GLOBAL
INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Approved by AICTE, Affiliated to Anna University & Accredited by NAAC)
257 / 1, Bangalore – Chennai High Way, Melvisharam & Ranipet -632 509

CCS335-CLOUDCOMPUTING LABORATORY

LIST OF EXPERIMENTS

Ex. No.	Date	Name of The Experiments	Page No.	Marks	Sign.
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 create dusting virtualbox and execute Simple Programs			
3		Install GoogleApp Engine Create hello world app And other simple web applications using Python/java.			
4		Use GAE launcher to launch the web applications			
5		Simulate a cloud scenario using Cloud Sim and run a scheduling algorithm that is not present in Cloud Sim.			
6		Find a procedure to transfer the files from one virtual machine to another virtual machine.			
7		Install Hadoopsing lenode cluster and run simple applications like word count.			
8		Creating and Executing Your First Container Using Docker and Run a container from Docker Hub.			
CONTENT BEYOND SYLLABUS					
9		Find a procedure to launch a virtual machine using TryStack (online OpenStack demo version)			
10		Simulate a cloud scenario using Grid Sim			

EX. No:1 **Install VirtualBox/VMware Workstation**

Aim:

Find procedure to Install VirtualBox/VMware Workstation with different flavours of Linux or windows OS on top of windows7 or 8.

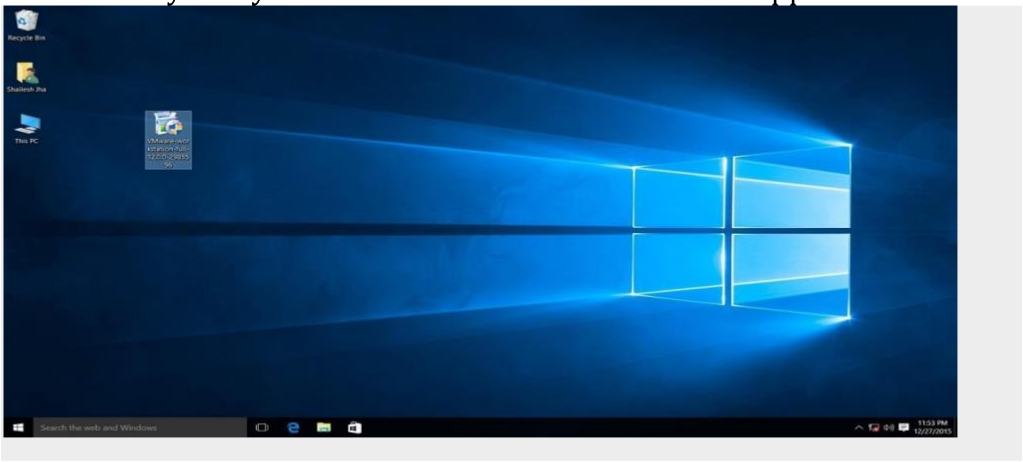
This experiment is to be performed through portal.

PROCEDURE TO INSTALL

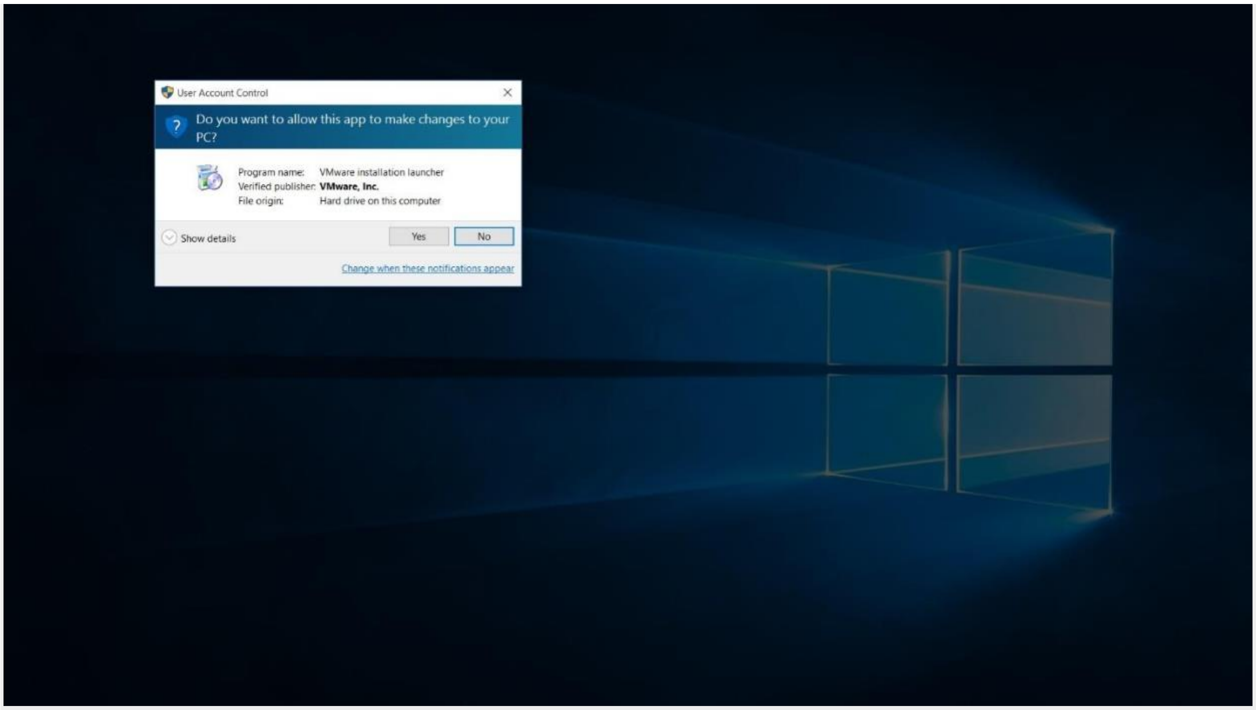
1- Download Link
Link for downloading the software is https://www.vmware.com/products/workstation-pro/workstation-pro-evaluation.html. Download the software for windows. Good thing is that there is no sign up process. Click and download begins. Software is around 541MB.

2- Download the installer file
It should probably be in the download folder by default, if you have not changed the settings in your browser. File name should be something like VMware-workstation-full-15.5.1-15018445.exe. This file name can change depending on the version of the software currently available for download. But for now, till the next version is available, they will all be VMware Workstation 15 Pro.

3- Locate the downloaded installer file
For demonstration purpose, I have placed the downloaded installer on my desktop. Find the installer on your system and double click to launch the application.


Recycle Bin
Shailesh Jha
This PC
VMware-workstation-full-12.0.0-2985596
Search the web and Windows
11:53 PM
12/27/2015

4- User Access Control (UAC) Warning
Now you should see User Access Control (UAC) dialog box. Click yes to continue.



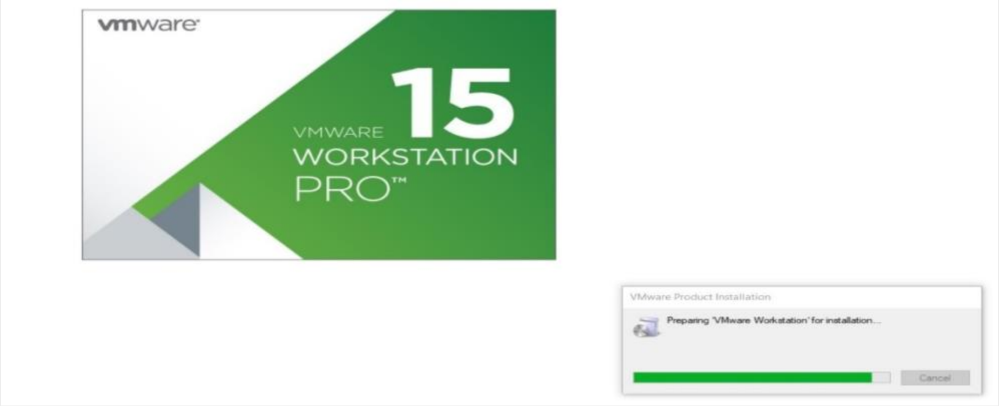
User Account Control

Do you want to allow this app to make changes to your PC?

Program name: VMware installation launcher
Verified publisher: VMware, Inc.
File origin: Hard drive on this computer

VMware Workstation 12 Pro installer windows 10 UAC screenshot

Initial Splash screen will appear. Wait for the process to complete.

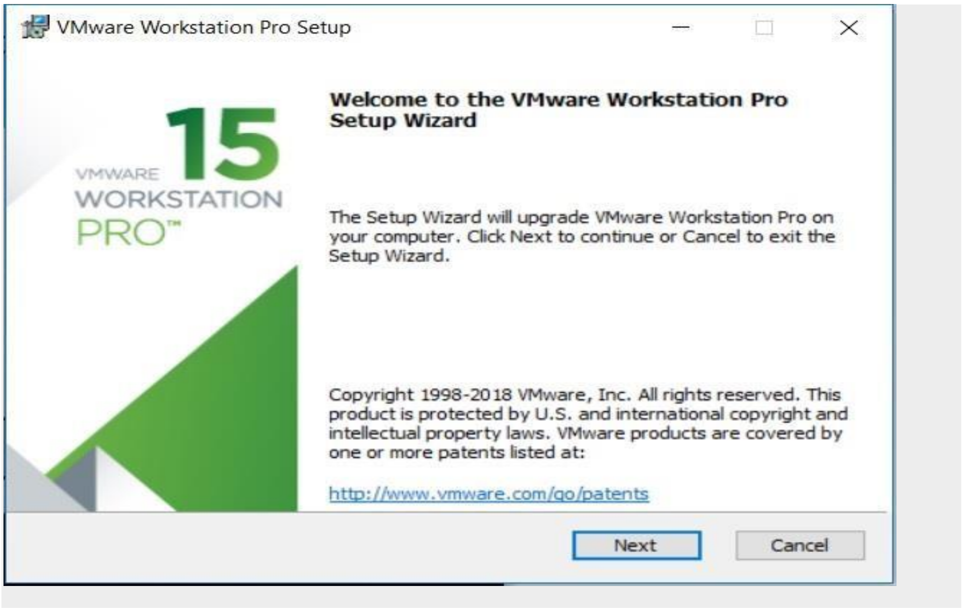


VMware Product Installation

Preparing 'VMware Workstation' for installation...

5- VMware Workstation Setup wizard

Now you will see VMware Workstation setup wizard dialog box. Click next to continue.



VMware Workstation Pro Setup

Welcome to the VMware Workstation Pro Setup Wizard

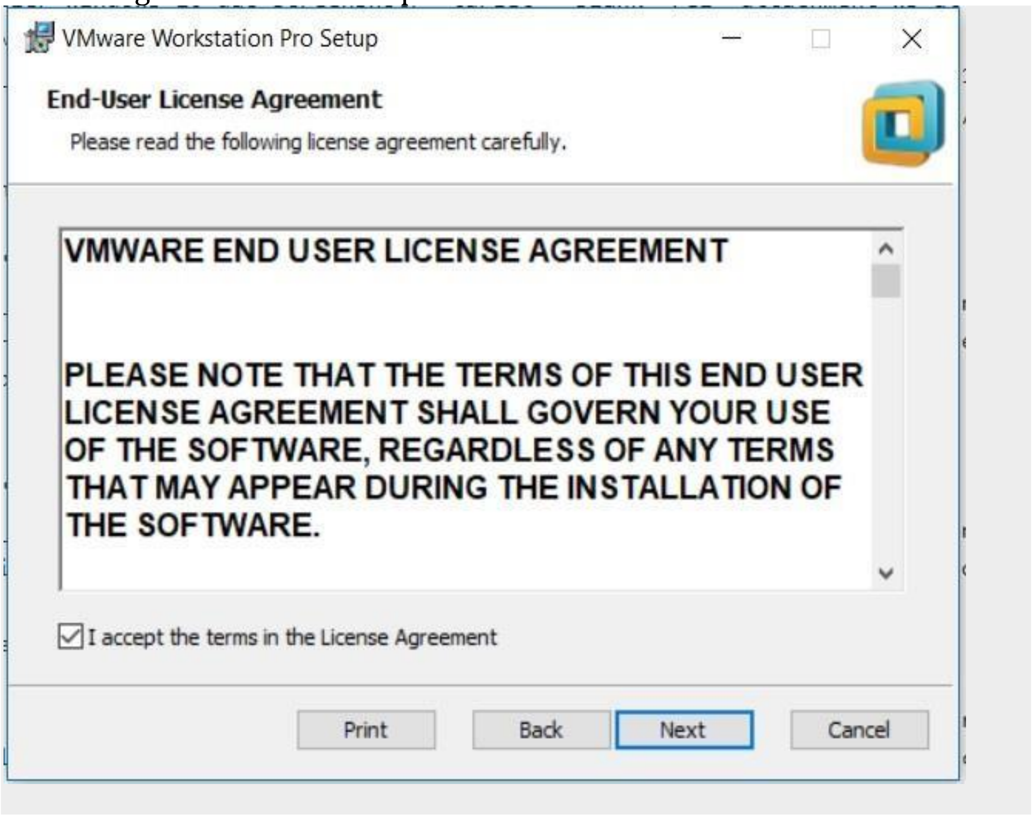
The Setup Wizard will upgrade VMware Workstation Pro on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Copyright 1998-2018 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at:

http://www.vmware.com/go/patents

6- End User Licence Agreement

This time you should see End User Licence Agreement dialog box. Check "I accept the terms in the Licence Agreement" box and press next to continue.



VMware Workstation Pro Setup

End-User License Agreement

Please read the following license agreement carefully.

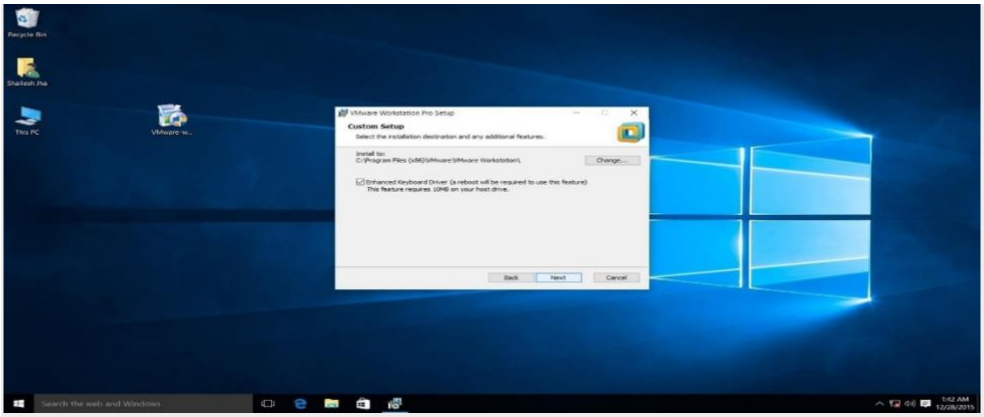
VMWARE END USER LICENSE AGREEMENT

PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.

☑ I accept the terms in the License Agreement

7- Custom Setup options

Select the folder in which you would like to install the application. There is no harm in leaving the defaults as it is. Also select Enhanced Keyboard Driver check box.



VMware Workstation Pro Setup

Custom Setup

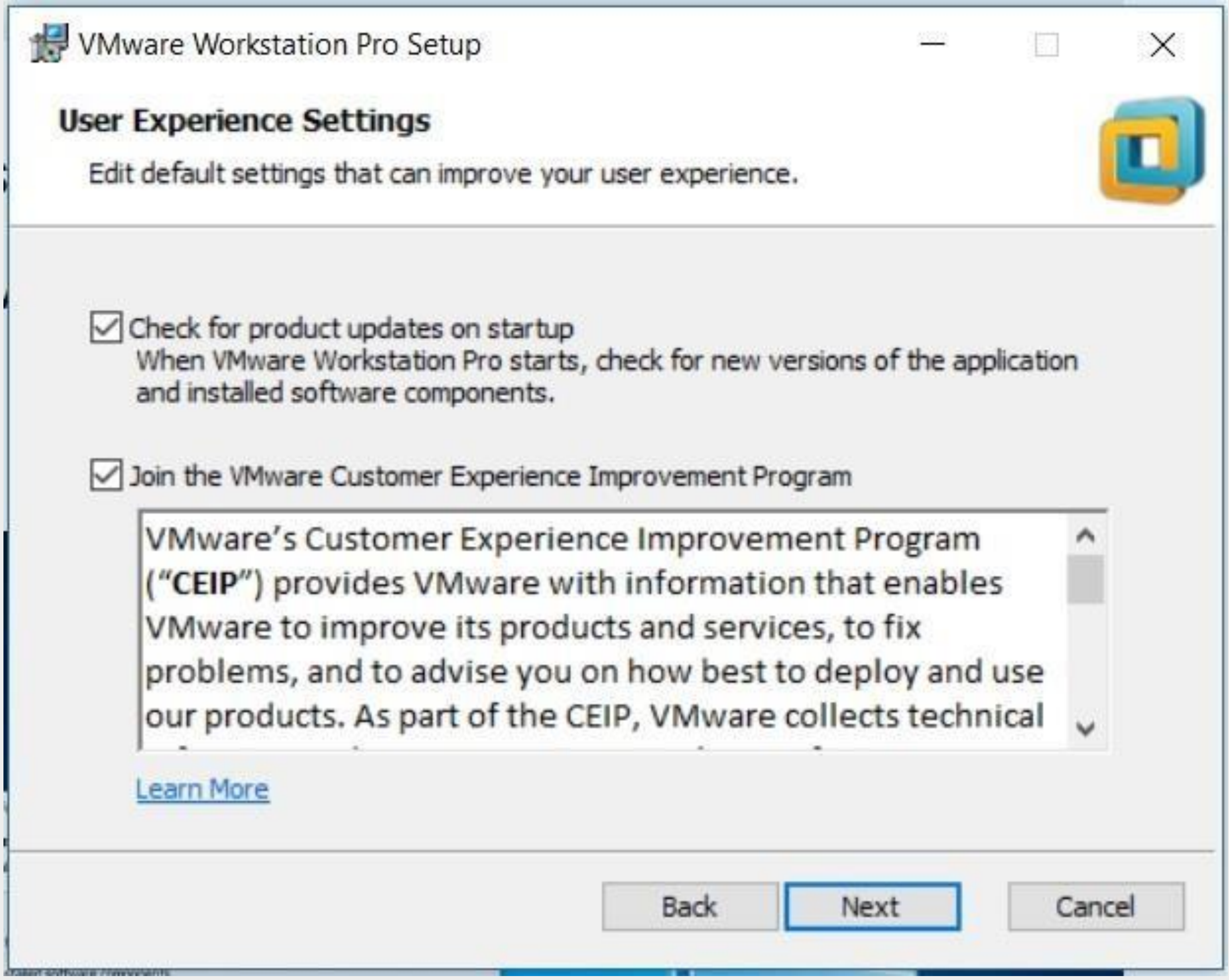
Select the installation destination and any additional features.

Install to:
C:\Program Files (x86)\VMware\VMware Workstation\

☑ Enhanced Keyboard Driver (a reboot will be required to use this feature)
This feature requires 10MB on your host drive.

8- User Experience Settings

Next you are asked to select “Check for Updates” and “Help improve VMware Workstation Pro”. Do as
you wish. I normally leave it to defaults that is unchecked.



VMware Workstation Pro Setup

User Experience Settings

Edit default settings that can improve your user experience.

☑ Check for product updates on startup
When VMware Workstation Pro starts, check for new versions of the application and installed software components.

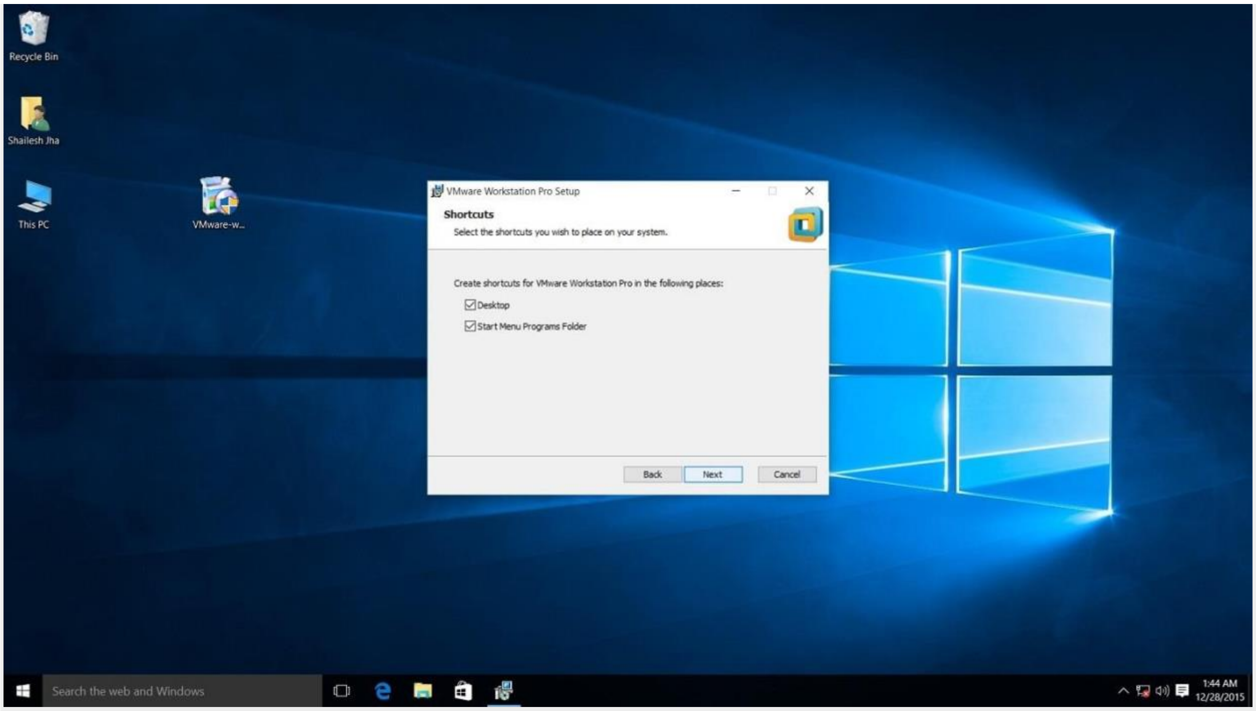
☑ Join the VMware Customer Experience Improvement Program

VMware’s Customer Experience Improvement Program (“**CEIP**”) provides VMware with information that enables VMware to improve its products and services, to fix problems, and to advise you on how best to deploy and use our products. As part of the CEIP, VMware collects technical

VMware Workstation 15 Installation – User Experience Settings

9- Application Shortcuts preference

Next step is to select the place you want the shortcut icons to be placed on your system to launch the application. Please select both the options, desktop and start menu and click next.



VMware Workstation Pro Setup

Shortcuts

Select the shortcuts you wish to place on your system.

Create shortcuts for VMware Workstation Pro in the following places:

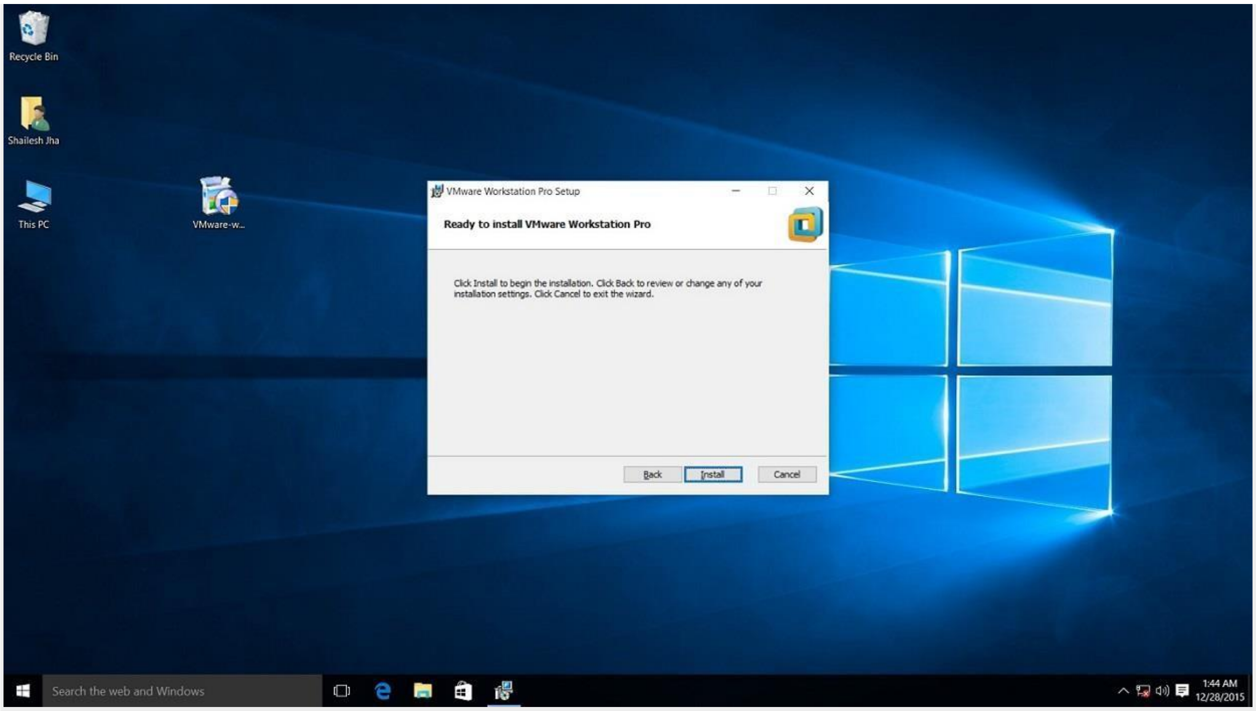
☑ Desktop

☑ Start Menu Programs Folder

VMware workstation 15 pro installation shortcut selection checkbox screenshot.

10- Installation begins

Now you see the begin installation dialog box. Click install to start the installation process.



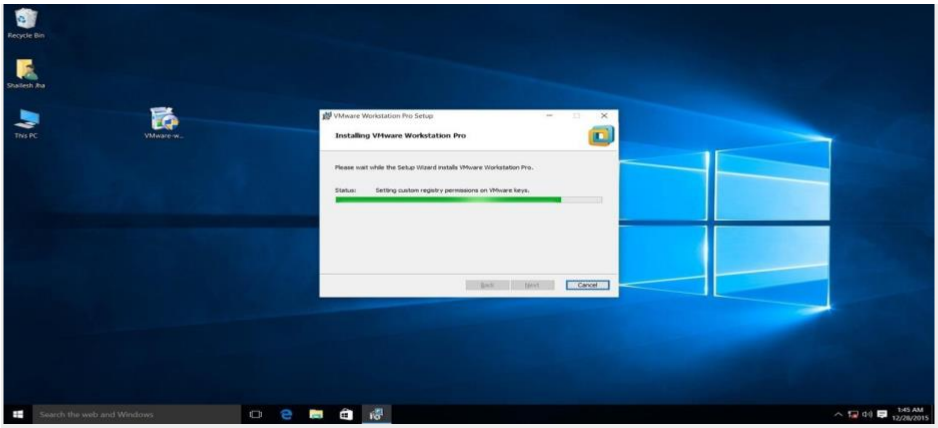
VMware Workstation Pro Setup

Ready to install VMware Workstation Pro

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

Screenshot for VMware Workstation 15 pro installation begin confirmation dialog box on windows 10.

Below screenshot shows Installation in progress. Wait for this to complete.

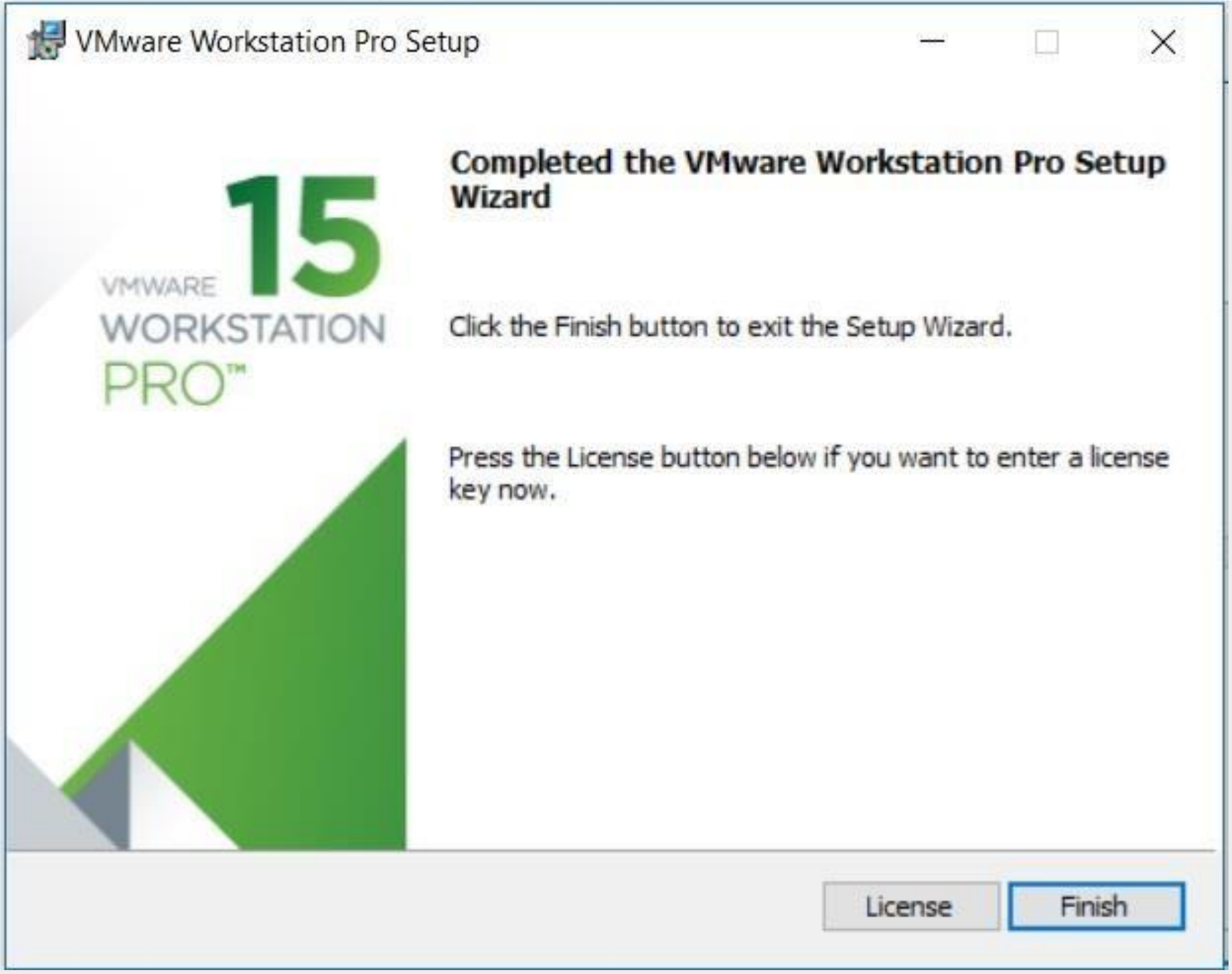


Installing VMware Workstation Pro

Please wait while the Setup Wizard installs VMware Workstation Pro.

Status: Setting custom registry permissions on VMware keys.

At the end you will see installation complete dialog box. Click finish and you are done with the installation process. You may be asked to restart your computer. Click on Yes to restart.



Completed the VMware Workstation Pro Setup Wizard

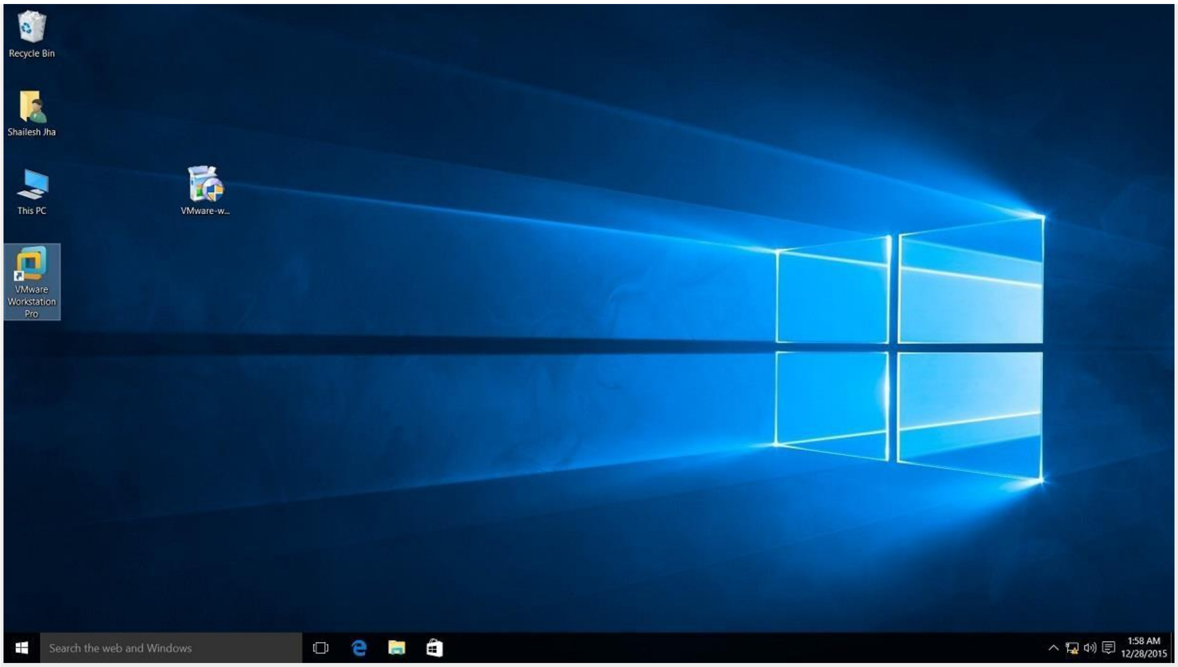
Click the Finish button to exit the Setup Wizard.

Press the License button below if you want to enter a license key now.

VMware Workstation 15 Installation – Installation Complete

11- Launch VMware Workstation

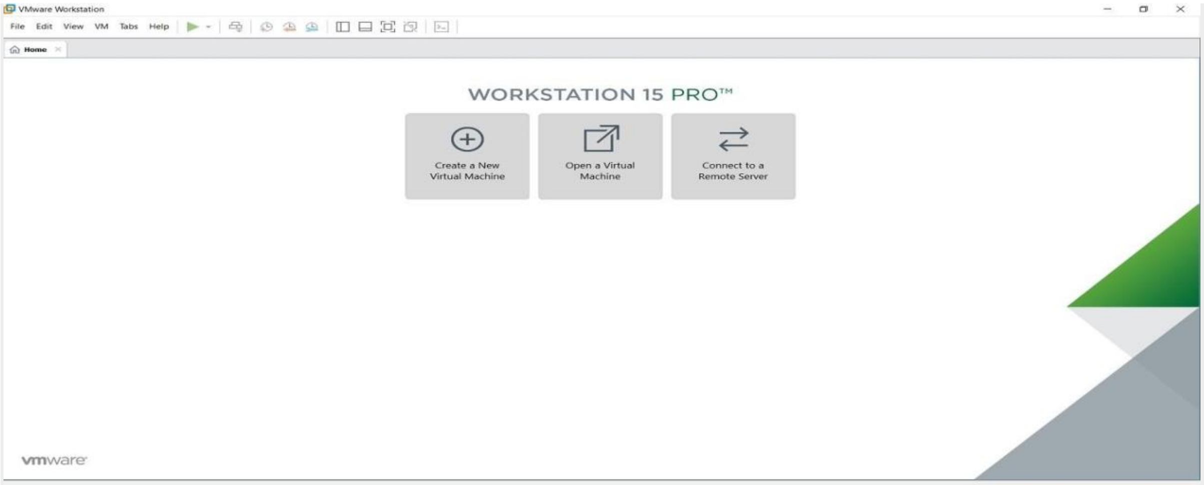
After the installation completes, you should see VMware Workstation icon on the desktop. Double click on it to launch the application.



Screenshot for VMware Workstation 15 Pro icon on windows 10 desktop.

12- Licence Key

If you see the dialog box asking for licence key, click on trial or enter the licence key. Then what you have is the VMware Workstation 15 Pro running on your windows 10 desktop. If don't have the licence key, you will have 30 day serial.



VMware Workstation 15 Pro home screen

13- At some point if you decide to buy

At some point of time if you decide to buy the License key, you can enter the License key by going to **Help->Enter a License** Key

You can enter the 25-characterlicense key in the dialog box shown below and click OK. Now you have the license version of the software.

RESULT:

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

EX NO: 2 Install a C compiler in the virtual machine and execute a sample program

Aim:

To Install a C compiler in the Virtual machine and execute a sample program

Procedure:

Step 1: Installation of GCC compiler

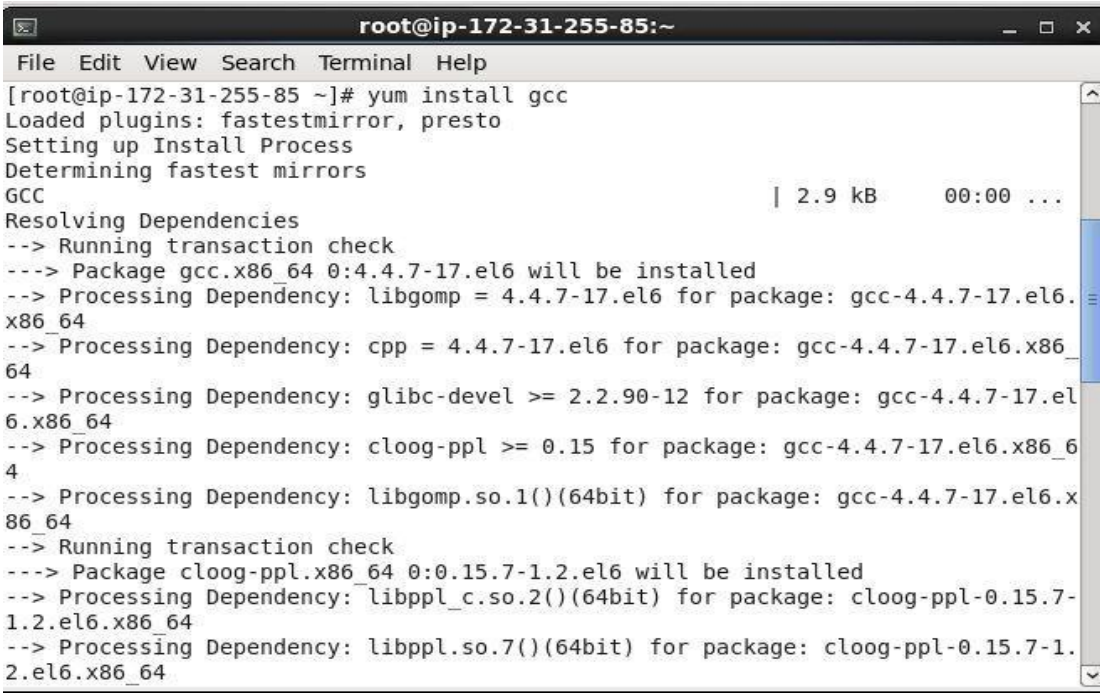
1.Yum install gcc

(GCC will be installed from the local repository. After completion of gcc compiler, ready to run the C program)

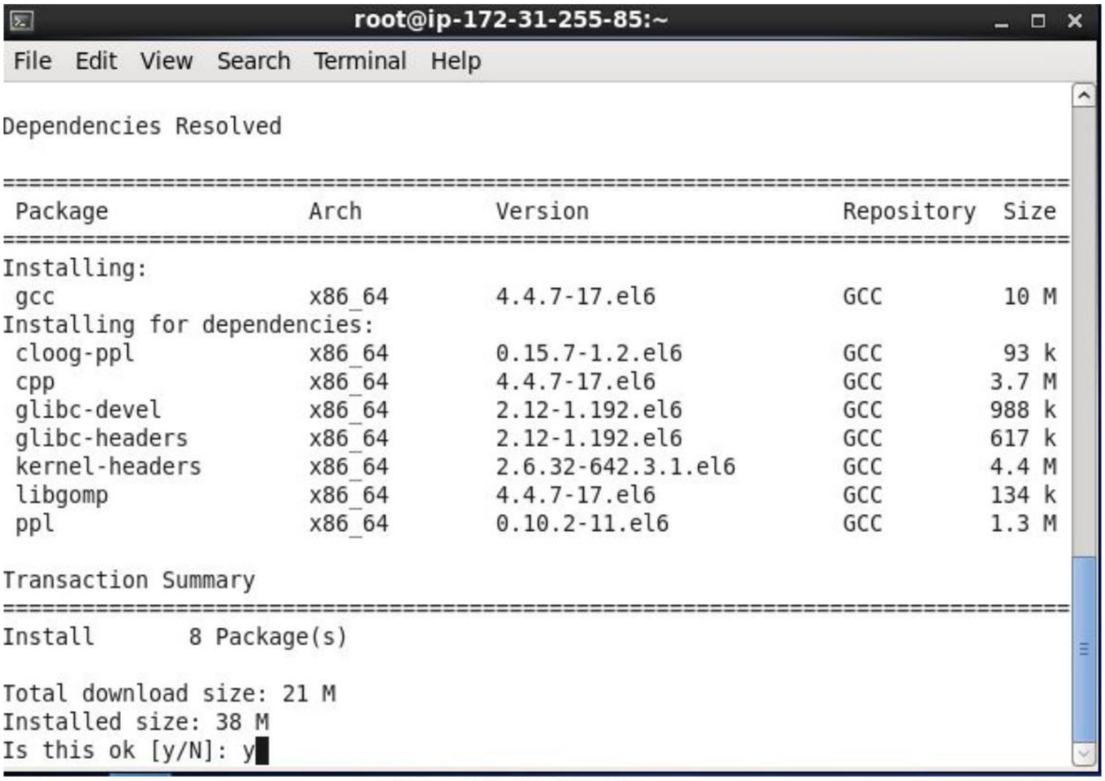
Ø To run the C program
Ø Open the terminal
Ø Execute the c program

2. vi programme.c (for opening the program)

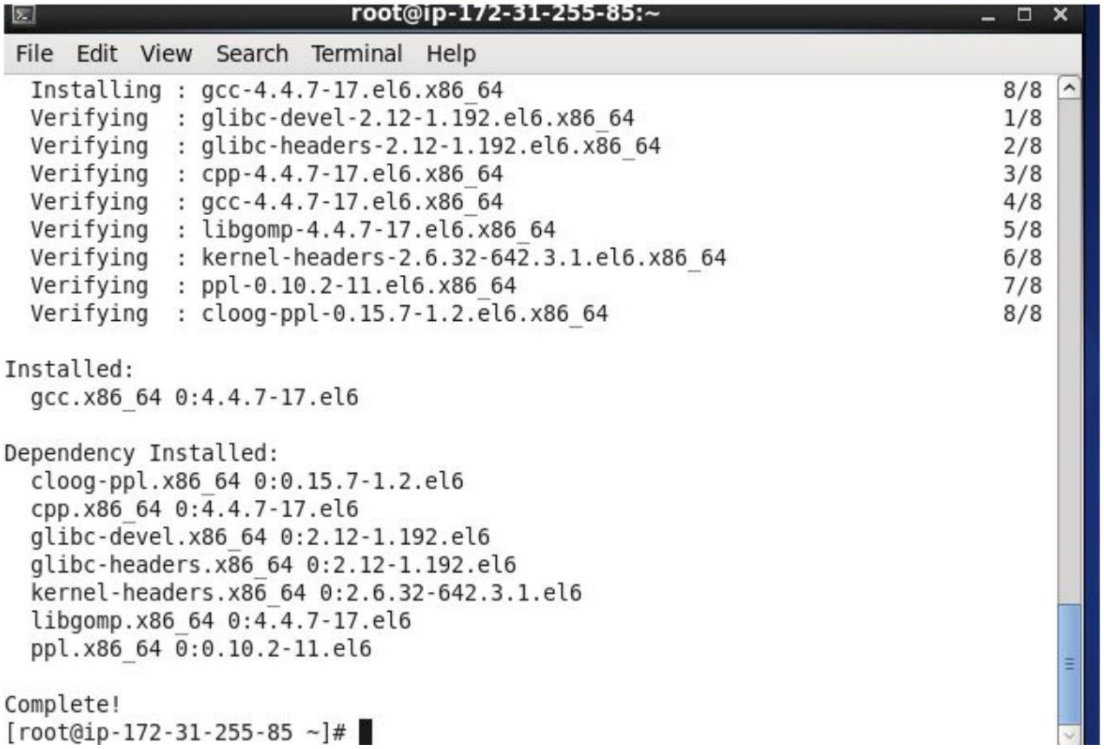
3. Press esc:wq (saving the program



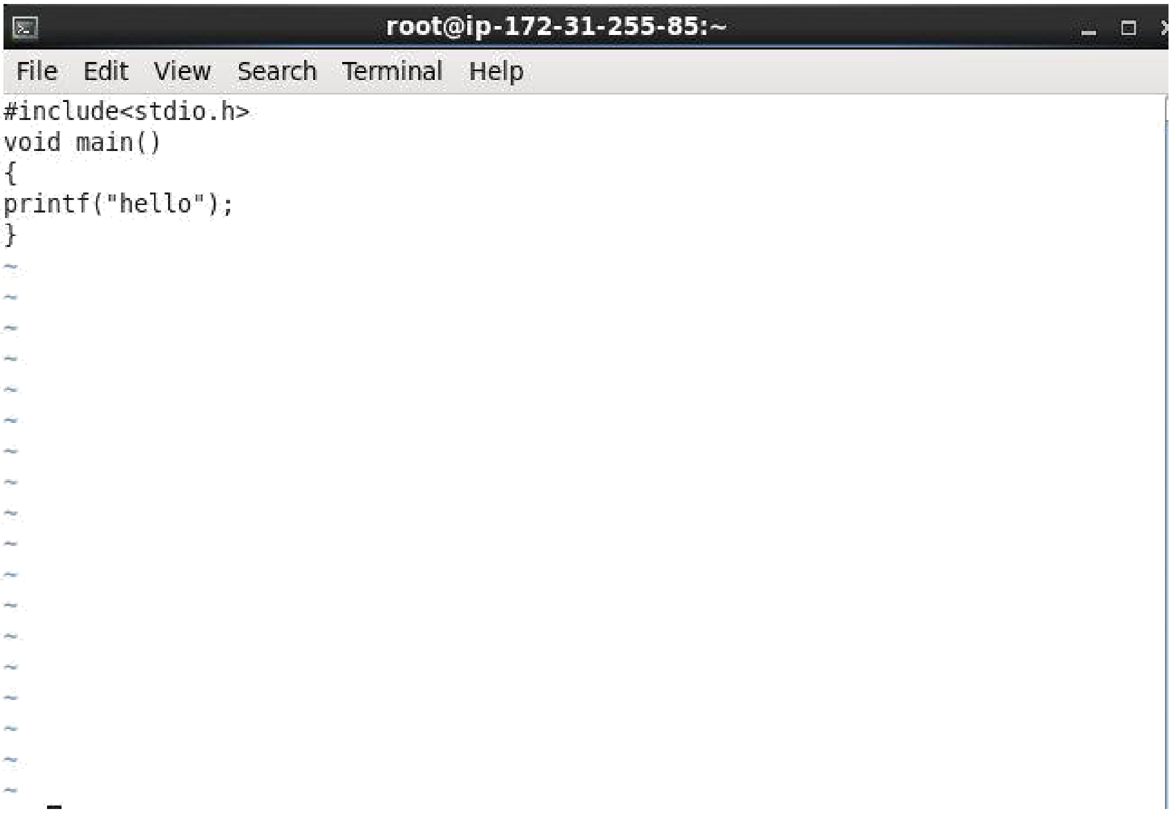
```
[root@ip-172-31-255-85 ~]# yum install gcc
Loaded plugins: fastestmirror, presto
Setting up Install Process
Determining fastest mirrors
GCC                                                   | 2.9 kB     00:00 ...
Resolving Dependencies
--> Running transaction check
---> Package gcc.x86_64 0:4.4.7-17.el6 will be installed
--> Processing Dependency: libgomp = 4.4.7-17.el6 for package: gcc-4.4.7-17.el6.
x86_64
--> Processing Dependency: cpp = 4.4.7-17.el6 for package: gcc-4.4.7-17.el6.x86_
64
--> Processing Dependency: glibc-devel >= 2.2.90-12 for package: gcc-4.4.7-17.el
6.x86_64
--> Processing Dependency: cloog-ppl >= 0.15 for package: gcc-4.4.7-17.el6.x86_6
4
--> Processing Dependency: libgomp.so.1()(64bit) for package: gcc-4.4.7-17.el6.x
86_64
--> Running transaction check
---> Package cloog-ppl.x86_64 0:0.15.7-1.2.el6 will be installed
--> Processing Dependency: libppl_c.so.2()(64bit) for package: cloog-ppl-0.15.7-
1.2.el6.x86_64
--> Processing Dependency: libppl.so.7()(64bit) for package: cloog-ppl-0.15.7-1.
2.el6.x86_64
```



```
Dependencies Resolved

================================================================================
 Package              Arch          Version                    Repository  Size
================================================================================
Installing:
 gcc                  x86_64        4.4.7-17.el6               GCC         10 M
Installing for dependencies:
 cloog-ppl            x86_64        0.15.7-1.2.el6             GCC         93 k
 cpp                  x86_64        4.4.7-17.el6               GCC        3.7 M
 glibc-devel          x86_64        2.12-1.192.el6             GCC        988 k
 glibc-headers        x86_64        2.12-1.192.el6             GCC        617 k
 kernel-headers       x86_64        2.6.32-642.3.1.el6         GCC        4.4 M
 libgomp              x86_64        4.4.7-17.el6               GCC        134 k
 ppl                  x86_64        0.10.2-11.el6              GCC        1.3 M

Transaction Summary
================================================================================
Install       8 Package(s)

Total download size: 21 M
Installed size: 38 M
Is this ok [y/N]: y
```



```
  Installing : gcc-4.4.7-17.el6.x86_64                                      8/8
  Verifying  : glibc-devel-2.12-1.192.el6.x86_64                            1/8
  Verifying  : glibc-headers-2.12-1.192.el6.x86_64                          2/8
  Verifying  : cpp-4.4.7-17.el6.x86_64                                      3/8
  Verifying  : gcc-4.4.7-17.el6.x86_64                                      4/8
  Verifying  : libgomp-4.4.7-17.el6.x86_64                                  5/8
  Verifying  : kernel-headers-2.6.32-642.3.1.el6.x86_64                     6/8
  Verifying  : ppl-0.10.2-11.el6.x86_64                                     7/8
  Verifying  : cloog-ppl-0.15.7-1.2.el6.x86_64                              8/8

Installed:
  gcc.x86_64 0:4.4.7-17.el6

Dependency Installed:
  cloog-ppl.x86_64 0:0.15.7-1.2.el6
  cpp.x86_64 0:4.4.7-17.el6
  glibc-devel.x86_64 0:2.12-1.192.el6
  glibc-headers.x86_64 0:2.12-1.192.el6
  kernel-headers.x86_64 0:2.6.32-642.3.1.el6
  libgomp.x86_64 0:4.4.7-17.el6
  ppl.x86_64 0:0.10.2-11.el6

Complete!
[root@ip-172-31-255-85 ~]#
```



```
#include<stdio.h>
void main()
{
printf("hello");
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

4. cc program name .c (for compiling the program)
5 ./a.out (to see the output)

Result:

Thus, the image (gcc) has been created and the C program has been compiled successfully.

Ex. No. 3 **Install Google App Engine**

Aim:

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

This document describes the installation of the Google App Engine Software Development Kit (SDK)on a Microsoft Windows and running a simple “hello world” application.

The App Engine SDK allows you to run Google App Engine Applications on your local computer. It simulates the run---time environment of the Google App Engine infrastructure.

Pre--Requisites:Python2.5.4

If you don't already have Python2.5.4 installed in your computer, download and Install Python2.5.4 from:

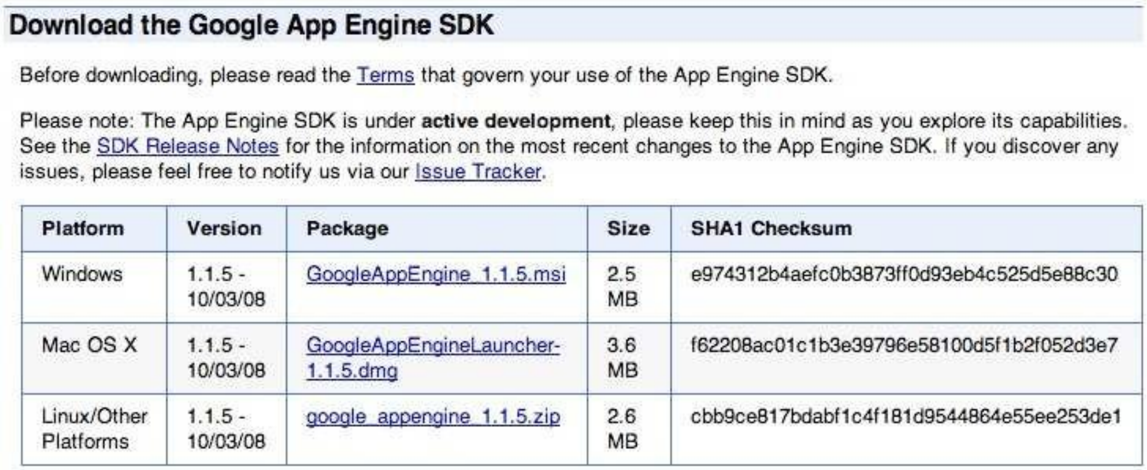
http://www.python.org/download/releases/2.5.4/

Download and Install

You can download the Google App Engine SDK by going to:

http://code.google.com/appengine/downloads.html

and download the appropriate install package.



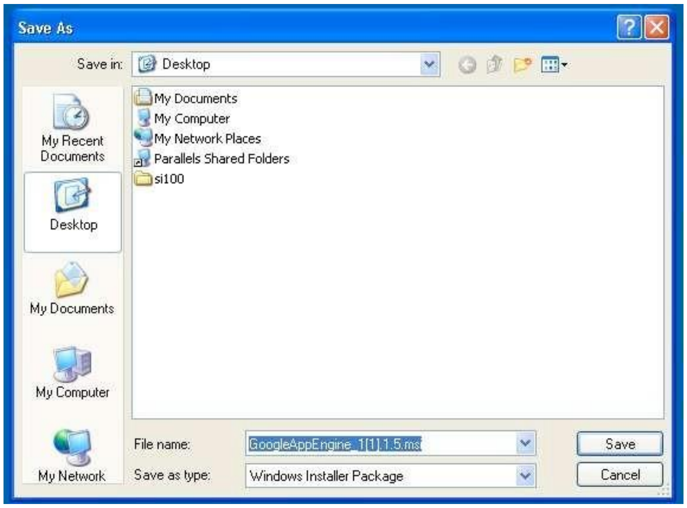
Download the Google App Engine SDK

Before downloading, please read the Terms that govern your use of the App Engine SDK.

Please note: The App Engine SDK is under **active development**, please keep this in mind as you explore its capabilities. See the SDK Release Notes for the information on the most recent changes to the App Engine SDK. If you discover any issues, please feel free to notify us via our Issue Tracker.

Platform	Version	Package	Size	SHA1 Checksum
Windows	1.1.5 - 10/03/08	GoogleAppEngine_1.1.5.msi	2.5 MB	e974312b4aefc0b3873ff0d93eb4c525d5e88c30
Mac OS X	1.1.5 - 10/03/08	GoogleAppEngineLauncher-1.1.5.dmg	3.6 MB	f62208ac01c1b3e39796e58100d5f1b2f052d3e7
Linux/Other Platforms	1.1.5 - 10/03/08	google_appengine_1.1.5.zip	2.6 MB	cbb9ce817bdabf1c4f181d9544864e55ee253de1

Download the Windows installer–the simplest thing is to download it to your Desktop or another folder that you remember.



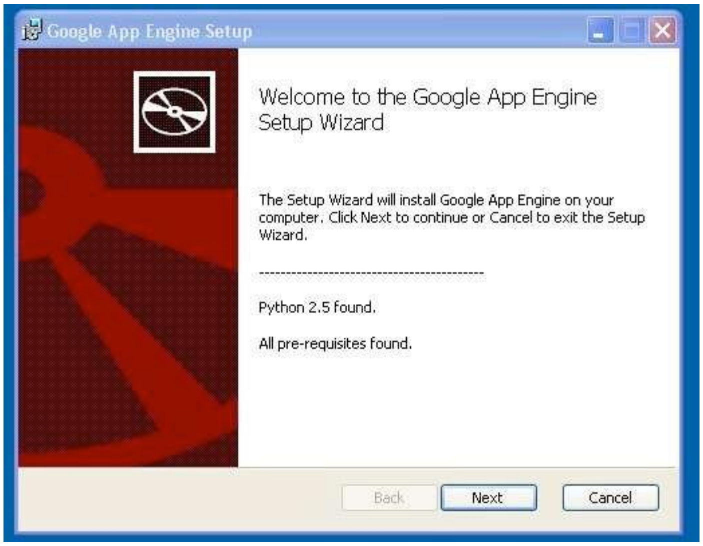
Save As

Save in: Desktop

File name: GoogleAppEngine_1[1].1.5.msi

Save as type: Windows Installer Package

Double Click on the **Google Application Engine** installer.



Google App Engine Setup

Welcome to the Google App Engine Setup Wizard

The Setup Wizard will install Google App Engine on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Python 2.5 found.

All pre-requisites found.

Click through the installation wizard, and it should install the App Engine. If you do not have Python2.5,it will install Python2.5 as well.

Once the install is complete you can discard the downloaded installer

Making your First Application

1. Now you need to create a simple application. We could use the“+”option to have the launcher make us an application–but instead we will do it by hand to get a better sense of what is going on.

Make a folder for your Google App Engine applications. I am going to make the Folder on my Desktop called “**apps**”–the path to this folder is:

C:\Documents and Settings\csev\Desktop\apps

And then make a sub--folder in with in **apps** called “**ae--01--trivial**”—the path to this folder would be:

C:\DocumentsandSettings\csev\Desktop\apps\ae--01--trivial

Using a text editor such as J Edit (www.jedit.org), create a file called **app.yaml** in the **ae--01--trivial** folder with the following contents:

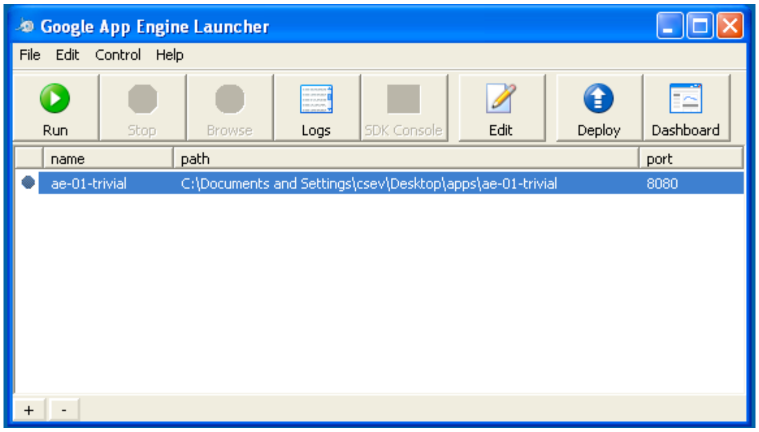
```
application: ae-01-trivial
version: 1
run time: python
 api _version: 1

handlers:
- url: /.*
script: index.py
```

Note: Please do not copy and paste these lines into your text editor–you might end up with strange characters–simply type the min to your editor.

Then create a file in the **ae--01--trivial** folder called **index.py** with three lines init:

```
Print 'Content-Type:
text/plain' print '  '
print 'Hello there Chuck'
```

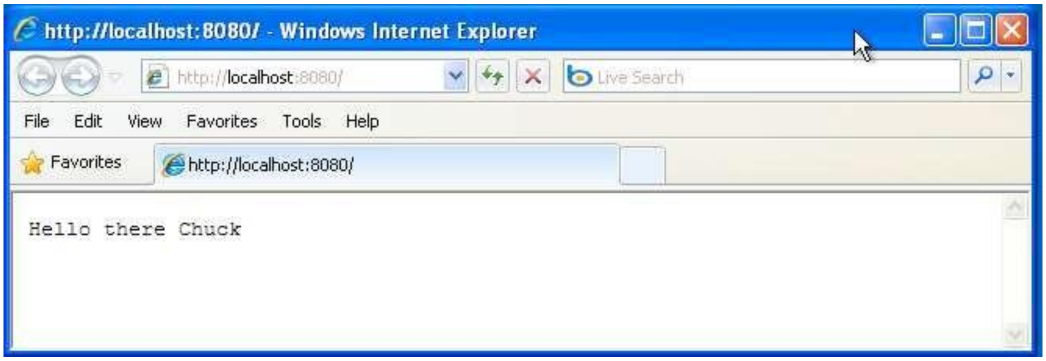
Then start the **Google App Engine Launcher** program that can be found under **Application** .Use the **File-->Add Existing Application** command and navigate into the **apps** directory and select the **ae--01--trivial** folder. Once you have added the application, select it so that you can control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



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

Once you have selected your application and press **Run**. After a few moments your application will start and the launcher will show a little green icon next to your application. Then press **Browse** to open a browser pointing at your application which is running at **http://localhost:8080/**

Paste **http://localhost:8080** into your browser and you should see your application as follows:



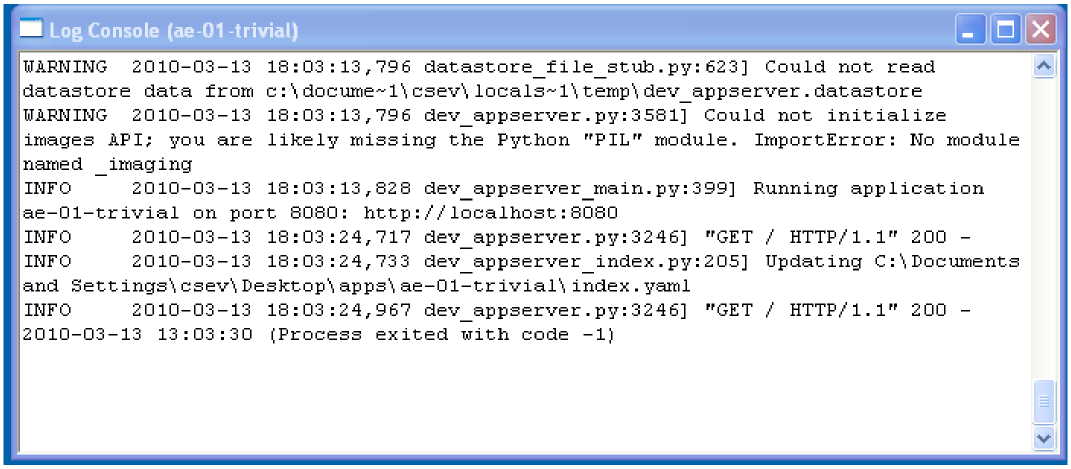
http://localhost:8080/ - Windows Internet Explorer
http://localhost:8080/
Live Search
File Edit View Favorites Tools Help
Favorites
http://localhost:8080/

```
Hello there Chuck
```

Just for fun, edit the **index.py** to change the name “Chuck” to your own name and press Refresh in the browser to verify your updates.

Watching the Log

You can watch the internal log of the actions that the web server is performing when you are interacting with your application in the browser .Select your application in the Launcher and press the **Logs** button to bring up a log window:

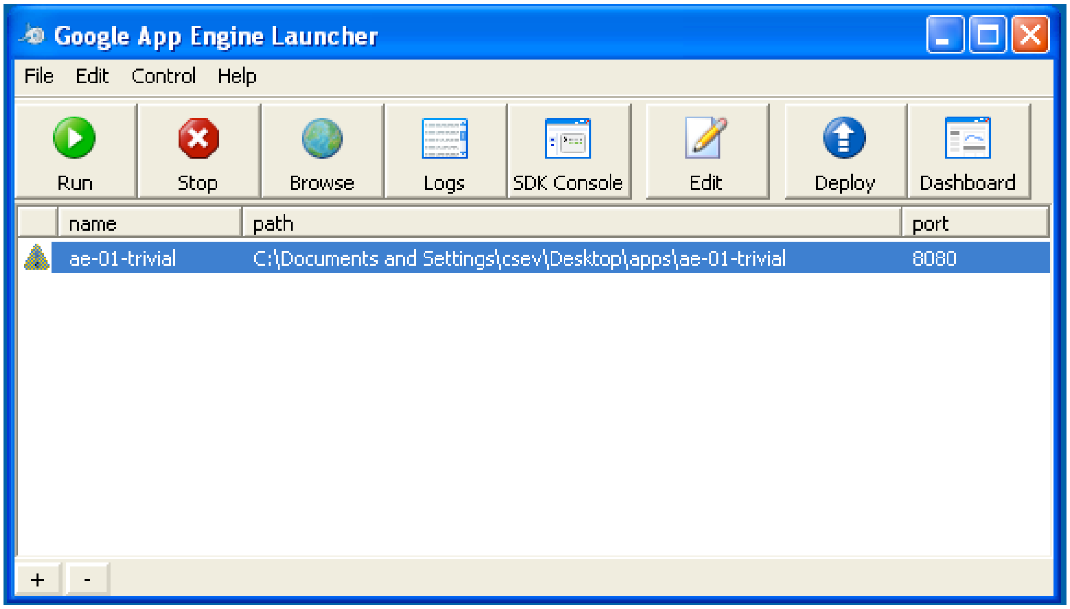


```
WARNING  2010-03-13 18:03:13,796 datastore_file_stub.py:623] Could not read
datastore data from c:\docume~1\csev\locals~1\temp\dev_appserver.datastore
WARNING  2010-03-13 18:03:13,796 dev_appserver.py:3581] Could not initialize
images API; you are likely missing the Python "PIL" module. ImportError: No module
named _imaging
INFO     2010-03-13 18:03:13,828 dev_appserver_main.py:399] Running application
ae-01-trivial on port 8080: http://localhost:8080
INFO     2010-03-13 18:03:24,717 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
INFO     2010-03-13 18:03:24,733 dev_appserver_index.py:205] Updating C:\Documents
and Settings\csev\Desktop\apps\ae-01-trivial\index.yaml
INFO     2010-03-13 18:03:24,967 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
2010-03-13 13:03:30 (Process exited with code -1)
```

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

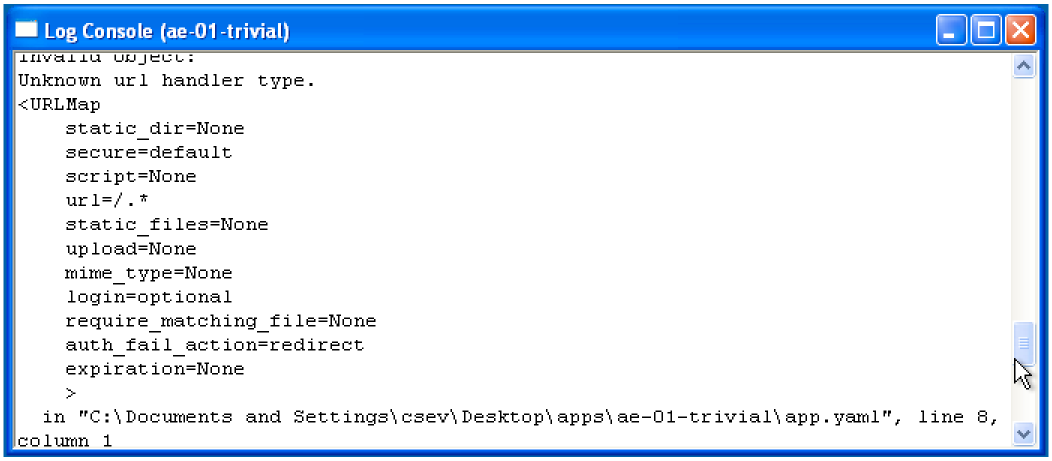
Dealing With Errors

With two files to edit, there are two general categories of errors that you may encounter. If you make a mistake on the **app. yaml** file, the App Engine will not start and your launcher will show a yellow icon near your application:



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

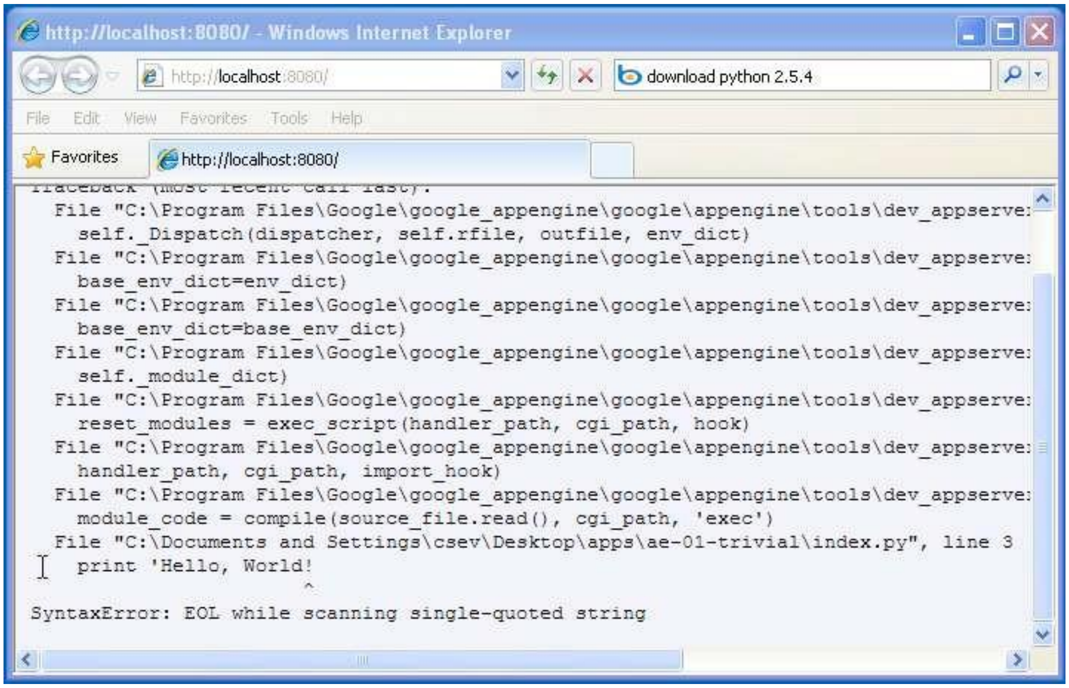
To get more detail on what is going wrong; take a look at the log for the application:



```
Invalid object:
Unknown url handler type.
<URLMap
    static_dir=None
    secure=default
    script=None
    url=/.*
    static_files=None
    upload=None
    mime_type=None
    login=optional
    require_matching_file=None
    auth_fail_action=redirect
    expiration=None
    >
  in "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\app.yaml", line 8,
column 1
```

In this instance—them is take is mis -indenting the last line in the **app .yaml** (line8).

If you make a syntax error in the **index.py**file ,a Python trace back error will appear in your browser.



```
Traceback (most recent call last):
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    self._Dispatch(dispatcher, self.rfile, outfile, env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    base_env_dict=env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    base_env_dict=base_env_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    self._module_dict)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    reset_modules = exec_script(handler_path, cgi_path, hook)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    handler_path, cgi_path, import_hook)
  File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve
    module_code = compile(source_file.read(), cgi_path, 'exec')
  File "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\index.py", line 3
    print 'Hello, World!
                       ^
SyntaxError: EOL while scanning single-quoted string
```

The error you need to see is likely to be the last few lines of the output–in this case I made a Python syntax error online one of our one-line application.

Reference: http://en.wikipedia.org/wiki/Stack_trace

When you make a mistake in the **app.yaml** file–you must the fix the mistake and attempt to start the application again.

If you make a mistake in a file like i**ndex.py** , you can simply fix the file and press refresh in your browser–there is none start the server.

Shutting Down the Server

To shut down the server, use the Launcher, select your application and press the **Stop** button.

Comments and questions to csev@umich.eduwww.dr-chuck.com

Result:

Thus, the Google App Engine is installed and the 'hello world' app and other simple web applications using python/java has been compiled successfully.

EX NO 4: Use GAE launcher to launch the web applications.

AIM:

To use GAE launcher to launch the web applications

Procedure:

You can use Google App Engine to hos t a static website. Static web pages can contain client-side technologies such as HTML, CSS, and JavaScript. Hosting your static site on App Engine can cost less than using a traditional hosting provider, As App Engine provides a free tier. Sites hosted on App ngine are hosted on the REGION_ID.r .app spot.com sub domain,
Such as [my-project-id].uc.r.appspot.com. After you deploy your site, you can
map your own domain name to your App Engine-hosted website

Before you begin

Before you can host your website on Google App Engine:

1,Create a new Cloud Console project or retrieve the project ID of an existing project to use:
Go to the Project page

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

Download the SDK

Creating a website to host on Google App Engine
Basic structure for the project This guide uses the following structure for the project:

app.yaml:
Configure the settings of your App Engine application

- www/: Directory to store all of your static files, such as HTML, CSS, images, and JavaScript.
- css/: Directory to store style sheets.
- style.css: Basic style sheet that formats the look and feel of your site.
- images/: Optional directory to store images.
- index.html: An HTML file that displays content for your website.
- js/: Optional directory to store JavaScript files.
- Other asset directories.

Creating the **app.yaml** file

The app.yaml file is a configuration file that tells App Engine how to map URLs to your static files. In the following steps, you will add handlers that will load www/index.html when someone visits your website,

and all static files will be stored in and called from the www directory.

Create the app.yaml file in your application's root directory:

1. Create a directory that has the same name as your project ID. You can find your

   project ID in the Console.

2. In directory that you just created, create a file named app.yaml.

3. Edit the app.yaml file and add the following code to the file:

```
runtime: python27
api_version: 1
threadsafe: true

handlers:
- url: /
  static_files: www/index.html
  upload: www/index.html

- url: /(.*)
  static_files: www/\1
  upload: www/(.*)
```

More reference information about the app.yaml file can be found in the app.yaml

Creating the **index.html** file

Create an HTML file that will be served when someone navigates to the root page of your website.

Store this file in your www directory.

```
 <html>
  <head>
    <title>Hello, world!</title>
    <linkrel="stylesheet"type="text/css"href="/css/style.css">
  </head>
  <body>
    <h1>Hello, world!</h1>
    <p>
      This is a simple static HTML file that will be served from Google App
      Engine.
    </p>
  </body>
</html>
```

Deploying your application to App Engine

When you deploy your application files, your website will be uploaded to App Engine.

To deploy your app, run the following command from within the root directory of your application where the app.yaml file is located:

```
g cloud app deploy
```

Optional flags:

- Include the --project flag to specify an alternate console project ID to what you initialized as the default
- in the g cloud CLI. Example: --project [YOUR_PROJECT_ID]
- Include the -v flag to specify a version ID, otherwise one is generated for you.

Example: -v [YOUR_VERSION_ID]

Viewing your application

To launch your browser and view the app at https://*PROJECT_ID*.*REGION_ID*.r.appspot.com,

run the following command:

```
g cloud app browse
```

Result:

Thus, the Google App Engine is used to launch the web applications successfully.

EX NO 5: Simulate a Cloud Scenario using Cloud Sim and Run a scheduling algorithm that is not present in Cloud Sim.

Aim:

To **Simulate a Cloud Scenario using Cloud Sim and Run a scheduling algorithm that is not present in Cloud Sim.**

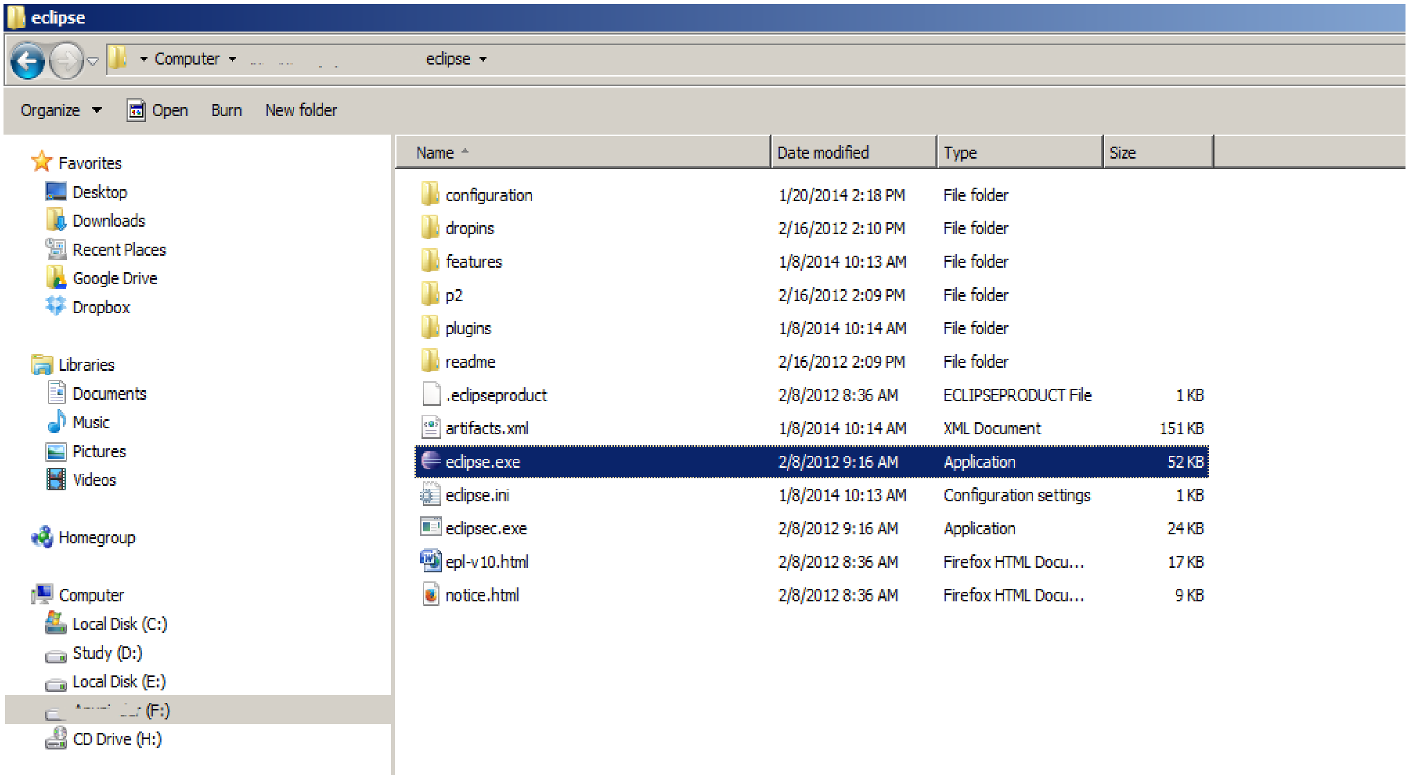
Procedure:

1. To start working with **'Cloud Sim simulation toolkit'** requires it to first configured Correctly.
2. Before you start to setup Cloud Sim, following resources must be Installed/downloaded on the local System Cloud sim simulation toolkit setup is easy. Before you start to setup Cloud Sim, following resources must be Installed/downloaded on the local system.

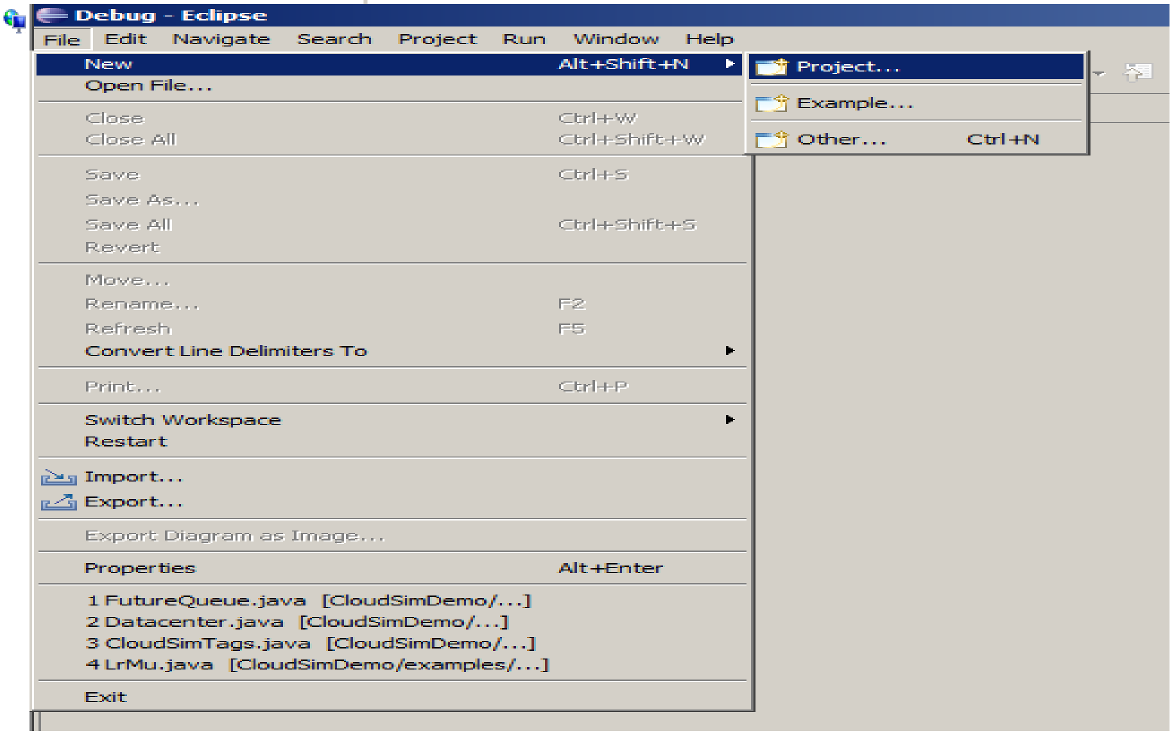
- **Java Development Kit(JDK):** As the Cloud sim simulation toolkit is a class library written in the
- Java programming language, therefore, the latest version of Java(JDK) should be installed on Your machine, which can be downloaded from Oracles Java portal. For assistance in the installation process, detailed documentation is provided by Oracle itself and you may follow the installation instructions
- **Eclipse IDE for Java developers:** As per your current installed operating system(Linux/ Windows). Before you download to make sure to check if 32-bit or 64-bit version is applicable to your Computer machine. Link for **Eclipse Kepler** version is available at the following link
- **Download Cloud Sim source code**: To date, various versions of Cloud Sim are released the latest version is 5.0, which is based on a container-based engine. Whereas to **keep the setup simple for beginners we will be setting up the most used version i.e. 3.0.3**, which can be directly downloaded by clicking on any of the following: Click for Windows or click for Linux.
- **One external requirement of Cloud sim** i.e. common jar package of math-related functions is to be downloaded from the Apache website or you may directly download by clicking here.

- Unzip **Eclipse**, **Cloud sim** and **Common Math libraries** to some common folder.

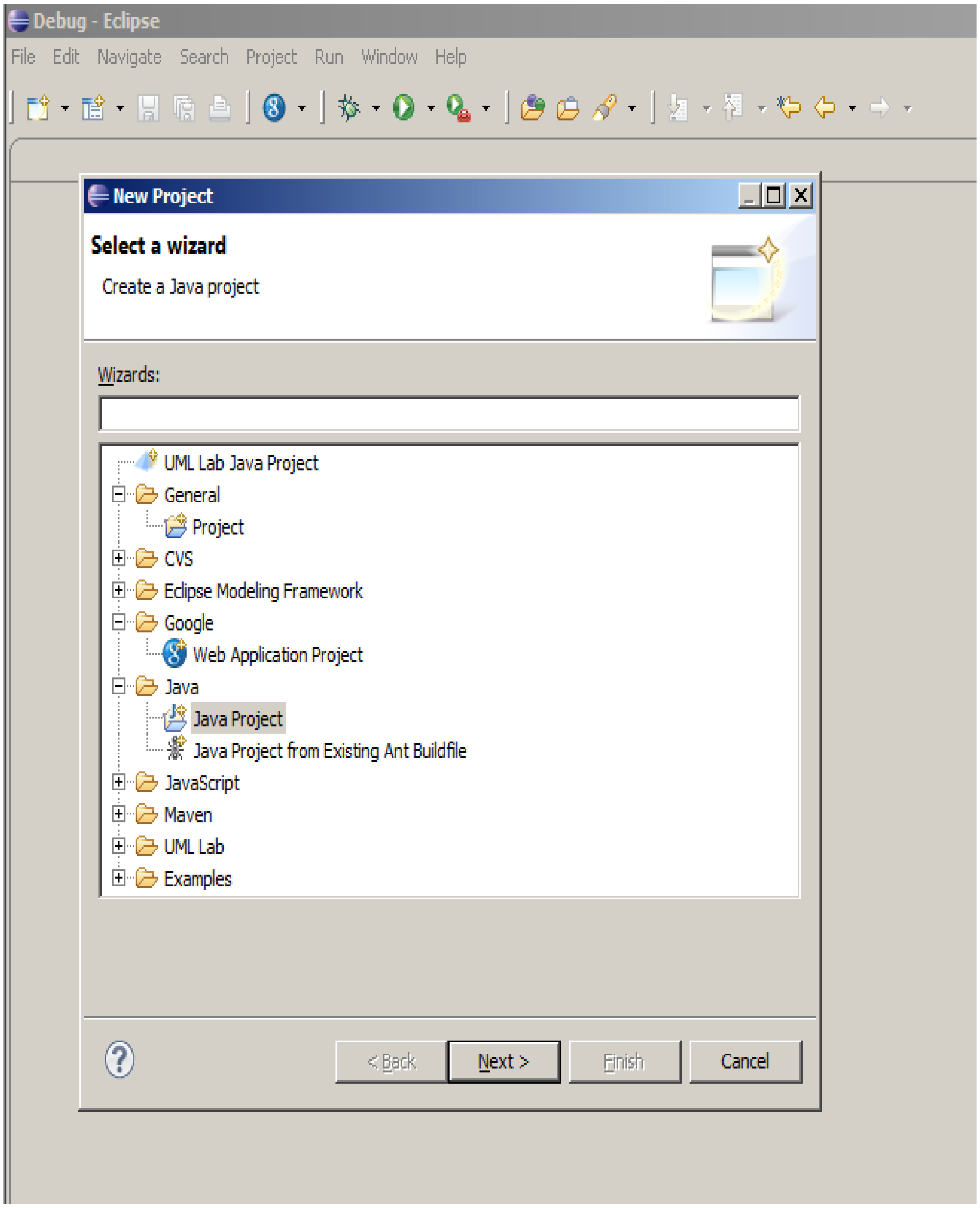
3. To install cloud sim First of all, navigate to the folder where you have unzipped the eclipse folder and open Eclipse.exe



Name	Date modified	Type	Size
configuration	1/20/2014 2:18 PM	File folder	
dropins	2/16/2012 2:10 PM	File folder	
features	1/8/2014 10:13 AM	File folder	
p2	2/16/2012 2:09 PM	File folder	
plugins	1/8/2014 10:14 AM	File folder	
readme	2/16/2012 2:09 PM	File folder	
.eclipseproduct	2/8/2012 8:36 AM	ECLIPSEPRODUCT File	1 KB
artifacts.xml	1/8/2014 10:14 AM	XML Document	151 KB
eclipse.exe	2/8/2012 9:16 AM	Application	52 KB
eclipse.ini	1/8/2014 10:13 AM	Configuration settings	1 KB
eclipsec.exe	2/8/2012 9:16 AM	Application	24 KB
epl-v10.html	2/8/2012 8:36 AM	Firefox HTML Docu...	17 KB
notice.html	2/8/2012 8:36 AM	Firefox HTML Docu...	9 KB



5 .A '***New Project***' wizard should open. There are a number of options displayed and you have to find & select the '***Java Project***' option, once done click '***Next***'.

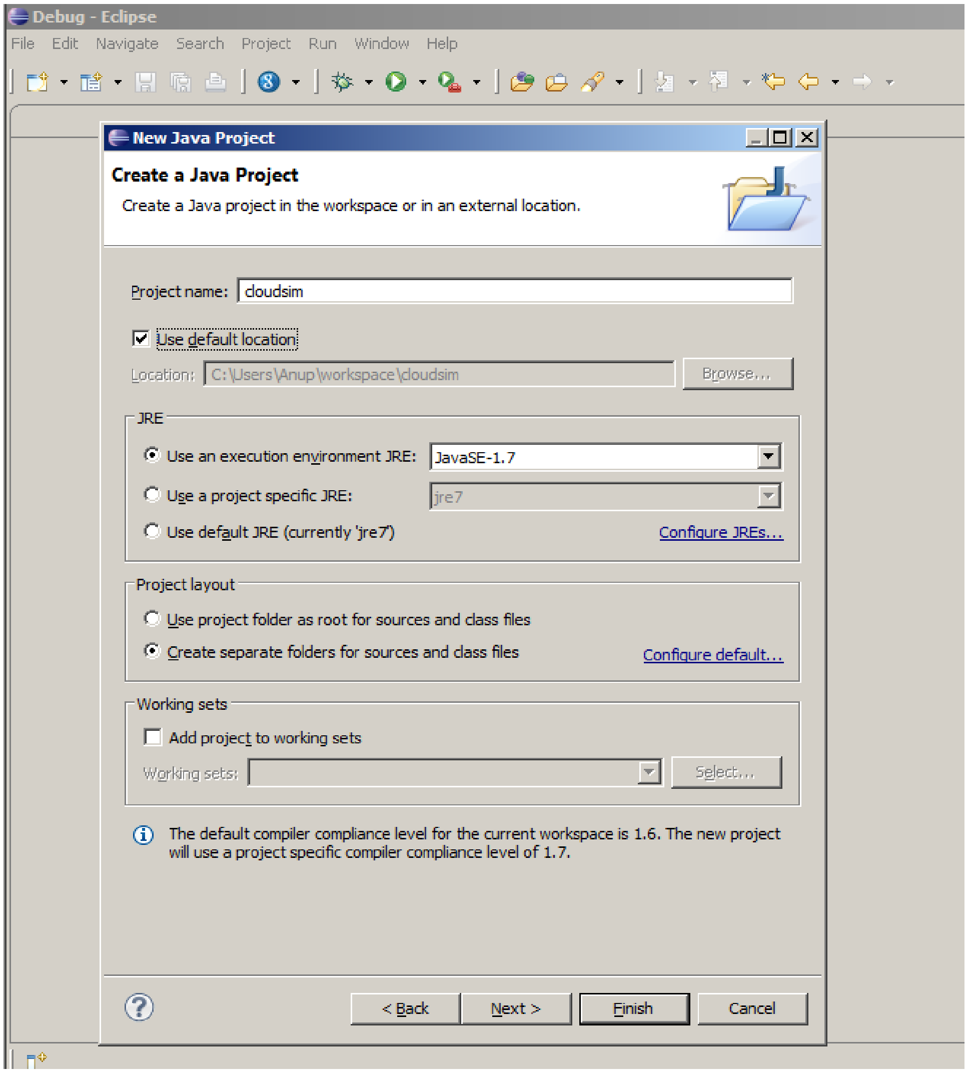


Select a wizard

Create a Java project

6. Now a detailed new project window will open, here you will provide the project name and the path of Cloud sim project source code, which will be done as follows:

- Project Name: CloudSim.



Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: cloudsim

☑ Use default location

Location: C:\Users\Anup\workspace\cloudsim [Browse...]

JRE

● Use an execution environment JRE: JavaSE-1.7

○ Use a project specific JRE: jre7

○ Use default JRE (currently 'jre7') Configure JREs...

Project layout

○ Use project folder as root for sources and class files

● Create separate folders for sources and class files Configure default...

Working sets

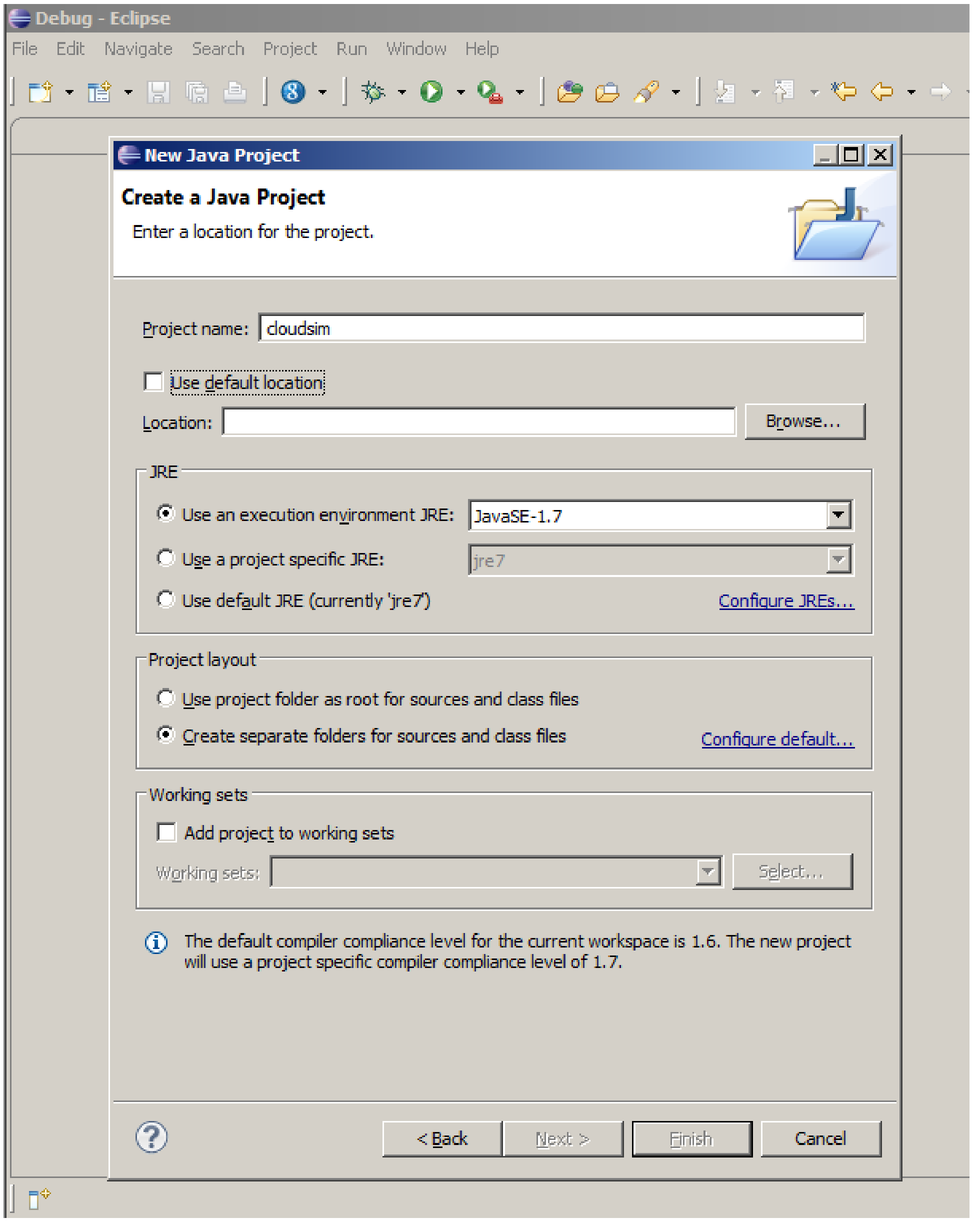
□ Add project to working sets

Working sets: [Select...]

The default compiler compliance level for the current workspace is 1.6. The new project will use a project specific compiler compliance level of 1.7.

< Back | Next > | Finish | Cancel

7. Unselect the ***'Use default location'*** option and then click on ***'Browse'*** to open the path where you have unzipped the Cloud sim project and finally click Next to set project settings.



Create a Java Project

Enter a location for the project.

Project name: cloudsim

☐ Use default location

Location:

JRE

● Use an execution environment JRE: JavaSE-1.7

○ Use a project specific JRE: jre7

○ Use default JRE (currently 'jre7')

Project layout

○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

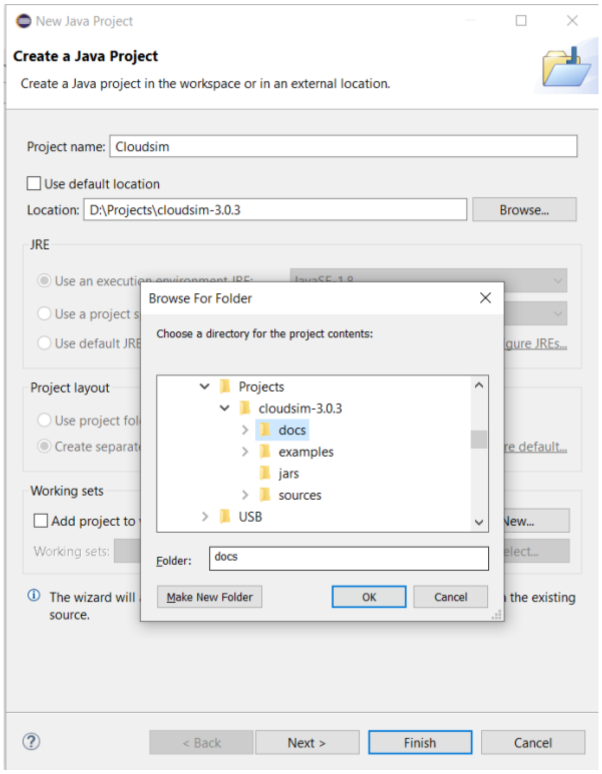
Working sets

☐ Add project to working sets

Working sets:

The default compiler compliance level for the current workspace is 1.6. The new project will use a project specific compiler compliance level of 1.7.

8 .Make sure you navigate the path till you can see the bin, docs, examples etc folder in the navigation plane.



New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: Cloudsim

□ Use default location

Location: D:\Projects\cloudsim-3.0.3

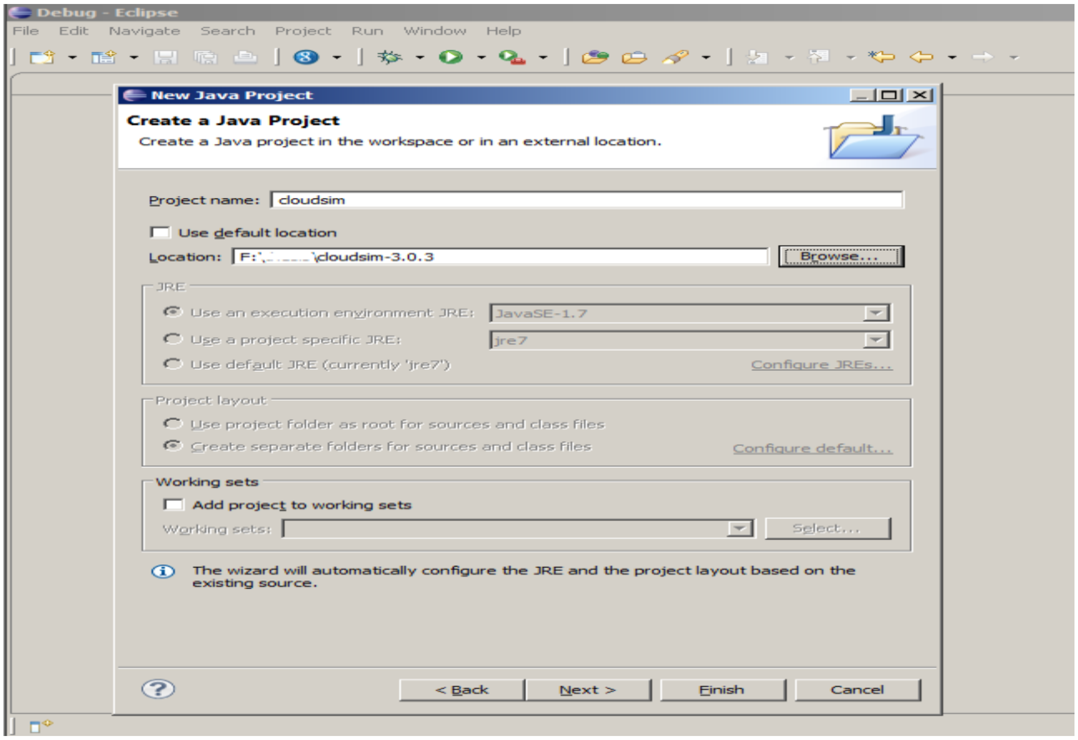
Browse For Folder

Choose a directory for the project contents:

- Projects
  - cloudsim-3.0.3
    - docs
    - examples
    - jars
    - sources
- USB

Folder: docs

9.Once done finally, click ‘Next’ to go to the next step i.e. setting up of project settings.



Debug - Eclipse

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: cloudsim

□ Use default location

Location: F:\____\cloudsim-3.0.3

JRE

● Use an execution environment JRE: JavaSE-1.7

○ Use a project specific JRE: jre7

○ Use default JRE (currently 'jre7')

Project layout

○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

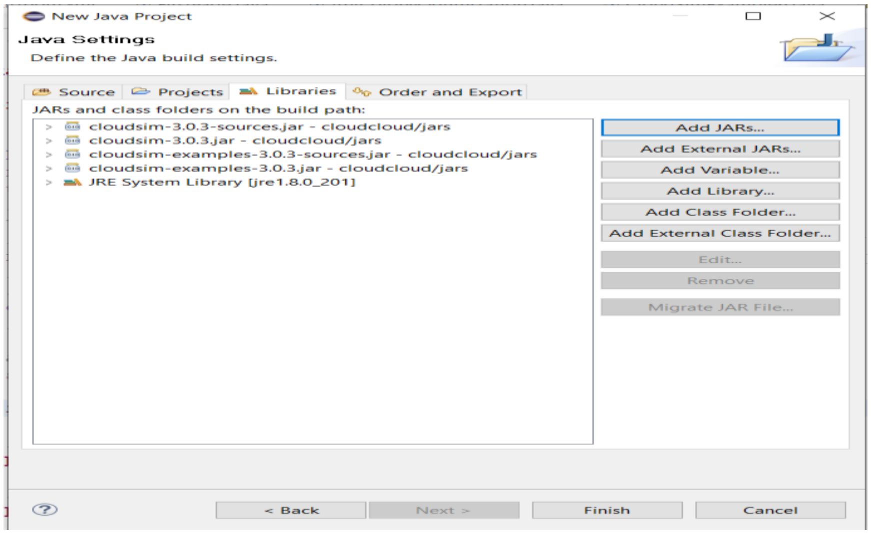
Working sets

□ Add project to working sets

Working sets:

The wizard will automatically configure the JRE and the project layout based on the existing source.

10. Now open ***'Libraries'*** tab and if you do not find commons-math3-3.x.jar *(here 'x' means the minor version release of the library which could be 2 or greater*) in the list then simply click on ***'Add External Jar'*** (commons-math3-3.x.jar will be included in the project from this step)



New Java Project

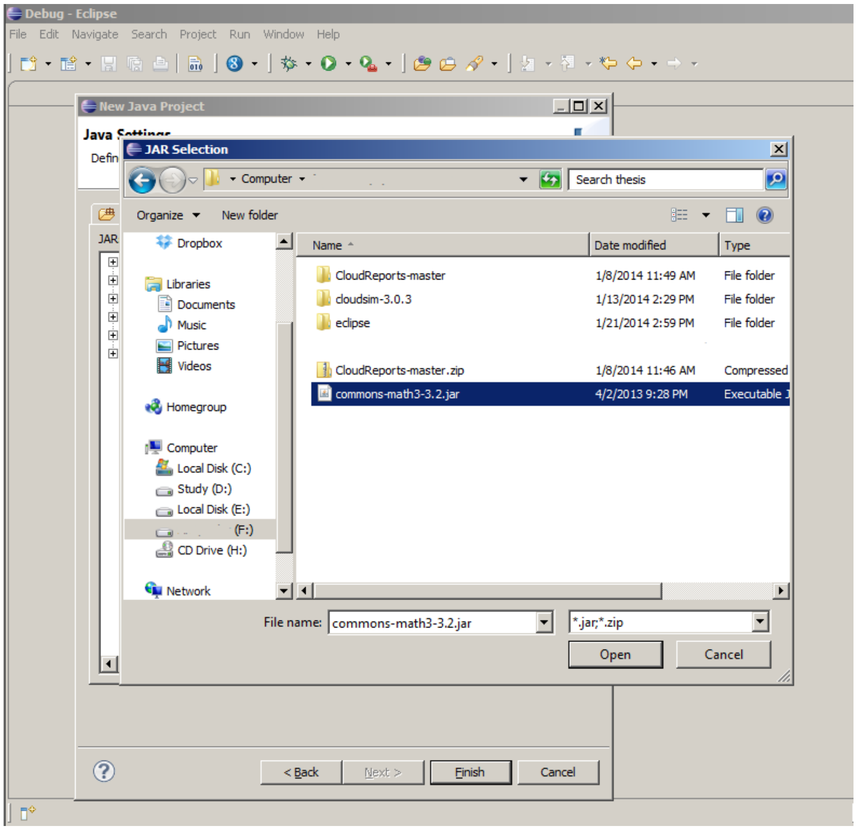
Java Settings

Define the Java build settings.

JARs and class folders on the build path:

- cloudsim-3.0.3-sources.jar - cloudcloud/jars
- cloudsim-3.0.3.jar - cloudcloud/jars
- cloudsim-examples-3.0.3-sources.jar - cloudcloud/jars
- cloudsim-examples-3.0.3.jar - cloudcloud/jars
- JRE System Library [jre1.8.0_201]

11. Once you have clicked on '***Add External JAR's***' Open the path where you have unzipped the commons-math binaries and select '***Commons-math3-3.x.jar***' and click on open.



Debug - Eclipse

New Java Project

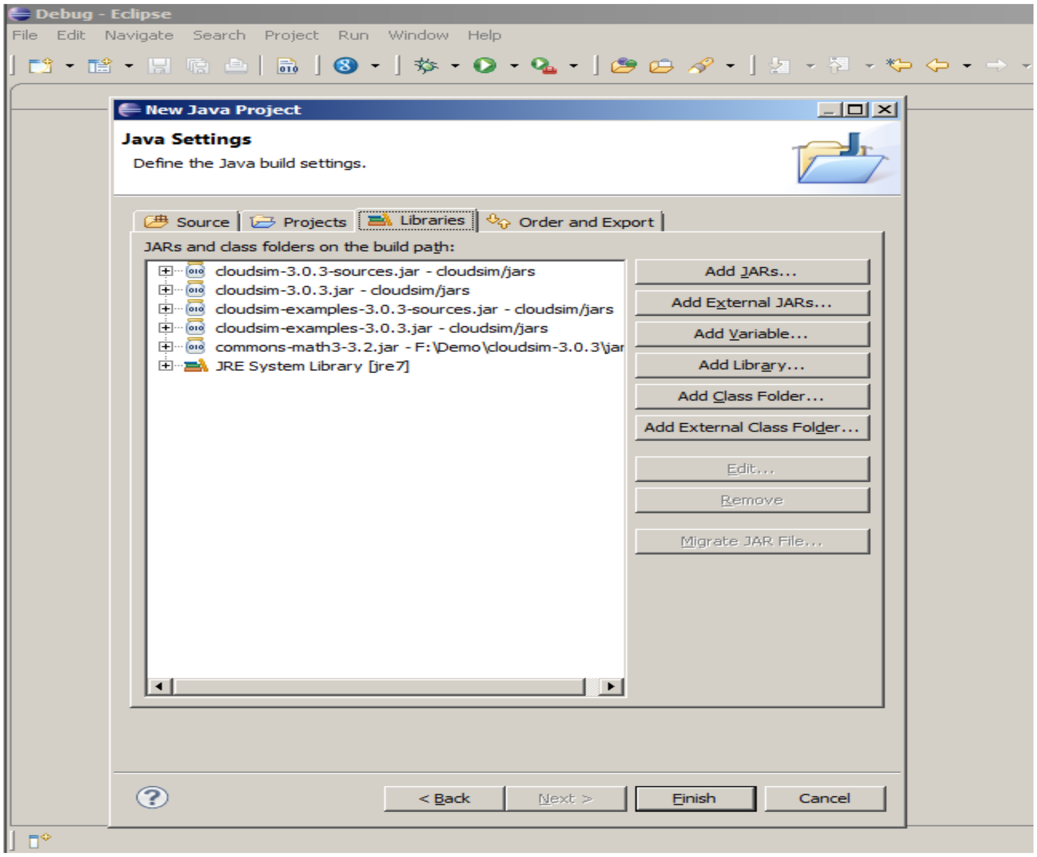
JAR Selection

Name	Date modified	Type
CloudReports-master	1/8/2014 11:49 AM	File folder
cloudsim-3.0.3	1/13/2014 2:29 PM	File folder
eclipse	1/21/2014 2:59 PM	File folder
CloudReports-master.zip	1/8/2014 11:46 AM	Compressed
commons-math3-3.2.jar	4/2/2013 9:28 PM	Executable J

File name: commons-math3-3.2.jar

.jar;.zip

12. Ensure external jar that you opened in the previous step is displayed in the list and then click on '***Finish***' (your system may take 2-3 minutes to configure the project)



Java Settings

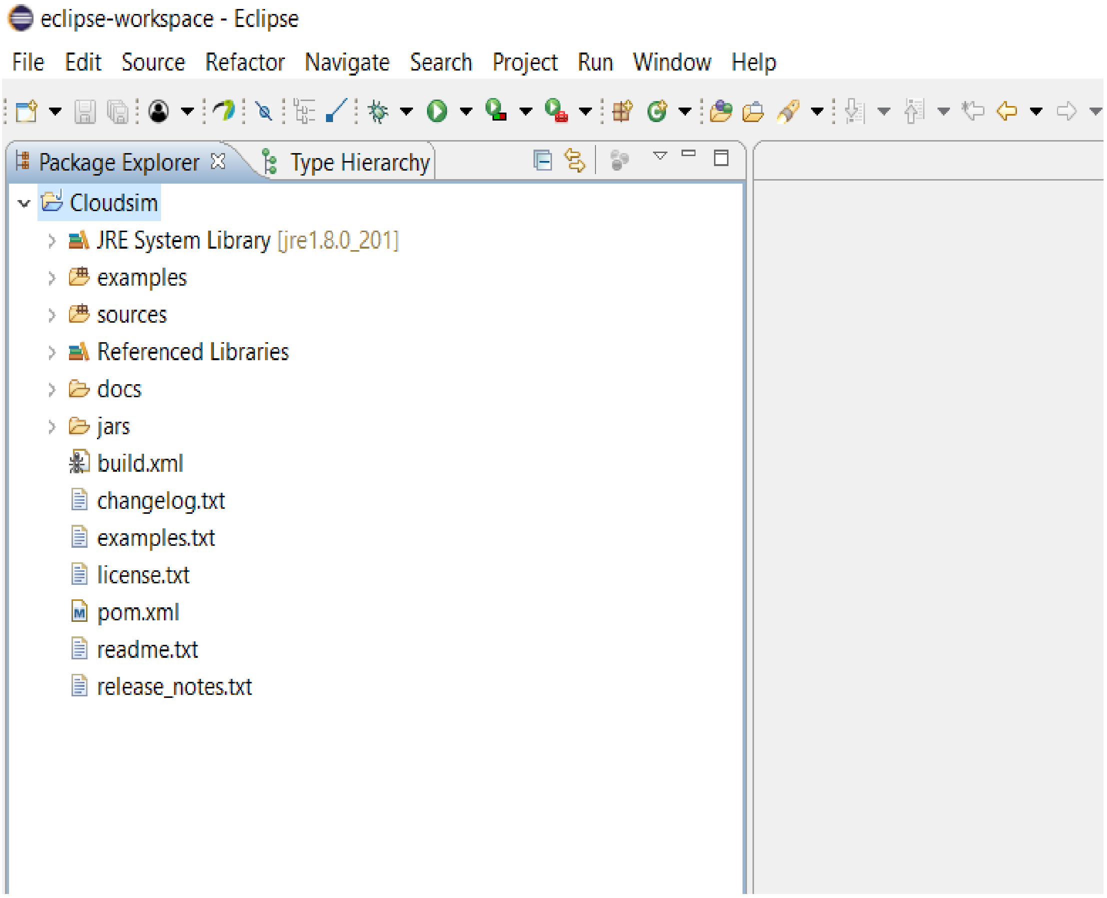
Define the Java build settings.

JARs and class folders on the build path:

- cloudsim-3.0.3-sources.jar - cloudsim/jars
- cloudsim-3.0.3.jar - cloudsim/jars
- cloudsim-examples-3.0.3-sources.jar - cloudsim/jars
- cloudsim-examples-3.0.3.jar - cloudsim/jars
- commons-math3-3.2.jar - F:\Demo\cloudsim-3.0.3\jar
- JRE System Library [jre7]

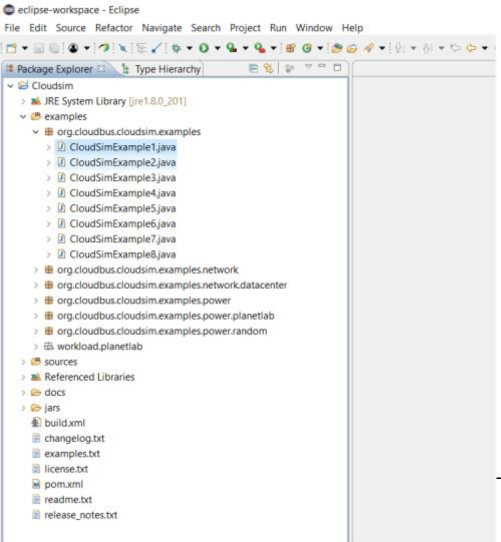
13. Once the project is configured you can open the '***Project Explorer***' and start exploring the Cloud sim project. Also for the first time eclipse automatically start building the workspace for newly configured Cloud sim project, which may take some time depending on the configuration of the computer system.

14. Following is the final screen which you will see after Cloud sim is configured.

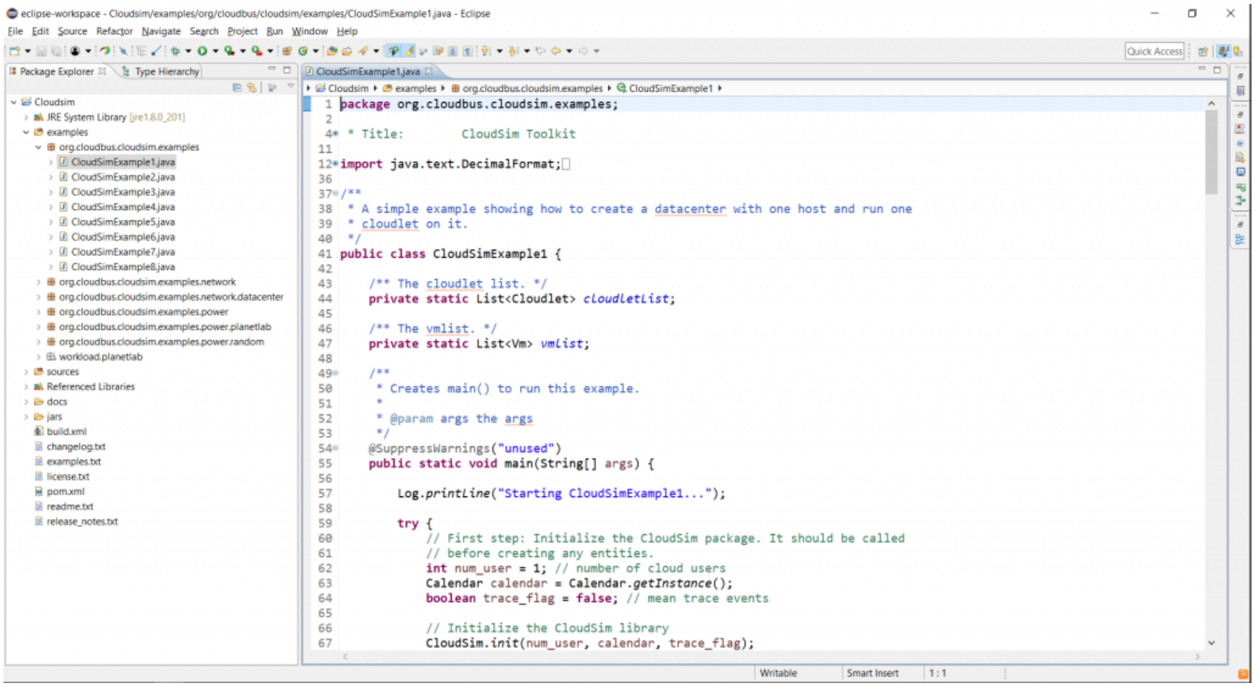

eclipse-workspace - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Type Hierarchy

Cloudsim
JRE System Library [jre1.8.0_201]
examples
sources
Referenced Libraries
docs
jars
build.xml
changelog.txt
examples.txt
license.txt
pom.xml
readme.txt
release_notes.txt

15.Now just to check you within the '**Project Explorer**', you should navigate to the '**examples**' folder, then expand the package '***org.cloudbus.cloud sim.examples***' and double click to open the '***CloudsimExample1.java***'.

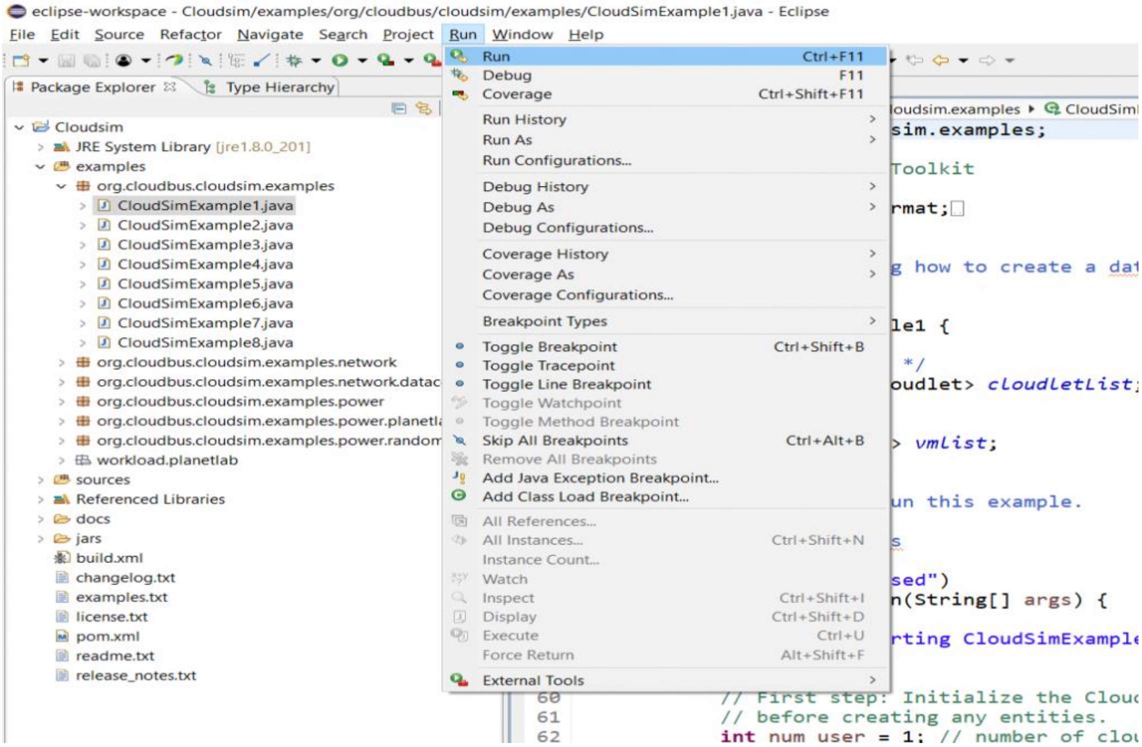

eclipse-workspace - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Type Hierarchy

Cloudsim
JRE System Library [jre1.8.0_201]
examples
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.datacenter
org.cloudbus.cloudsim.examples.power
org.cloudbus.cloudsim.examples.power.planetlab
org.cloudbus.cloudsim.examples.power.random
workload.planetlab
sources
Referenced Libraries
docs
jars
build.xml
changelog.txt
examples.txt
license.txt
pom.xml
readme.txt
release_notes.txt


eclipse-workspace - Cloudsim/examples/org/cloudbus/cloudsim/examples/CloudSimExample1.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
Type Hierarchy
Cloudsim
JRE System Library [jre1.8.0_201]
examples
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.datacenter
org.cloudbus.cloudsim.examples.power
org.cloudbus.cloudsim.examples.power.planetlab
org.cloudbus.cloudsim.examples.power.random
workload.planetlab
sources
Referenced Libraries
docs
jars
build.xml
changelog.txt
examples.txt
license.txt
pom.xml
readme.txt
release_notes.txt
CloudSimExample1.java
Cloudsim ▸ examples ▸ org.cloudbus.cloudsim.examples ▸ CloudSimExample1 ▸
Writable
Smart Insert
1 : 1

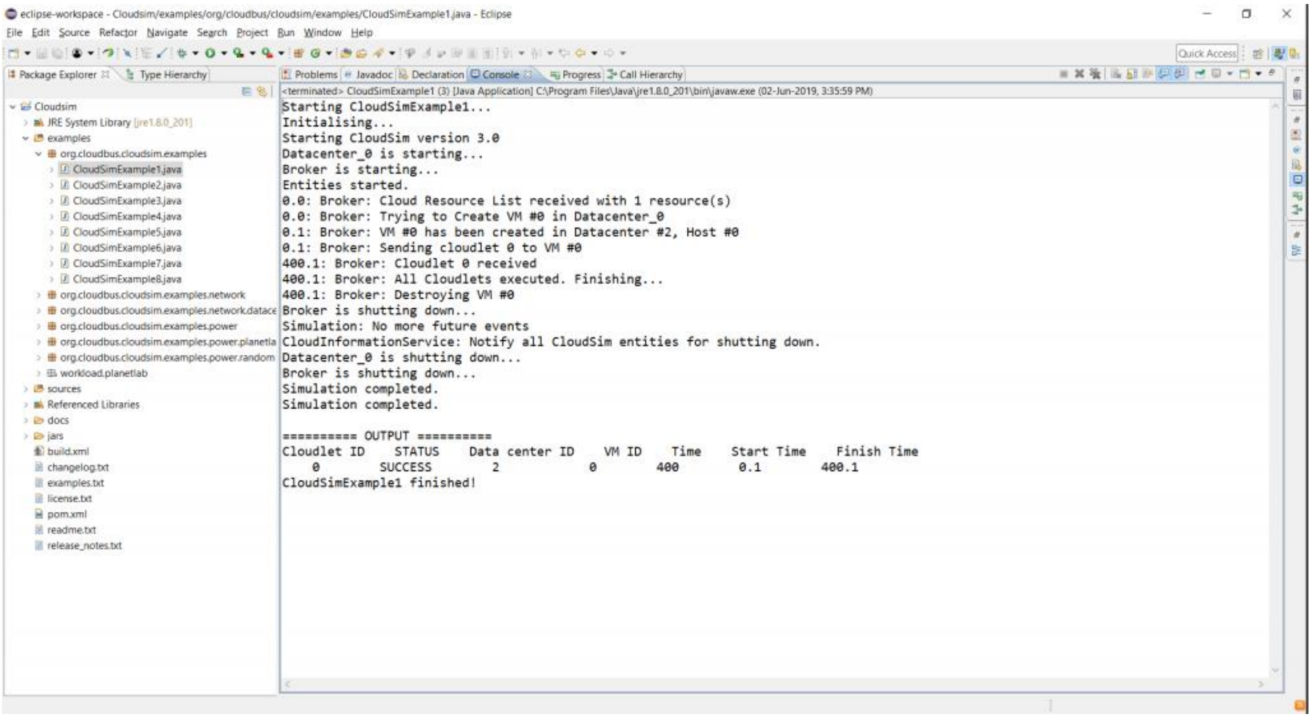
```java
package org.cloudbus.cloudsim.examples;

 * Title:        CloudSim Toolkit

import java.text.DecimalFormat;

/**
 * A simple example showing how to create a datacenter with one host and run one
 * cloudlet on it.
 */
public class CloudSimExample1 {

    /** The cloudlet list. */
    private static List<Cloudlet> cloudletList;

    /** The vmlist. */
    private static List<Vm> vmlist;

    /**
     * Creates main() to run this example.
     *
     * @param args the args
     */
    @SuppressWarnings("unused")
    public static void main(String[] args) {

        Log.printLine("Starting CloudSimExample1...");

        try {
            // First step: Initialize the CloudSim package. It should be called
            // before creating any entities.
            int num_user = 1; // number of cloud users
            Calendar calendar = Calendar.getInstance();
            boolean trace_flag = false; // mean trace events

            // Initialize the CloudSim library
            CloudSim.init(num_user, calendar, trace_flag);
```

16. Now navigate to the Eclipse menu ‘***Run -> Run***‘ or directly use a keyboard shortcut ***‘Ctrl + F11’*** to execute the ‘***CloudsimExample1.java***‘.



```java
sim.examples;
Toolkit
rmat;
g how to create a dat
le1 {
 */
oudlet> cloudletList;
> vmList;
un this example.
s
sed")
n(String[] args) {
rting CloudSimExampl
        // First step: Initialize the Clou
        // before creating any entities.
        int num user = 1; // number of clou
```

17. if it is successfully executed it should be displaying the following type to output in the console window of the Eclipse IDE.



eclipse-workspace - Cloudsim/examples/org/cloudbus/cloudsim/examples/CloudSimExample1.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
Type Hierarchy
Cloudsim
JRE System Library [jre1.8.0_201]
examples
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.datace
org.cloudbus.cloudsim.examples.power
org.cloudbus.cloudsim.examples.power.planetla
org.cloudbus.cloudsim.examples.power.random
workload.planetlab
sources
Referenced Libraries
docs
jars
build.xml
changelog.txt
examples.txt
license.txt
pom.xml
readme.txt
release_notes.txt
Problems
Javadoc
Declaration
Console
Progress
Call Hierarchy

```
<terminated> CloudSimExample1 (3) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (02-Jun-2019, 3:35:59 PM)
Starting CloudSimExample1...
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
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
400.1: Broker: Cloudlet 0 received
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!
```

Now you have successfully setup/Installed the Cloud Sim Simulation Toolkit on your computer machine and you are ready to start with the experimentation.

EX. NO 6: Find a Procedure to Transfer the files from one Virtual Machine To Another Virtual Machine

AIM

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

PROCEDURE

1. You can copy few (or more) lines with **copy & paste** mechanism. For this you need to share clipboard between host OS and guest OS, installing **Guest Addition** on both the virtual machines (probably setting bidirectional and restarting them). You copy from guest OS in the clipboard that is shared with the host OS.

   Then you paste from the host OS to the second guest OS.

2. You can enable **dragand drop** too with the same method (Click on the machine, settings, general, advanced, drag and drop: set to bidirectional)

3. You can have **common Shared Folders** on both virtual machines and use one of the directory shared as buffer to copy.

   Installing **Guest Additions** you have the possibility to set Shared Folders too. As you put a file in a shared folder from host OS or from guest OS, is immediately visible to the other. (Keep in mind that can arise some problems for date/time of the files when there are different clock settings on the different virtual machines).

   If you use the same folder shared on more machines you can exchange files directly copying them in this folder.

4. You can use **usual method to copy files between 2 different computer** with client-server application. (e.g. scp with sshd active for linux, win scp... you can get some info about SSH servers e.g. here)

   You need an active server(sshd)on the receiving machine and a client on the sending machine. Of course you need to have the authorization setted (via password or, better, via an automatic authentication method).

   Note: many Linux/Ubuntu distribution install sshd by default: you can see if it is running with pgrep sshd from a shell. You can install with sudo apt-get install open ssh-server.

5. You can **mount part of the file system** of a virtual machine via NFS or SSHFS on the other, or you can **share file and directory** with Samba.

   You may find interesting the article Sharing files between guest and host without Virtual Box shared folders with detailed step by step instructions.

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

- Each virtual machine has its own operative system running on and acts as a physical machine.
- Each virtual machine is an instance of a program owned by an user in the hosting operative system and should undergo the restrictions of the user in the hosting OS.

  E.g Let we say that Hastur and Meow are users of the hosting machine, but they did not allow each other to see their directories (no read/write/execute authorization). When each of them run a virtual machine, for the hosting OS those virtual machine are two normal programs owned by Hastur and Meow and cannot see the private directory of the other user. This is a restriction due to the hosting OS. It's easy to overcame it: it's enough to give authorization to read/write/execute to a directory or to chose a different directory in which both users can read/write/execute.

- Windows likes mouse and Linux fingers.:-)

  I mean I suggest you to enable Drag & drop to be copy with the Windows machines and the Shared folders or to be copy with Linux.

  When you will need to be fast with Linux **you will feel the need** of ssh-keygen and to Generate once SSH Keys to copy files on/from a remote machine without writing password anymore. In this way it functions bash auto-completion remotely too!

RESULT

Thus ,the procedure to transfer the files from one virtual machine to another virtual machine was completed successfully.

Ex no. 7 Install Hadoop Single Node Cluster and Run Simple Applications like Word Count

Aim:

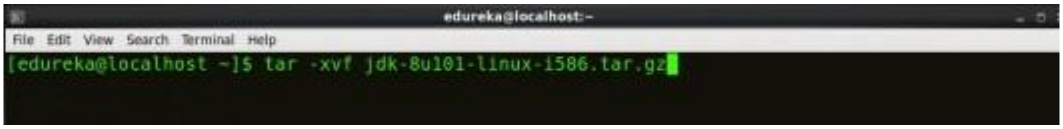
To install Hadoop single node cluster and run simple Applications like word count

PROCEDURE:

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

Step 2: Extract the Java Tar File.

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

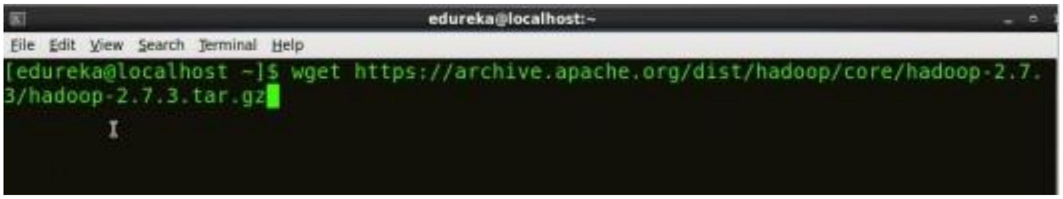


```
[edureka@localhost ~]$ tar -xvf jdk-8u101-linux-i586.tar.gz
```

Fig: Hadoop Installation – Extracting Java Files

Step 3: Download the Hadoop 2.7.3 Package.

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

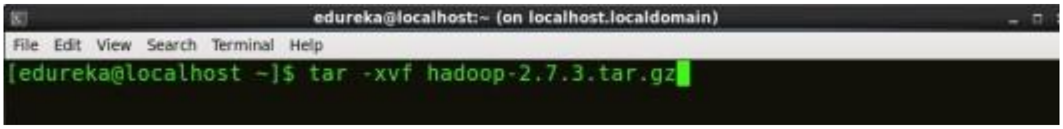


```
[edureka@localhost ~]$ wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.
3/hadoop-2.7.3.tar.gz
```

Fig: Hadoop Installation – Downloading Hadoop

Step 4: Extract the Hadoop tar File.

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


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

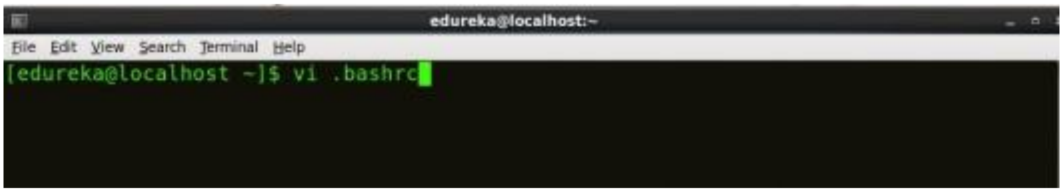
```
[edureka@localhost ~]$ tar -xvf hadoop-2.7.3.tar.gz
```

Fig: Hadoop Installation – Extracting Hadoop Files **Step 5:**

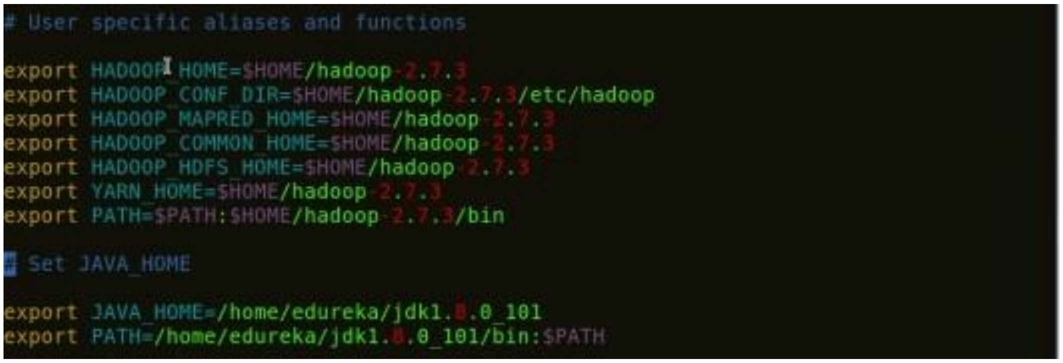
Add the Hadoop and Java paths in the bash file (.bashrc). Open**. bashrc** file.

Now, add Hadoop and Java Path as shown below.

Command**:** vi .bashrc



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



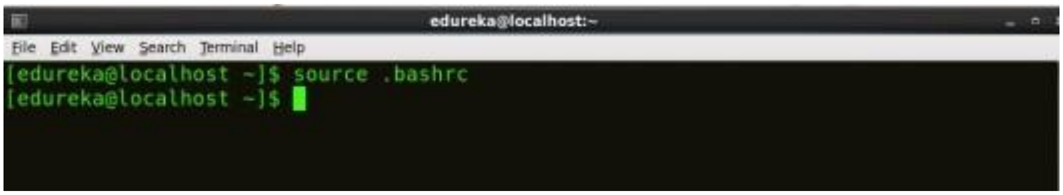
```
# User specific aliases and functions

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

# Set JAVA_HOME

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

Fig: Hadoop Installation – Setting Environment Variable

Then, save the bash file and close it.

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

Command**:** source .bashrc

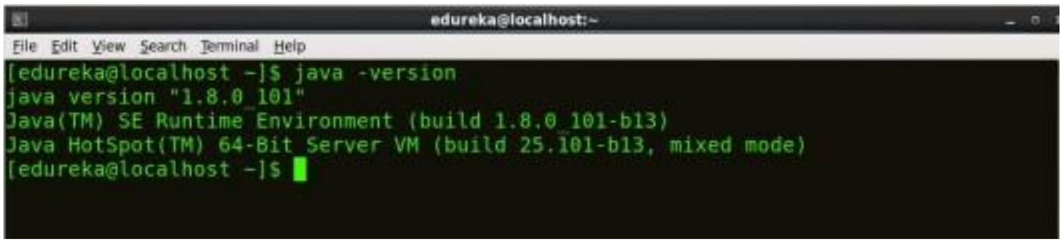


```
[edureka@localhost ~]$ source .bashrc
[edureka@localhost ~]$ 
```

Fig: Hadoop Installation – Refreshing environment variables

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

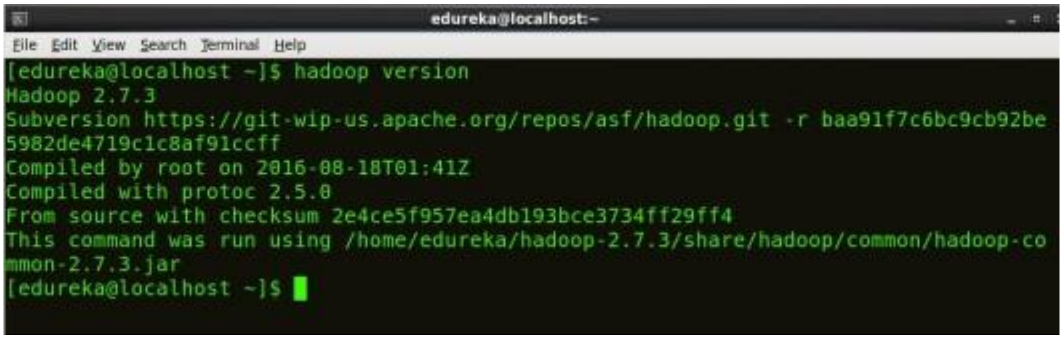
Command**:** java -version



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

Fig: Hadoop Installation – Checking Java Version

***Command*:** hadoop version



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

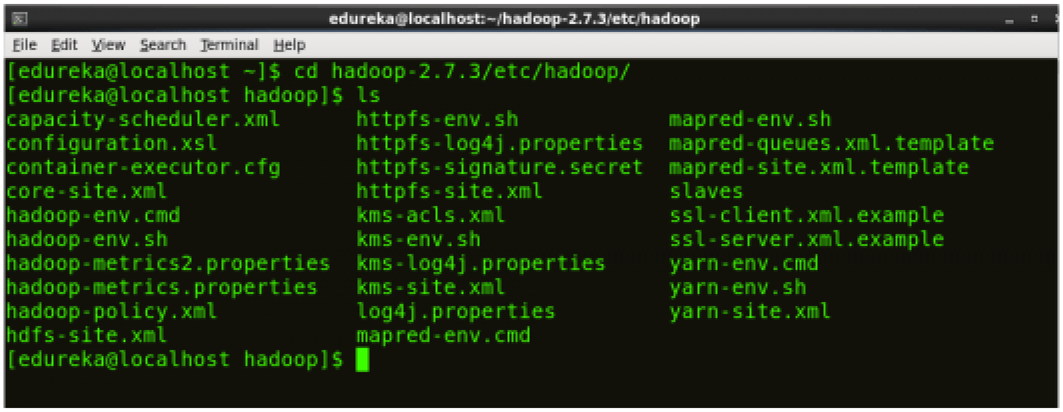
Fig: Hadoop Installation – Checking Hadoop Version

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

Command: cd hadoop-2.7.3/etc/hadoop/

Command: ls

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


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

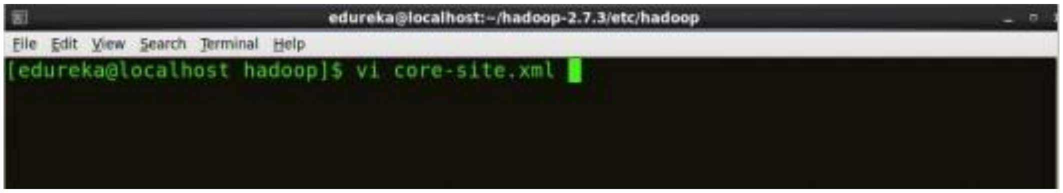
```
[edureka@localhost ~]$ cd hadoop-2.7.3/etc/hadoop/
[edureka@localhost hadoop]$ ls
capacity-scheduler.xml      httpfs-env.sh             mapred-env.sh
configuration.xsl           httpfs-log4j.properties   mapred-queues.xml.template
container-executor.cfg      httpfs-signature.secret   mapred-site.xml.template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

Fig: Hadoop Installation – Hadoop Configuration Files

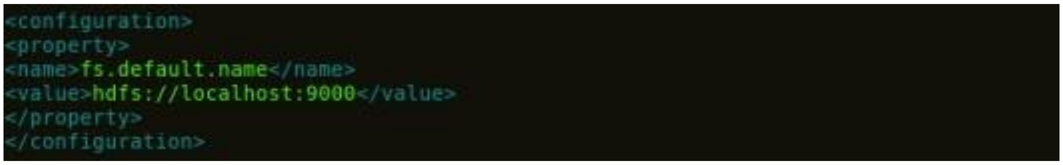
Step 7: Open *core-site.xml* and edit the property mentioned below inside configuration tag:

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

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


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

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



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

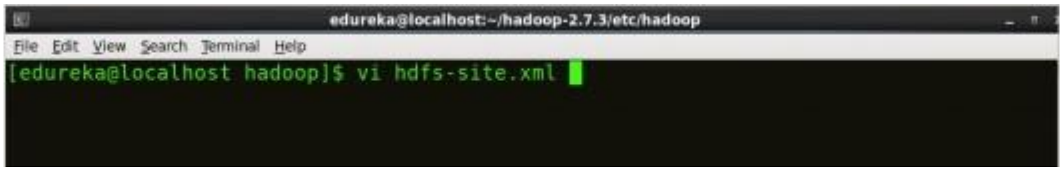
Fig: Hadoop Installation – Configuring core-site.xml

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

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

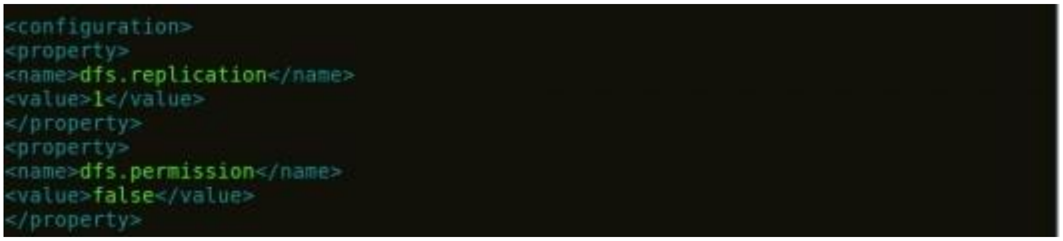
hdfs-site.xml contains configuration settings of HDFS daemons (i.e. Name Node, Data Node, Secondary Name Node). It also includes the replication factor and block size of HDFS.

Command: vi hdfs-site.xml



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

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



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

Fig: Hadoop Installation – Configuring hdfs-site.xml

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

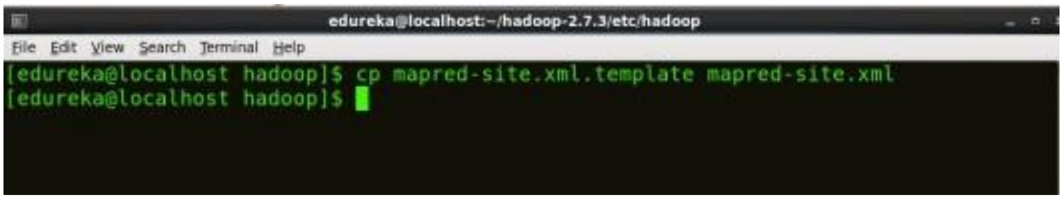
Step 9: Edit the *mapred-site.xml* file and edit the property mentioned below inside configuration tag:

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

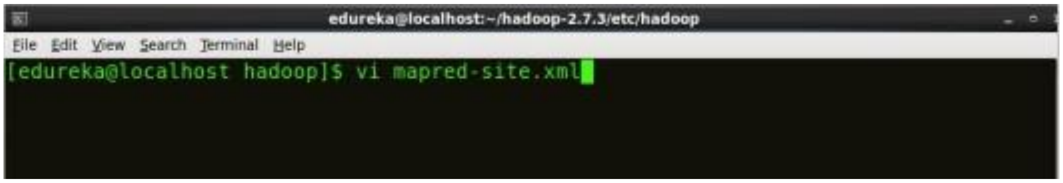
In some cases, mapred-site.xml file is not available. So, we have to create the map red-site.xml file using mapred-site.xml template.

Command: cp map red-site.xml.template mapred-site.xml

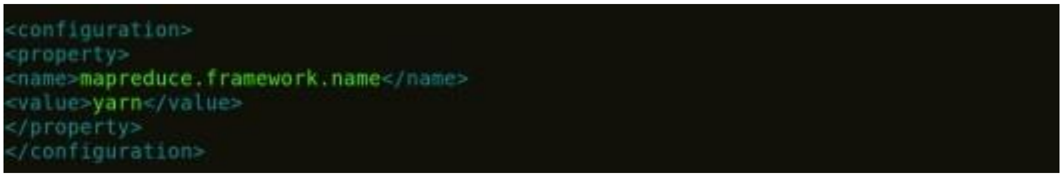
***Command*:** vi mapred-site.xml.


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

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


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

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



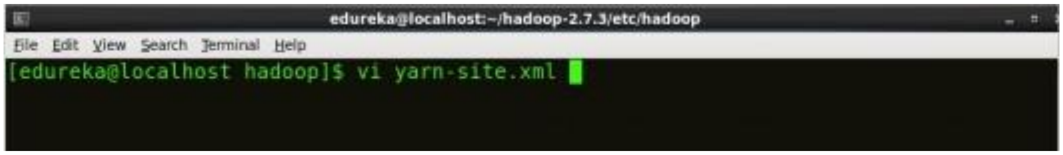
```
<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>
```

Fig: Hadoop Installation – Configuring mapred-site.xml

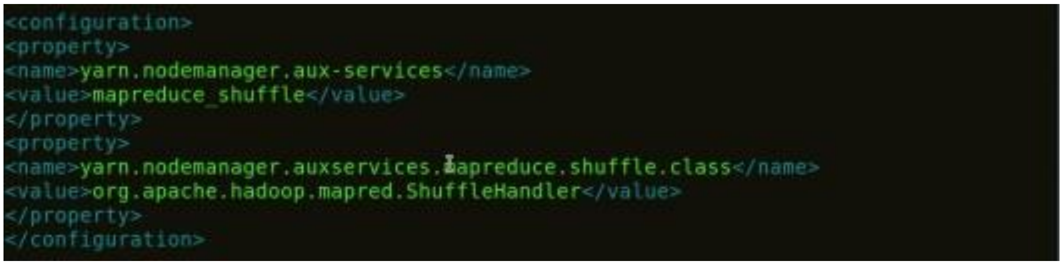
```
<?xmlversion="1.0"encoding="UTF-8"?>
<?xml-stylesheettype="text/xsl"href="configuration.xsl"?>
<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>
```

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

yarn-site.xml contains configuration settings of Resource Manager and Node Manager like application memory management size, the operation needed on program & algorithm, etc.
***Command*:** vi yarn-site.xml


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

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



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

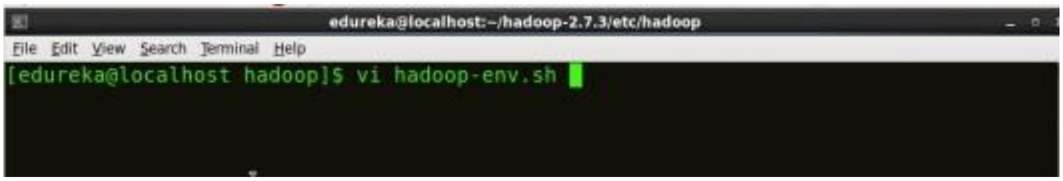
Fig: Hadoop Installation – Configuring yarn-site.xml

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

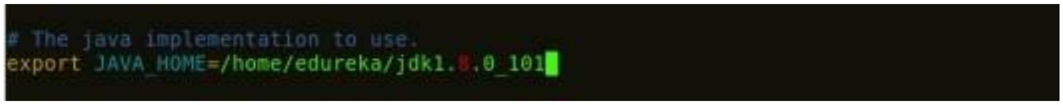
Step 11: Edit *hadoop-env.sh* and add the Java Path as mentioned below:

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

***Command*:** vi hadoop–env.sh


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

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



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

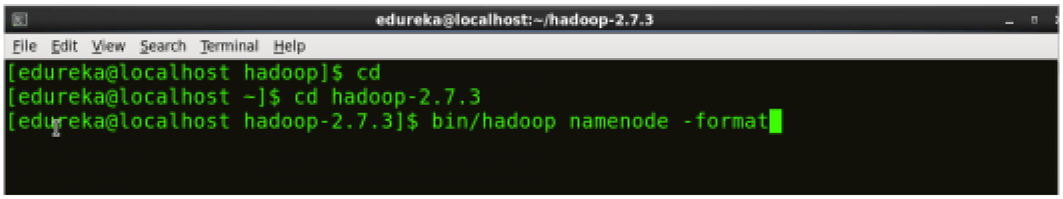
Fig: Hadoop Installation – Configuring hadoop-env.sh

Step 12: Go to Hadoop home directory and format the Name Node.

Command: cd

Command: cd hadoop-2.7.3

Command: bin/hadoop name node -format


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

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

Fig: Hadoop Installation – Formatting Name Node

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

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

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

Command: cd hadoop-2.7.3/sbin

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

Command: ./start-all.sh

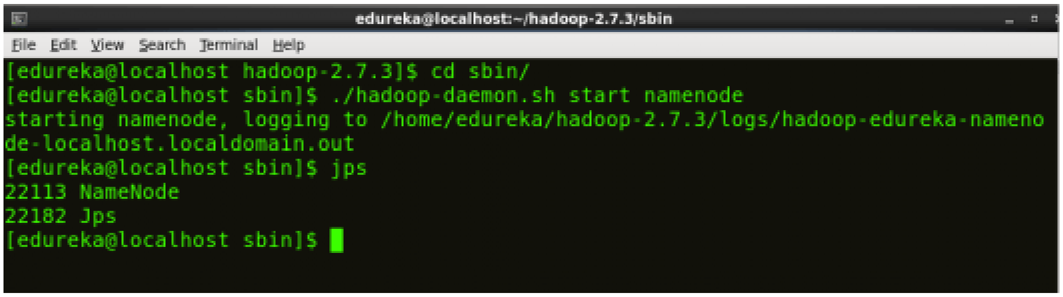
The above command is a combination of ***start-dfs.sh, start-yarn.sh*** &***mr-job history-daemon.sh***

Or you can run all the services individually as below:

Start Name Node:

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

Command: ./hadoop-daemon.sh start name node


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

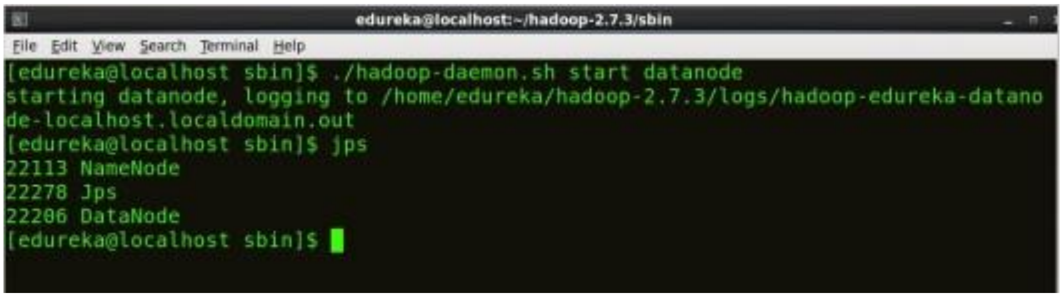
```
[edureka@localhost hadoop-2.7.3]$ cd sbin/
[edureka@localhost sbin]$ ./hadoop-daemon.sh start namenode
starting namenode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-nameno
de-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22182 Jps
[edureka@localhost sbin]$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

Start Data Node:

Fig: Hadoop Installation – Starting Name Node

On startup, a Data Node connects to the Name node and it responds to the requests from the Name node for different operations.

Command: ./hadoop-daemon.sh start data node


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

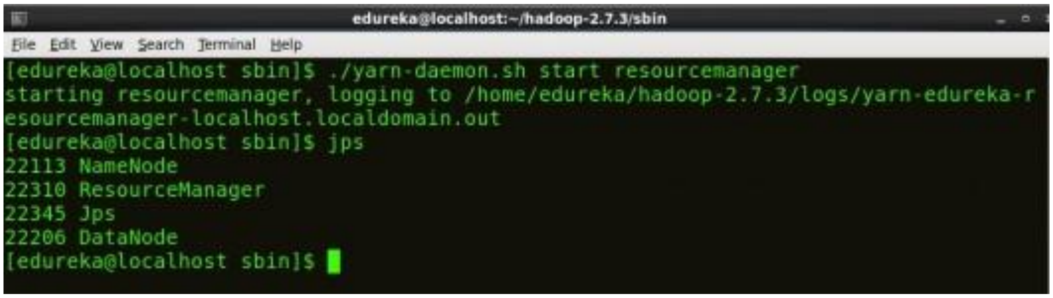
```
[edureka@localhost sbin]$ ./hadoop-daemon.sh start datanode
starting datanode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-datano
de-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22278 Jps
22206 DataNode
[edureka@localhost sbin]$
```

Fig: Hadoop Installation – Starting Data Node

Start Resource Manager:

Resource Manager is the master that arbitrates all the available cluster resources and thus helps in managing the distributed applications running on the YARN system. Its work is to manage each Node Managers and the each application's Application Master.

Command: ./yarn-daemon.sh start resource manager


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

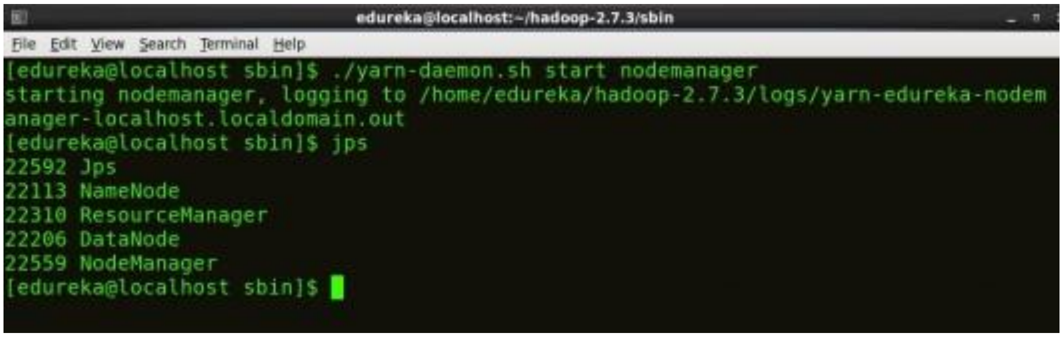
```
[edureka@localhost sbin]$ ./yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /home/edureka/hadoop-2.7.3/logs/yarn-edureka-r
esourcemanager-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22310 ResourceManager
22345 Jps
22206 DataNode
[edureka@localhost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

Fig: Hadoop Installation – Starting Resource Manager

Start Node Manager:

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

Command: ./yarn-daemon.sh start node manager


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

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

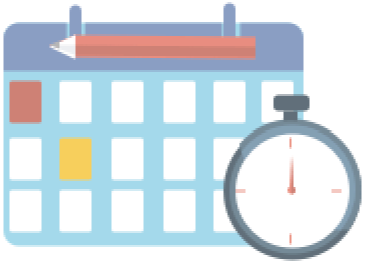


Fig: Hadoop Installation – Starting Node Manager

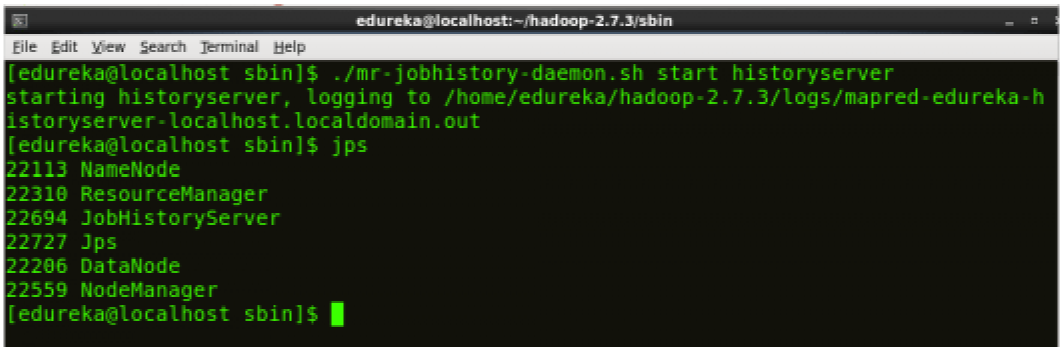
Start Job History Server:

Job History Server is responsible for servicing all job history related requests from client.

Command: ./mr-jobhistory-daemon.sh start history server

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

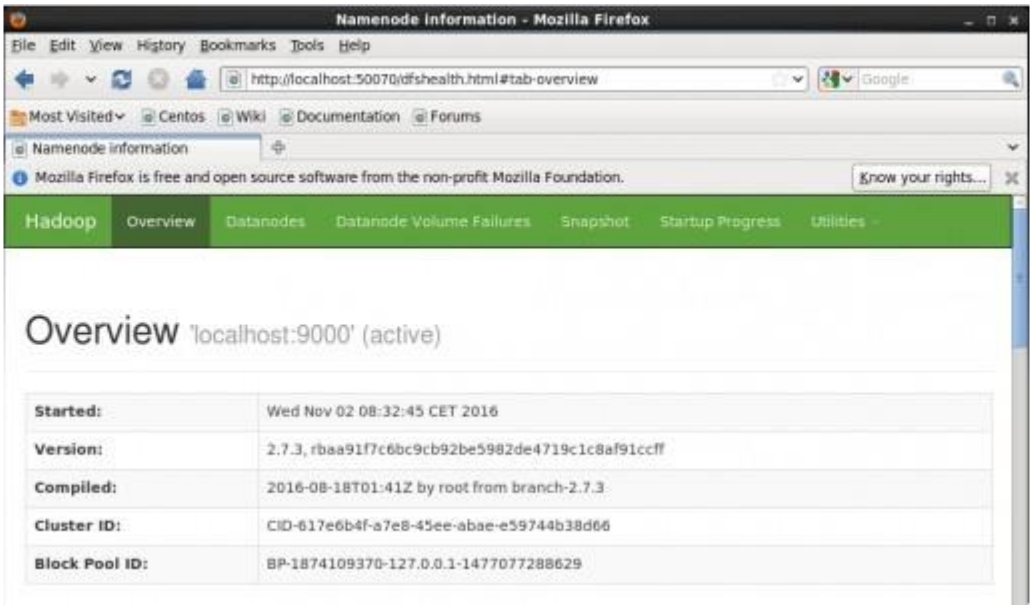
Command: jps


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

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

Fig: Hadoop Installation – Checking Daemons

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


Namenode information - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:50070/dfshealth.html#tab-overview
Google
Most Visited
Centos
Wiki
Documentation
Forums
Namenode information
Mozilla Firefox is free and open source software from the non-profit Mozilla Foundation.
Know your rights...
Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities

Overview 'localhost:9000' (active)

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

Fig: Hadoop Installation – Starting WebUI

Congratulations, you have successfully installed a single node Hadoop cluster.

RESULT

Thus, the installation of a Hadoop single node cluster and run simple applications like word count.

Ex no. 8 Install Hadoop Single Node Cluster and Run Simple Applications like Word Count

AIM:

To Create and Execute Your First Container Using Docker and Run a container from Docker Hub.

PROCEDURE:

Run a container from docker hub

Run docker-h,

```
$docker-h
Flag short hand-hhas been deprecated ,please use—help Usage:

Docker [OPTIONS]COMMAND

A self-sufficient run time for containers


...
Management Commands:
builder      Manage builds
config       Manage Docker configs
container Manage containers engine
             Manage the docker engine
image        Manage images
network        Manage networks
node         Manage Swarm nodes
plug in      Manage plug ins
secret       Manage Docker secrets
service      Manage services
stack        Manage Docker stacks
swarm         Manage Swarm
system        Manage Docker
trust        Manage trust on Docker images
volume        Manage volumes
```

The Docker command line can be used to manage several features of the Docker Engine. In this lab, we will mainly focus on the container command.

If podman is installed, you can run the alternative command for comparison.

```
Sudo podman-h
```

You can additionally review the version of your Docker installation, docker version

```
Client:
 Version:     19.03.6
  ...

Server :Docker Engine-Community
 Engine
 Version:    19.03.5
    ...

Sudo podman version--events-backend=none
Version:      2.1.1
APIVersion:2.0.0GoV
ersion:go1.15.2
Built:
              ThuJan100:00:001970O
S/Arch:        linux /amd64
```

Step1:Run your first container

We are going to use the Docker CLI to run our first container. Open

alterminal on your local computer

Run docker container run-tubuntu top

Use the docker container run command to run a container with the ubuntu image using The top command. The-tflags allocate a pseudo-TTY which we need for the top to work correctly.

```
$docker container run-it ubuntu top Unable
to find image 'ubuntu: latest 'locally latest:
Pulling from library/ubuntuaafe6b5e13de:
Pull complete 0a2b43a72660: Pull
complete18bdd1e546d2:Pullcomplete819834
2c3e05:
Pull completef56970a44fd4:Pull complete
Digest:sha256:f3a61450ae43896c4332bda5e78b453f4a93179045f20c8181043b26b5e79028
                                                                        Status:
Downloaded newer image for ubuntu:latest
```

The docker run command will result first in a docker pull to download the ubuntu image onto your host Once it is downloaded, it will start the container. The output for the running container should look like this:

```
top-20:32:46up3days, 17:40,0users,loadaverage:0.00,0.01,0.00
Tasks:1total, 1running, 0sleeping, 0 stopped, 0 zombie
%Cpu(s):0.0us,0.1 sy,0.0ni,99.9id,0.0wa,0.0 hi,0.0si,0.0 st

KiBMem:2046768total,173308free, 117248used,1756212buff/cache
```

```
PIDUSER     PRNI     VIRTRESSHR S%CPU
   %MEMTIME+COMMAND1root2003663630722640R0.30.20:00.
   04top
```

Inspect the container with docker container exec

The docker container exec command is a way to "enter" a running container's namespaces with a new process.

Open a new terminal .On cognitive class. Ai ,select Terminal>New Terminal.

Using play-with-docker.com, to open a new terminal connected to node1, click "Add New Instance" on the left hand side,then sshfromnode2into node1 using the IP that is listedby'node1'.Forexample:

```
[node2](local)root@192.168.0.17~
$ssh192.168.0.18
[node1](local)root@192.168.0.18~
$
```

In the new terminal, use the docker container ls command to get the ID of the running container you just created.

```
$docker containerls
CONTAINERID     IMAGE         COMMAND           CREATED        STATUS                PORTS
NAMES
b3ad2a23fab3         ubuntu               "top"              29minutesago        Up29minutes
goofy_nobel

$ docker container exec -it b3ad2a23fab3
bashroot@b3ad2a23fab3:/#
```

And Voila! We just used the docker container exec command to "enter" our container's namespaces with our bash process. Using docker container exec with bash is a common pattern to inspect a docker container.

Notice the change in the prefix of your terminal. e.g. root@b3ad2a23fab3:/. This is an indication that we are running bash "inside" of our container.

From the same termina,runps-ef to inspect the running

```
processes.root@b3ad2a23fab3:/#ps-ef
UID      PID  PPIDCSTIME    TTY      TIMECMD
 root       1    0020:34?           00:00:00top
 root      17     00 21:06 ?          00:00:00bash
 root      27    17021:14?            00:00:00ps -ef
```

You should see only the top process, bash process and our ps

```
process.root@b3ad2a23fab3:/#exit
exit
$ps-ef
```

```
# Lots of processes
!docker ps -a

Docker rm<CONTAINERID>
```

Step2:Run Multiple Containers

Explore the Docker Hub

The Docker Hub is the public central reg is try for Docker images, which contains community and official images. Run an Ng in x server

Let's run a container using the official Ng in x image from the Docker Hub.

```
$ docker container run --detach --publish 8080:80 --name ng in x ng
in x Unable to find image 'ng in x :latest'  locally
latest:Pullingfromlibrary/nginx36a46ebd501
9:Pullcomplete57168433389f:
Pullcomplete332ec8285c50:Pullcomplete
Digest:sha256:c15f1fb8fd55c60c72f940a76da76a5fccce2fefa0dd9b17967b9e40b035531
Status :Downloaded newer image for ng in x:latest

5e1bf0e6b926bd73a66f98b3cbe23d04189c16a43d55dd46b8486359f6fdf048
```

Ng in x is a light weight web server .You can access it on port 8080on your local host.

Access the ng in x server on localhost:8080.

Curl localhost:8080

Will return the HTML home page of Ng in x,

```
<!DOCTYPEhtml>
<html>
<head>


<title>Welcome to ng in x!</title>
<style>
  body
  {
    width:35em;margin:
    0 auto;
    font-family:Tahoma,Verdana,Arial,sans-serif;
  }

</style>
</head>
<body>
```

```
<h1>Welcome to ng in x!</h1>
```

If you are using play-with-docker, look for the 8080 link near the top of the page, or if you run a Docker client with access to a local browser,

Run a mongo DB server

Now, run a mongo DB server. We will use the official mongo DB image from the Docker Hub. Instead of using the latest tag (which is the default if no tag is specified), we will use a specific version of the mongo image.

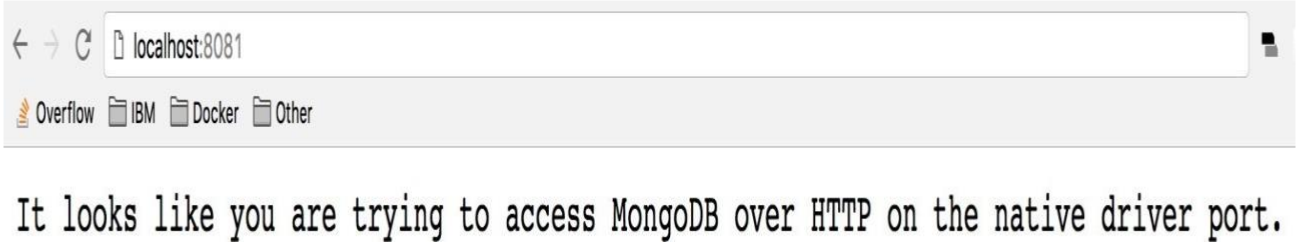
```
$ docker container run --detach --publish 8081:27017 --name mongo mongo:4.4
Unable to find imagemongo:4.4 locally
4.4: Pulling from
library/mongod13d02fa248
d: Already
existsbc8e2652ce92: Pull
complete3cc856886986:
Pull
completec319e9ec4517:
Pull
completeb4cbf8808f94:
Pull
completecb98a53e6676:
Pull
completef0485050cd8a:
Pull
completeac36cdc414b3:
Pull
complete61814e3c487b:
Pull
complete523a9f1da6b9:
Pull
complete3b4beaef77a2:Pul
lcomplete
Digest:sha256:d13c897516e497e898c229e2467f4953314b63e48d4990d3215d876ef9d1fc7c
                                                                                  S
tatus:Downloaded newer image for
mongo:4.4d8f614a4969fb1229f538e171850512f10f490cb1a96fca27e4aa89ac082eba5
```

Access localhost:8081toseesome outputfrom

mongo.curllocalhost:8081

whichwillreturnawarningfromMongoDB,

It looks like you are trying to access Mongo DB over HTTP on the native driver port. If you are using play-with-docker, look for the 8080 link near the top of the page.



It looks like you are trying to access MongoDB over HTTP on the native driver port.

Check your running containers with docker container ls

$docker container ls

```
CONTAINER ID IMAGE COMMAND CREATED  STATUS  PORTS         NAMES

d6777df89fea    ng in x  "ng inx-g'daemon..."     Less than a second ago
                         Up2seconds0.0.0.0:8080-
>80/tcp     nginx
ead80a0db505     mongo    "docker-entry               17secondsago      Up19seconds      0.0.0.0:8081-
>27017/tcp     point..."mongo
af549dccd5cf    ubuntu     "top"    5minutesago      Up5minutes      priceless_kepler
```

Step3:Clean Up

First get a list of the containers running using docker container ls.

```
$docker container ls
CONTAINERID     IMAGE         COMMAND          CREATED        STATUS
                PORTSNAMES
d6777df89fea       nginx           "nginx-g'daemon..." 3 minutes ago
                                   Up3minutes0.0.0.0:8080-
>80/tcp      nginx
ead80a0db505       mongo           "docker-entry
                   point..."3minutesagoUp3minutes0.0.0.0:8081-
                   >27017/tcpmongo
af549dccd5cf      ubuntu          "top"              8 minutes
                  agoUp8 minutes priceless_kepler
```

Next, run docker container stop [container id] for each container in the list. You can also use the names of the containers that you specified before.

```
$ docker container
stopd67ead af5d67
e
a
d

a
f
5
```

Remove the stopped containers

docker system prune is a really handy command to clean up your system. It will remove any stopped containers, unused volumes and networks ,and dangling images.

```
$ docker system
prune
WARNING!Thi
swill remove:
      - All stopped containers
      - All volumes not used by atleast one container
      - All networks not used by atleast one container
      - all dangling
images Are you sure
you want to continue?
[y/N] y Deleted



      - Containers:
7872fd96ea4695795c41150a06067d605f69702dbcb9ce49492c9029f0e1b44b

60abd5ee65b1e2732ddc02b971a86e22de1c1c446dab165462a08b037ef78
35c31617fdd8e5f584c51ce182757e24a1c9620257027665c20be75aa3ab65
91740

Total reclaimedspace:12B
```

Result:

Thus, Creating and Executing of our First Container Using Docker and to Run a container from Docker has been successfully executed

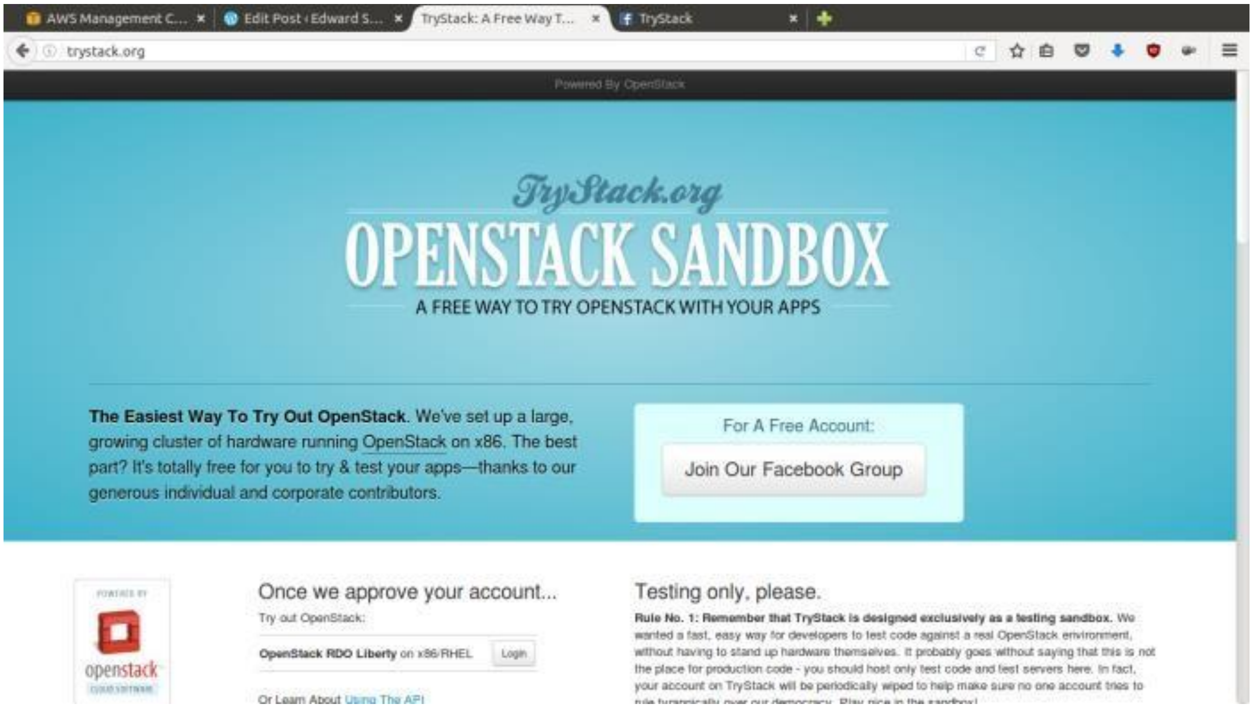
Ex. No 9 :Find a procedure to launch virtual machine using Try Stack

Aim : To Find a procedure to launch virtual machine using Try Stack

Procedure :

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

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



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

Once we approve your account...

Try out OpenStack:

OpenStack RDO Liberty on x86/RHEL

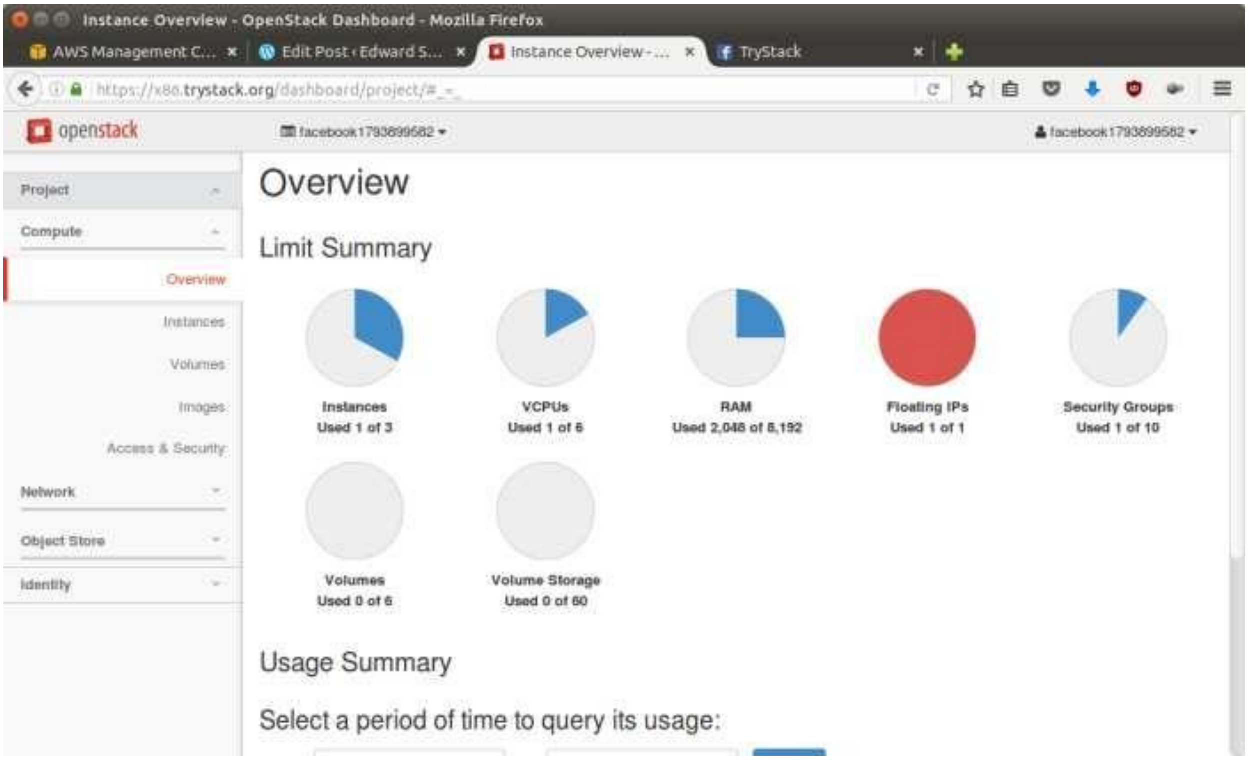
Or Learn About Using The API

Testing only, please.

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

Try Stack.org Homepage

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



Overview

Limit Summary

Instances
Used 1 of 3

VCPUs
Used 1 of 6

RAM
Used 2,048 of 8,192

Floating IPs
Used 1 of 1

Security Groups
Used 1 of 10

Volumes
Used 0 of 6

Volume Storage
Used 0 of 60

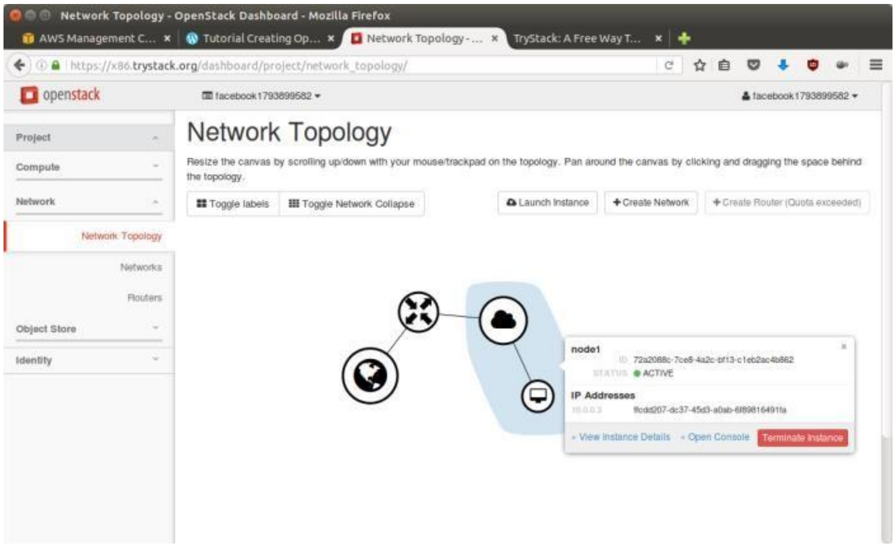
Usage Summary

Select a period of time to query its usage:

Open Stack Compute Dashboard

Overview: What we will do?

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



Network Topology

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

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

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

Step 1: Create Network

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

1. Go to **Network > Networks** and then click **Create Network.**
2. In **Network** tab, fill **Network Name** for example internal and then click **Next.**
3. In **Subnet** tab,
   1. Fill **Network Address** with appropriate CIDR, for example 192.168.1.0/24. Use private network CIDR block as the best practice.
   2. Select **IP Version** with appropriate IP version, in this caseIPv4.
   3. Click **Next.**
4. In **Subnet Details** tab, fill **DNS Name Servers** with 8.8.8.8 (Google DNS) and then click **Create.**

Step 2: Create Instance

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

1. Go to **Compute > Instances** and then click **Launch Instance.**
2. In **Details** tab,
   1. Fill **Instance Name**, for example Ubuntu1.
   2. Select **Flavor**, for examplem1.medium.
   3. Fill **Instance Count** with**1.**
   4. Select **Instance Boot Source** with **Boot from Image.**
   5. Select **Image Name** with **Ubuntu 14.04 amd64 (243.7 MB)** if you want install Ubuntu 14.04 in your virtual machine.
3. In **Access & Security** tab,
   1. Click [**+**] button of **Key Pair** to import key pair. This key pair is a public and private key that we will use to connect to the instance from our machine.
   2. In **Import Key Pair** dialog,
      1. Fill **Key Pair Name** with your machine name (for example Edward-Key).
      2. Fill **Public Key** with your **SSH public key** (usually is in ~/.ssh/id_rsa.pub). See description in Import Key Pair dialog box for more information. If you are using Windows, you can use **Putty gen** to generate key pair.
      3. Click **Import key pair.**
   3. In **Security Groups**, mark/check **default.**
4. In **Networking** tab,
   1. In **Selected Networks**, select network that have been created in Step 1, for example internal.

5. Click **Launch.**
6. If you want to create multiple instances, you can repeat step 1-5. I created one more instance with instance name Ubuntu2.

Step 3: Create Router

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

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

Step 4: Configure Floating IP Address

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

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

Step 5: Configure Access & Security

Open Stack has a feature like a firewall. It can white list/blacklist your in/out connection. It is called *Security Group*.

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

Step 6: SSH to Your Instance

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

RESULT:

Thus, a procedure to launch a virtual machine using Try Stack was completed successfully.

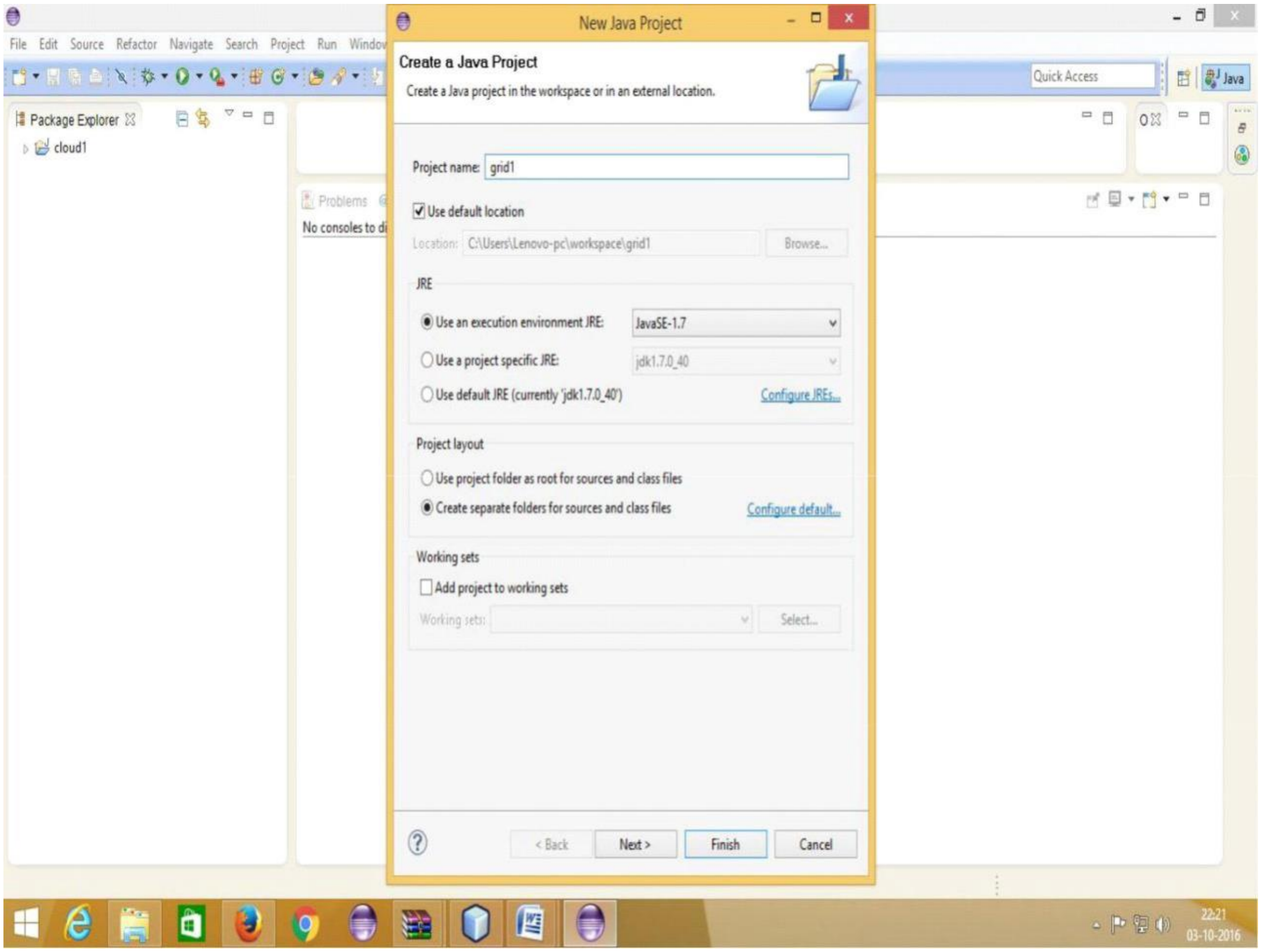
Ex. No: 10 Simulate a cloud scenario using Grid Sim

Aim:

To install Grid Sim and implement the java code for Grid Sim Environment to create 'n' users and datacenters with the Grid lets running in each datacenter.

Procedure:

1. Open the eclipse tool using the path **E:\cloud grid\eclipse-SDK-4.2.2-win32x86_64\eclipse\eclipse.exe\**
2. Click on File Menu → New → Java Project
3. Give name to the project (for instance, grid1) and click on **Next** Button



New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: grid1

☑ Use default location

Location: C:\Users\Lenovo-pc\workspace\grid1

JRE

● Use an execution environment JRE: JavaSE-1.7

○ Use a project specific JRE: jdk1.7.0_40

○ Use default JRE (currently 'jdk1.7.0_40')

Project layout

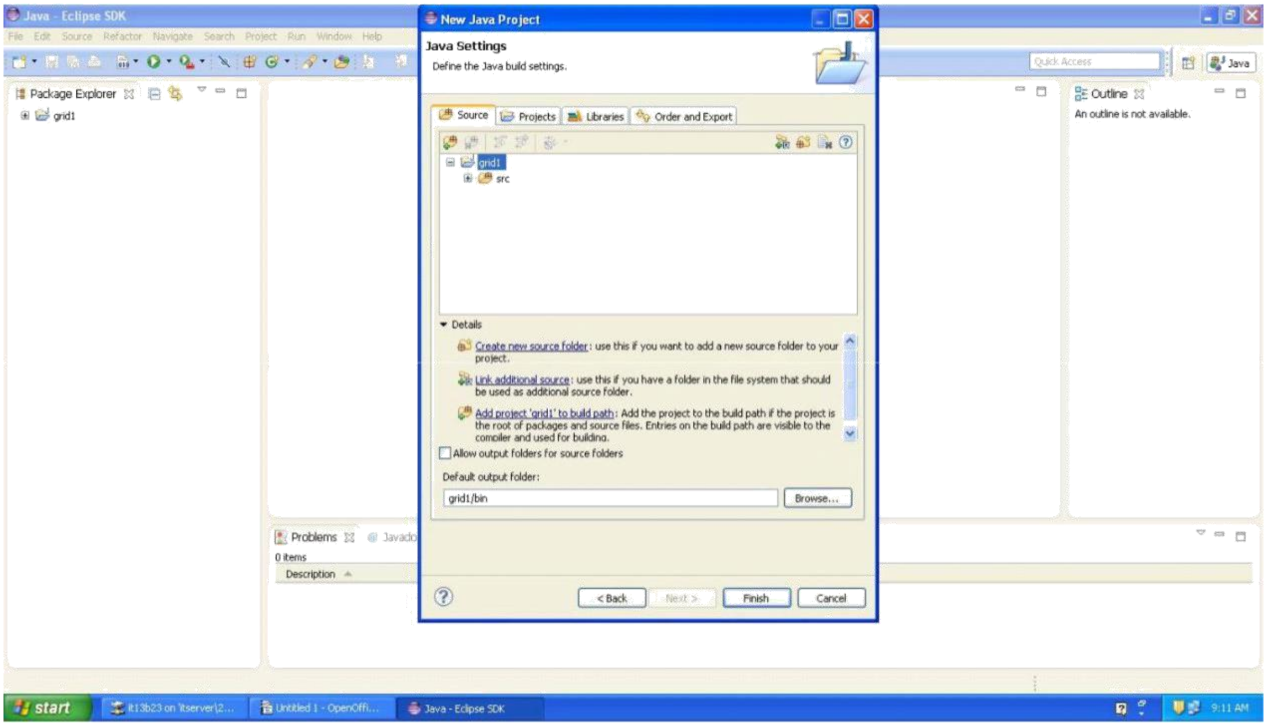
○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

Working sets

□ Add project to working sets

Working sets:



Java Settings

Define the Java build settings.

Create new source folder: use this if you want to add a new source folder to your project.

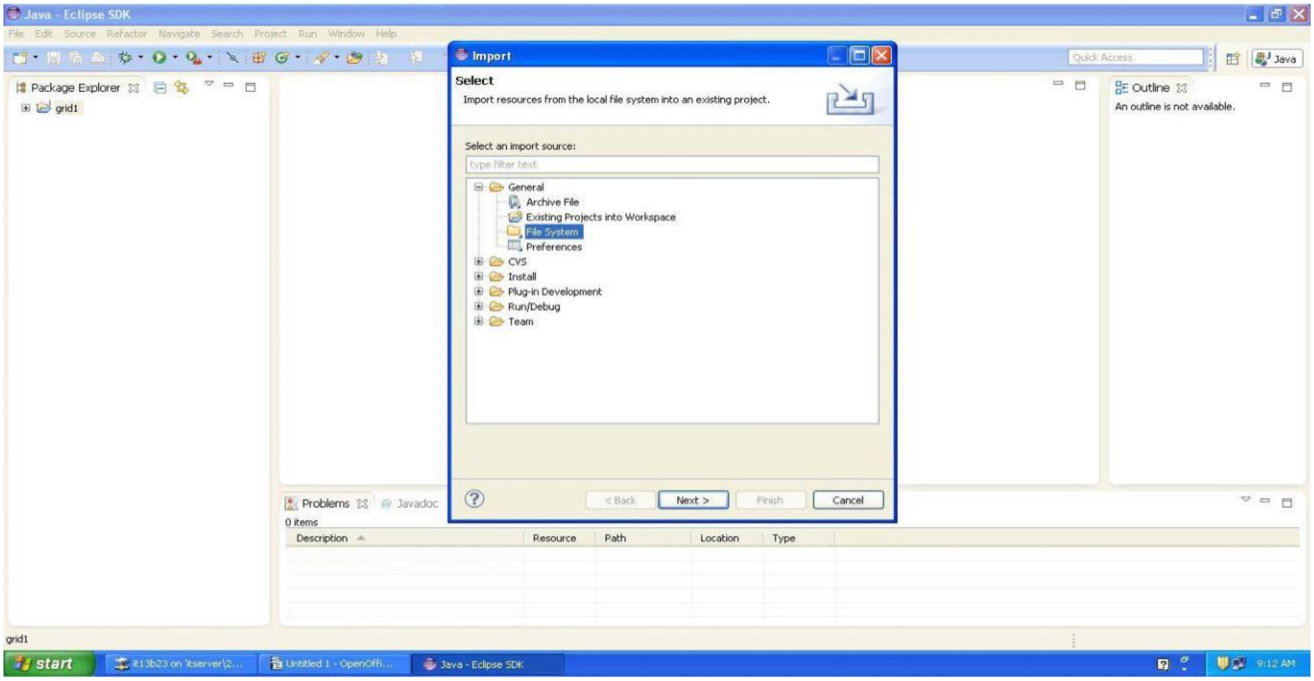
Link additional source: use this if you have a folder in the file system that should be used as additional source folder.

Add project 'grid1' to build path: Add the project to the build path if the project is the root of packages and source files. Entries on the build path are visible to the compiler and used for building.

☐ Allow output folders for source folders

Default output folder: grid1/bin

4. Click **Finish** Button
5. Right Click on the project in project explorer --> click **Import**



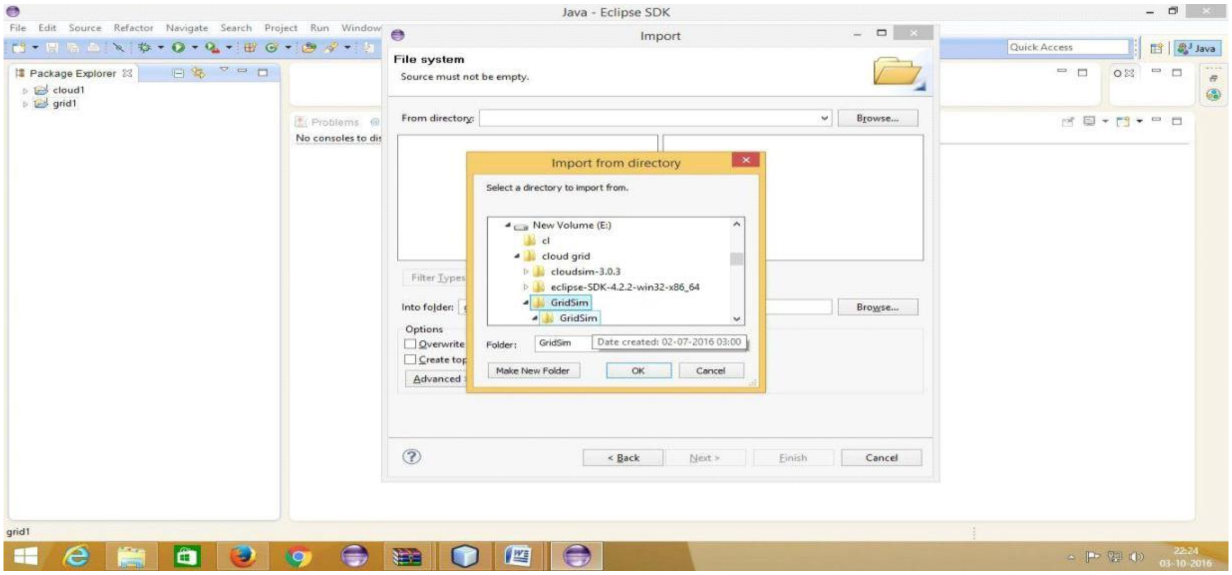
Select

Import resources from the local file system into an existing project.

Select an import source:

6.Select General → File System and click Next

7.Click **Browse** and browse the grid Sim folder and Select the folder.



File system

Source must not be empty.

No consoles to dis

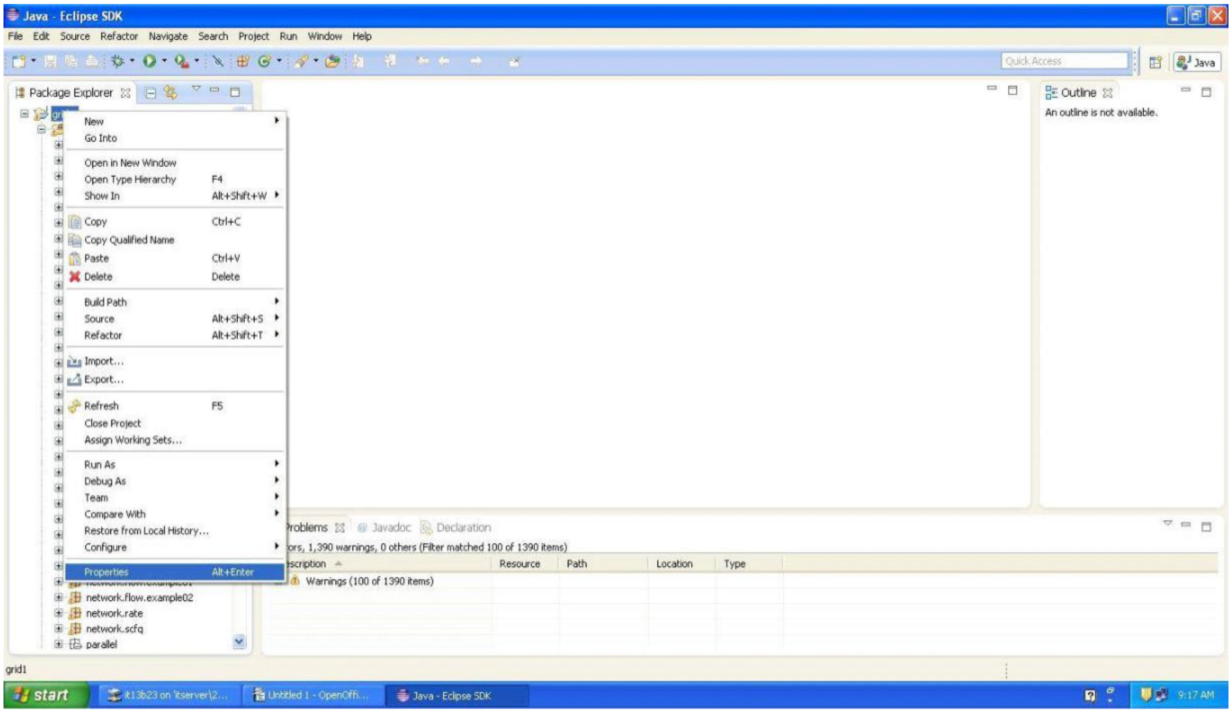
Select a directory to import from.

Folder: GridSim

Date created: 02-07-2016 03:00

22:24 03-10-2016

8. Click OK ,Then Select the Grid Sim folder
9. Click **Finish** Button-->Click **yes to All** to import
10. Right Click the project in project explorer and click **properties**



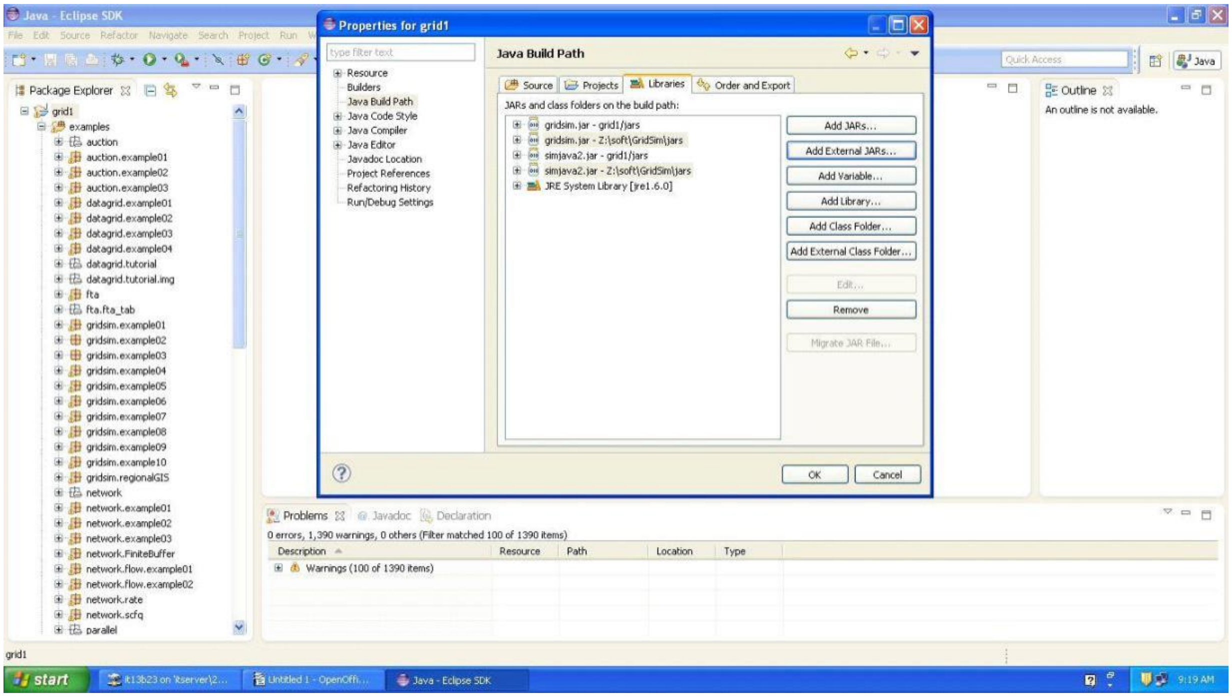
An outline is not available.

ors, 1,390 warnings, 0 others (Filter matched 100 of 1390 items)

Warnings (100 of 1390 items)

11.Click on **Java Build Path** and then **libraries** tab
12.Click on **Add External Jars**

13. Browse and Select all the jar files within gridSim folder → jars



Java Build Path

JARs and class folders on the build path:

- gridsim.jar - grid1/jars
- gridsim.jar - Z:\soft\GridSim\jars
- simjava2.jar - grid1/jars
- simjava2.jar - Z:\soft\GridSim\jars
- JRE System Library [jre1.6.0]

```
0 errors, 1,390 warnings, 0 others (Filter matched 100 of 1390 items)
Description    Resource    Path    Location    Type
  Warnings (100 of 1390 items)
```

14. Expand the project and Navigate into grid1 → examples → gridsim.example01 →

example1
15. open the source code by double clicking the example code

Example code:

Question:

Implement the following scenario in a grid environment. Create 6 users and 2 datacenters with 25 gridlets running in each datacenter. Share the workload and maintain the load balancing in the 2 datacenters./*note the bolded portion is the code to be added in the program in example 2*/

```
package gridsim.example02;

/*
        *   Author Anthony Su listio
        *   Date: April 2003
        *   Description: A simple program to demonstrate of how to use Grid Sim package.
        *   This example shows how to create one or more Grid users.
        *   A Grid user contains one or more Grid lets.
```

```
               *    Therefore, this example also shows how to create Grid lets with
               *    and without using Grid Sim Random class.
*
        *  NOTE: The values used from this example are taken from the Grid Sim paper.
        *  http://www.gridbus.org/gridsim/
        *  $Id: Example2.java,v 1.4 2003/05/19 13:17:49 anthony Exp $
*/
import  java. util.*;
import gridsim.*;
/**
        *   This class shows how to create one or more grid users. In addition, the
        *   creation of Grid lets also discussed.
*/
class Example2
{
  /**
        *   Main function to run this
            example */
  public static void main(String[] args)
  {
     System.out.println("Starting example of how to create Grid users");
     System.out.println();

     try
     {

// Creates a list of Gridlets

        Grid let List list = create Grid let(); System. out. println("Creating " +
        list. size() + " Grid lets"); Resource User List user List = create Grid
        User(list); System .out. println("Creating " + user List. size() + " Grid
        users");

        // print the Grid lets
```

```
            Print Grid let List(list);

            System.out.println("Finish the example");
        }
        catch (Exception e)
        {
            e.printStackTrace();
            System.out.println("Unwanted error happens");
        }
    }
    /**
          *  A Grid user has many Grid lets or jobs to be processed.
          *  This method will show you how to create Grid lets with and without
          *  Grid Sim Random class.
          *  @return a Grid let List object
     */
    private static Grid let List create Grid let()
    {
   //  Creates a container to store Grid lets

         Grid let List list = new Grid let List();

   //  We create three Grid lets or jobs/tasks manually without the help
   //  of Grid Sim Random
        int id = 0;
        double length = 3500.0;
        long file_size = 300;
        long output_size = 300;
        Grid let gridlet1 = new Grid let(id, length, file_size, output_size); id++;

  Grid let gridlet2 = new Grid let(id, 5000, 500, 500);

        id++;
        Grid let gridlet3 = new Grid let(id, 9000, 900, 900);
```

```
 //  Store the Grid lets into a list
     list.add(gridlet1);
     list.add(gridlet2);
     list.add(gridlet3);
 //  We create 5 Gridlets with the help of Grid SimRandom and
 //  Grid Sim Standard PE class
     Random random = new Random();
 //  sets the PE MIPS Rating
 //  Grid SimStandardPE.setRating(100);
 //  creates 22 Gridlets
     int count = 22;
     double min_range = 0.10;
     double max_range = 0.50;
     for (int i = 1; i < count+1; i++)
     {
 //  the Grid let length determines from random values and the
 //  current MIPS Rating for a PE
        length = Grid Sim Standard PE.toMIs(random.nextDouble()*output_size);
 //  determines the Grid let file size that varies within the range
 //  100 + (10% to 50%)
        file_size = (long) Grid Sim Random.real(100, min_range, max_range,
                          random.nextDouble());
       //  determines the Grid let output size that varies within the range
       //  250 + (10% to 50%)
        output_size = (long) Grid SimRandom.real(250, min_range, max_range,
                          random.nextDouble());
        // creates a new Grid let object

Grid let grid let = new Grid let(id + i, length, file_size,
                          output_size);
//     add the Grid let into a list
       list.add(grid let);
      }
```

```
    return list;
}

/**
*    Creates Grid users. In this example, we create 3 users. Then assign
*    these users to Grid lets.
*    @return a list of Grid users
 */
private static Resource User List create GridUser(GridletList list)
{
    Resource User List user List = new ResourceUserList();

    userList.add(0);       // user ID starts from 0
    userList.add(1);
    userList.add(2);
    userList.add(3);
    userList.add(4);
    userList.add(5);
    int userSize = userList.size();
    int gridletSize = list.size();
    int id = 0;

//    assign user ID to particular Grid lets
  for (int i = 0; i < gridletSize; i++)
    {
        if (i != 0 && i % userSize == 0)
            id++;
        ( (Gridlet) list.get(i) ).setUserID(id);
    }

return userList;
}

private static void printGridletList(GridletList list)
```

```
  {
     int size = list.size();
     Gridlet gridlet;

     String indent = "         ";
     System.out.println();
     System.out.println("Gridlet ID" + indent + "User ID" + indent +
           "length" + indent + " file size" + indent +
           "output size");

     for (int i = 0; i < size; i++)
     {
        gridlet = (Gridlet) list.get(i);
        System.out.println(indent + gridlet.getGridletID() + indent + indent
              + indent + gridlet.getUserID() + indent + indent + (int)
              gridlet.getGridletLength() + indent + indent +
              (int) gridlet.getGridletFileSize() + indent + indent +
              (int) gridlet.getGridletOutputSize() );
     }
  }

} // end class
```

Output:

```
Starting example of how to create Grid users

Creating 25 Gridlets
   Creating 6 Grid users
Gridlet ID          User ID     length        file size  output size
    0          0         3500         300         300
    1          0         5000         500         500
    2          0         9000         900         900
```

```
3        0    24798      91       253
4        0    4921      136       362
5        0    13917      112        361
6        1    22766      103        244
7        1    22048      129        354
8        1    12355      140        348
9        1    4891      136       324

10        1    26116       108       241
11        1    7467       144       334
12        2    17698       111       301
13        2    2130       129       300
14        2    16183       115       253
15        2    7842       128       354
16        2    31062       115       238
17        2    3469       135       225
18        3   17362        133       288
19        3    2061       135       346
20        3  13402         113       368
21        3      30585      106       348
22        3      22877      134       318
23        3      5026      94       271
24        4      14461      109       308
```

Result:

Thus, grid is simulated in Grid Sim environment and the program is executed successfully.