment, both the code for service operations and for the resource properties are put in the same class for convenience. More complex services and resources would be defined in separate classes. The Java code for the service and its resource properties is located within the GT4services folder at:

GT4services\org\globus\examples\services\core\first\impl\MathService.java.

Deployment Descriptor -- The deployment descriptor gives several different important sets of information about the service once it is deployed. It is located within the GT4services folder at:

GT4services\org\globus\examples\services\core\first\deploy-server.wsdd.

Step 2 – Building the Math Service

It is now necessary to package all the required files into a GAR (Grid Archive) file. The build tool ant from the Apache Software Foundation is used to achieve this as shown overleaf: Generating a GAR file with Ant (from <http://gdp.globus.org/gt4-tutorial/multiplehtml/ch03s04.html>)

Ant is similar in concept to the Unix make tool but a java tool and XML based.

Build scripts are provided by Globus 4 to use the ant build file. The windows version

of the build script for MathService is the Python file called globus-build-service.py, which held in the GT4services directory. The build script takes one argument, the name of your service that you want to deploy. To keep with the naming convention in [1], this service will be called first. In the Client Window, run the build script from the GT4services directory with:

`globus-build-service.py first`

The output should look similar to the following:

Buildfile: build.xml

.

.

.

.

.

BUILD SUCCESSFUL

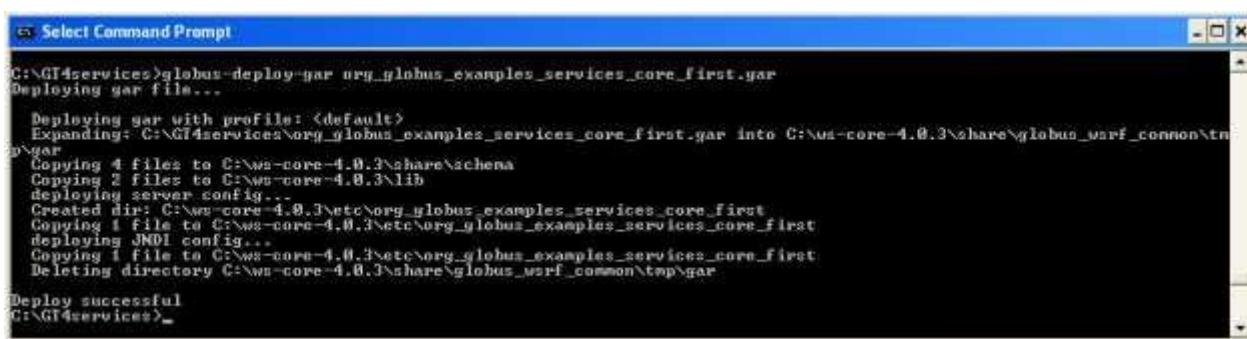
Total time: 8 seconds

During the build process, a new directory is created in your GT4Services directory that is named build. All of your stubs and class files that were generated will be in that directory and its subdirectories. More importantly, there is a GAR (Grid Archive) file called org_globus_examples_services_core_first.gar. The GAR file is the package that contains every file that is needed to successfully deploy your Math Service into the Globus container. The files contained in the GAR file are the Java class files, WSDL, compiled stubs, and the deployment descriptor.

Step 3 – Deploying the Math Service

If the container is still running in the Container Window, then stop it using Control-C. To deploy the Math Service, you will use a tool provided by the Globus Toolkit called globus-deploy-gar. In the Container Window, issue the command:

`globus-deploy-gar org_globus_examples_services_core_first.gar`



```
C:\gt4services>globus-deploy-gar org_globus_examples_services_core_first.gar
Deploying gar file...
Deploying gar with profile: <default>
Expanding: C:\gt4services\org_globus_examples_services_core_first.gar into C:\ws-core-4.0.3\share\globus_worf_common\tmp\gar
Copying 4 files to C:\ws-core-4.0.3\share\schema
Copying 2 files to C:\ws-core-4.0.3\lib
deploying server config...
Created dir: C:\ws-core-4.0.3\etc\org_globus_examples_services_core_first
Copying 1 file to C:\ws-core-4.0.3\etc\org_globus_examples_services_core_first
deploying JNDI config...
Copying 1 file to C:\ws-core-4.0.3\etc\org_globus_examples_services_core_first
Deleting directory C:\ws-core-4.0.3\share\globus_worf_common\tmp\gar

Deploy successful
C:\gt4services>
```

Successful output of the command is :

The service has now been deployed.

Check service is deployed by starting container from the Container Window: You should see the service called MathService.

Step 4 – Compiling the Client

A client has already been provided to test the Math Service and is located in the GT4Services directory at:

GT4Services\org\globus\examples\clients\MathService_instance\Client.java and contains

```
C:\>globus-start-container -nosec
Starting SOAP server at: http://166.82.130.77:8080/wsrf/services/
With the following services:
[1]: http://166.82.130.77:8080/wsrf/services/AdminService
[2]: http://166.82.130.77:8080/wsrf/services/AuthzCalloutTestService
[3]: http://166.82.130.77:8080/wsrf/services/ContainerRegistryEntryService
[4]: http://166.82.130.77:8080/wsrf/services/ContainerRegistryService
[5]: http://166.82.130.77:8080/wsrf/services/CounterService
[6]: http://166.82.130.77:8080/wsrf/services/ManagementService
[7]: http://166.82.130.77:8080/wsrf/services/NotificationConsumerFactoryService
[8]: http://166.82.130.77:8080/wsrf/services/NotificationConsumerService
[9]: http://166.82.130.77:8080/wsrf/services/NotificationTestService
[10]: http://166.82.130.77:8080/wsrf/services/PersistenceTestSubscriptionManager
[11]: http://166.82.130.77:8080/wsrf/services/SampleAuthzService
[12]: http://166.82.130.77:8080/wsrf/services/SecureCounterService
[13]: http://166.82.130.77:8080/wsrf/services/SecurityTestService
[14]: http://166.82.130.77:8080/wsrf/services/ShutdownService
[15]: http://166.82.130.77:8080/wsrf/services/SubscriptionManagerService
[16]: http://166.82.130.77:8080/wsrf/services/TestAuthzService
[17]: http://166.82.130.77:8080/wsrf/services/TestRPCService
[18]: http://166.82.130.77:8080/wsrf/services/TestService
[19]: http://166.82.130.77:8080/wsrf/services/TestServiceRequest
[20]: http://166.82.130.77:8080/wsrf/services/TestServiceWrongWSDL
[21]: http://166.82.130.77:8080/wsrf/services/Version
[22]: http://166.82.130.77:8080/wsrf/services/WidgetNotificationService
[23]: http://166.82.130.77:8080/wsrf/services/WidgetService
[24]: http://166.82.130.77:8080/wsrf/services/examples/core/first/MathService
[25]: http://166.82.130.77:8080/wsrf/services/gsi/AuthenticationService
```

You should see the service called MathService.

Step 4 – Compiling the Client

A client has already been provided to test the Math Service and is located in the GT4Services directory at:

GT4Services\org\globus\examples\clients\MathService_instance\Client.java

and contains the following code:

```
package org.globus.examples.clients.MathService_instance; import
org.apache.axis.message.addressing.Address;
import org.apache.axis.message.addressing.EndpointReferenceType; import
org.globus.examples.stubs.MathService_instance.MathPortType; import
org.globus.examples.stubs.MathService_instance.GetValueRP; import
org.globus.examples.stubs.MathService_instance.service.MathServiceAddressingL
ocator;
public class Client {
    public static void main(String[] args) { MathServiceAddressingLocator locator =
new MathServiceAddressingLocator()
    try {
        String serviceURI = args[0];
```

```

// Create endpoint reference to service EndpointReferenceType endpoint = new
EndpointReferenceType(); endpoint.setAddress(new Address(serviceURI));
MathPortType math;
// Get PortType
math = locator.getMathPortTypePort(endpoint);
// Perform an addition

math.add(10);
// Perform another addition math.add(5);
// Access value System.out.println("Current value: "
+ math.getValueRP(new GetValueRP()));
// Perform a subtraction math.subtract(5);
// Access value System.out.println("Current value: "
+ math.getValueRP(new GetValueRP()));
} catch (Exception e) { e.printStackTrace();
}
}
}
}

```

When the client is run from the command line, you pass it one argument. The argument is the URL that specifies where the service resides. The client will create the end point reference and incorporate this URL as the address. The end point reference is then used with the getMathPortTypePort method of a MathServiceAddressingLocator object to obtain a reference to the Math interface (portType). Then, we can apply the methods available in the service as though they were local methods. Notice that the call to the service (add and subtract method calls) must be in a “try {} catch(){}” block because a “RemoteException” may be thrown. The code for the “MathServiceAddressingLocator” is created during the build process. (Thus you don’t have to write it!)

Setting the Classpath

To compile the new client, you will need the JAR files from the Globus toolkit in your CLASSPATH. Do this by executing the following command in the Client Window:

```
%GLOBUS_LOCATION%\etc\globus-devel-env.bat
```

You can verify that this sets your CLASSPATH, by executing the command:

```
echo %CLASSPATH%
```

You should see a long list of JAR files.

Running \gt4\etc\globus-devel-env.bat only needs to be done once for each Client Window that you open. It does not need to be done each time you compile.

Compiling Client

Once your CLASSPATH has been set, then you can compile the Client code by typing in the following command:

```
javac -classpath  
build\classes\org\globus\examples\services\core\first\impl\;%CLASSPATH%  
org\globus\examples\clients\MathService_instance\Client.java
```

Step 5 – Start the Container for your Service

Restart the Globus container from the Container Window with:

```
globus-start-container -nosec
```

if the container is not running.

Step 6 – Run the Client

To start the client from your GT4Services directory, do the following in the Client Window, which passes the GSH of the service as an argument:

```
java -classpath  
build\classes\org\globus\examples\services\core\first\impl\;%CLASSPATH%  
org.globus.examples.clients.MathService_instance.Client  
http://localhost:8080/wsrf/services/examples/core/first/MathService
```

which should give the output:

Current value: 15

Current value: 10

Step 7 – Undeploy the Math Service and Kill a Container

Before we can add functionality to the Math Service (Section 5), we must undeploy the service. In the Container Window, kill the container with a Control-C. Then to undeploy the service, type in the following command:

```
globus-undeploy-gar org_globus_examples_services_core_first
```

which should result with the following output:

Undeploying gar... Deleting /.

.

.

Undeploy successful

6 Adding Functionality to the Math Service

In this final task, you are asked to modify the Math service and associated files so the service supports the multiplication operation. To do this task, you will need to modify:

MathService.java) Math.wsdl)

- Service code (
- WSDL file (

The exact changes that are necessary are not given. You are to work them out yourself. You will need to fully understand the contents of service code and WSDL files and then modify them accordingly. Appendix A gives an explanation of the important parts of these files. Keep all file names the same and simply redeploy the service afterwards. You will also need to add a code to the client code (Client.java) to test the modified service to include multiplication.

Result: