

<b>EX. No 1</b>	<b>INSTALL VIRTUALBOX/VMWARE WORKSTATION WITH DIFFERENT FLAVOURS OF LINUX OR WINDOWS OS ON TOP OF WINDOWS7 OR 8.</b>
-----------------	--

### **AIM:**

To find the procedure to run the virtual machine of different configuration.  
And also check how many virtual machines can be utilized at particular time.

### **ALGORITHM:**

**Step 1:** Start the VMWare workstation 9.0 icon on your desktop.

**Step 2:** Go to file and create new virtual machine.

**Step 3:** Home page of VMWare wizard will get displayed.

**Step 4:** Select the option create new virtual machine.

**Step 5:** Create a new virtual machine by following the step by step options.

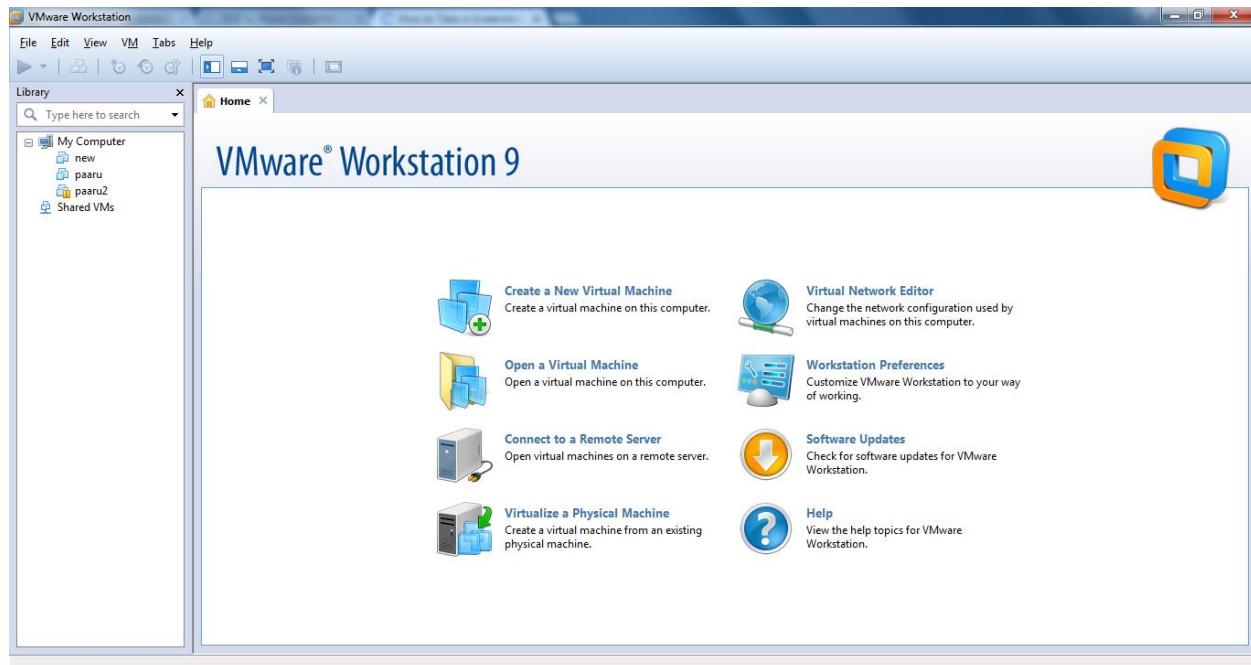
**Step 6:** New virtual machine will be created.

### **VIRTUAL MACHINE:**

A virtual machine is a software computer that, like a physical machine, runs an operating system and applications. A virtual machine uses the physical resources of the physical machine on which it runs, which is called the host system. Virtual machines have virtual devices that provide the same functionality as physical hardware, but with the additional benefits of portability, manageability, and security. A virtual machine has an operating system and virtual resources that you manage in much the same way that you manage a physical computer.

## **INSTALLATION OF VM:**

Create a new virtual machine in Workstation by using the New Virtual Machine wizard.



### **1. SELECTING A VIRTUAL MACHINE CONFIGURATION**

When you start the New Virtual Machine wizard, the wizard prompts you to select a typical or custom configuration.

#### **Typical Configuration**

If you select a typical configuration, you must specify or accept defaults for a few basic virtual machine settings.

- How you want to install the guest operating system.
- A name for the virtual machine and a location for the virtual machine files.
- The size of the virtual disk and whether to split the disk into multiple virtual disk files.
- Whether to customize specific hardware settings, including memory allocation, number of virtual processors, and network connection type.

#### **Custom Configuration**

Select a custom configuration if you need to perform any of the following hardware customizations.

- Create a virtual machine that has a different Workstation version than the default hardware compatibility setting.
- Select the I/O controller type for the SCSI controller.

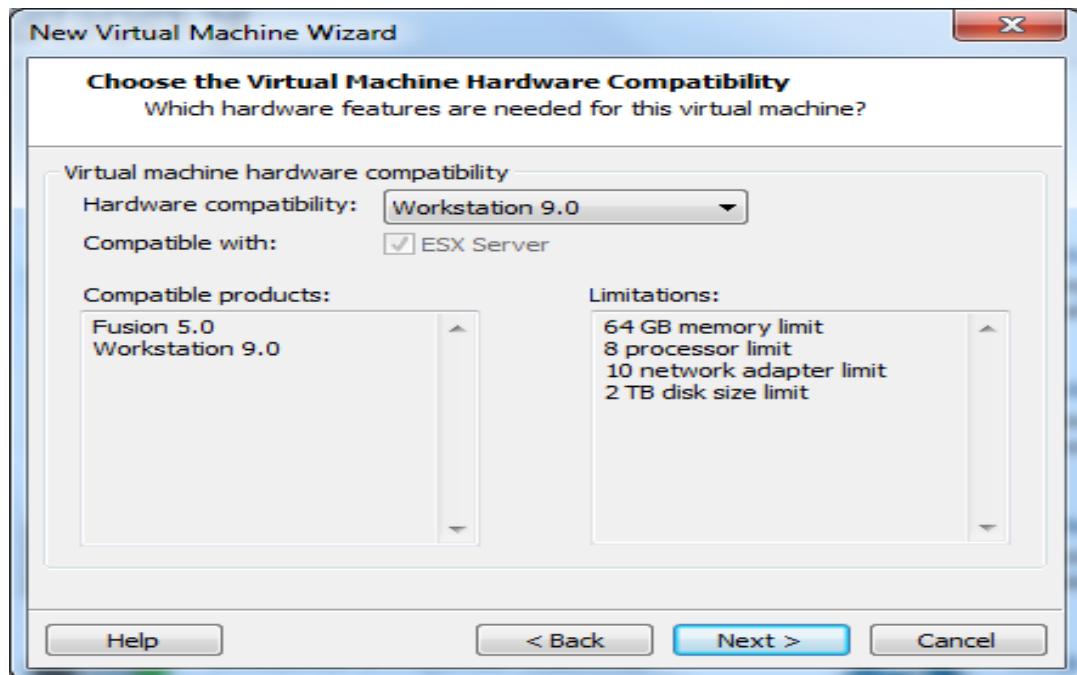
- Select the virtual disk device type.
- Configure a physical disk or an existing virtual disk instead of create a new virtual disk.
- Allocate all virtual disk space rather than let disk space gradually grow to the maximum disk size.



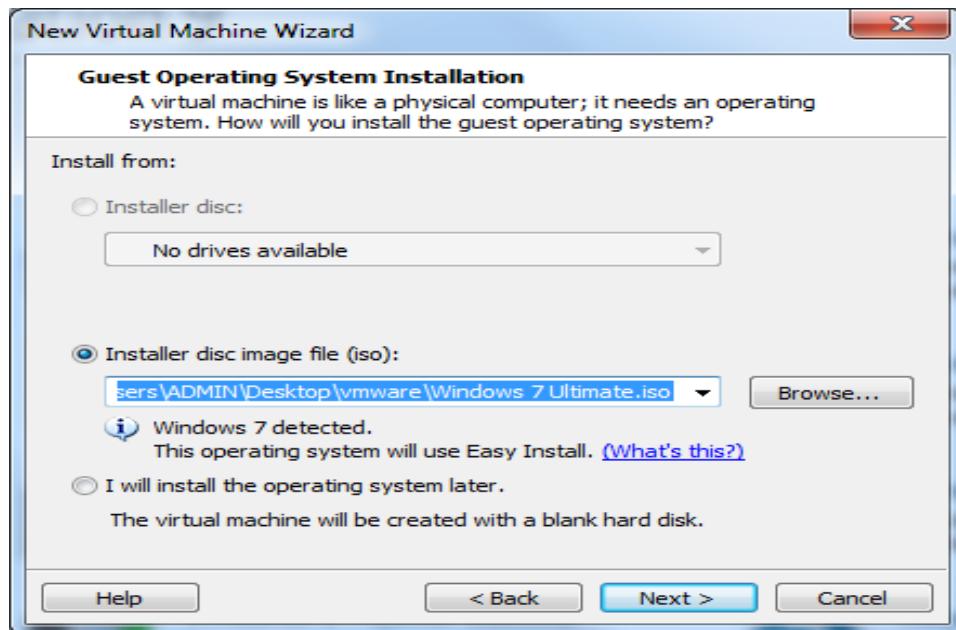
## **2. VIRTUAL MACHINE HARDWARE COMPATIBILITY**

To select a custom configuration, the New Virtual Machine wizard prompts you to select a hardware compatibility setting for the virtual machine. When you select a hardware compatibility setting, a list of the VMware products and versions that are compatible with your selection appears. Limitations and features that are not available for your selection are also listed. If a feature compatibility check box is available for your selection, you can select that check box to see a list of the additional limitations.

To deploy virtual machines to run on a different VMware product, you might need to select a hardware compatibility setting that is compatible with that product.



The New Virtual Machine prompts you to select the source media for the operating system that will run inside the virtual machine. You can specify an installer disc inserted in a physical drive, an ISO image file, or you can instruct the New Virtual Machine wizard to create a virtual machine that has a blank hard disk. If you select an installer disc or an ISO image file and the operating system supports Easy install, the guest operating system installation is automated and VMware Tools is installed. If the installer disc or ISO image file contains a product key number and is already set up to perform an unattended installation, the only benefit of using Easy Install is the automatic installation of VMware Tools.



When the New Virtual Wizard detects an operating system that supports Easy Install, the wizard prompts you for information about the guest operating system. After the virtual machine is created, the guest operating system installation is automated and VMware Tools is installed. For Windows guest operating systems, you must provide the following Easy Install information.

### **3. EASY INSTALL INFORMATION FOR WINDOWS GUESTS**

**Windows product key (Optional)** Type a product key unless the installation media contains a volume license product key. If you provide a product key here, you are not prompted to provide a product key when you install the guest operating system.

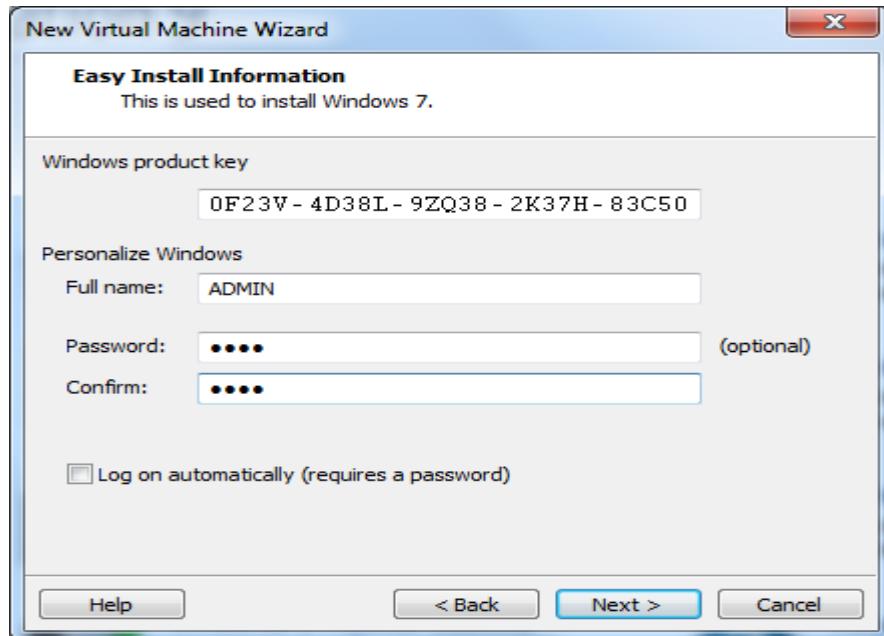
**Version of Windows to install** For Windows Vista, Windows 7, and Windows Server 2008 guest operating systems, select the operating system edition.

**Full name** The name to use to register the guest operating system. Do not use the name Administrator or Guest. If you use one of these names, you must enter a different name when you install the guest operating system.

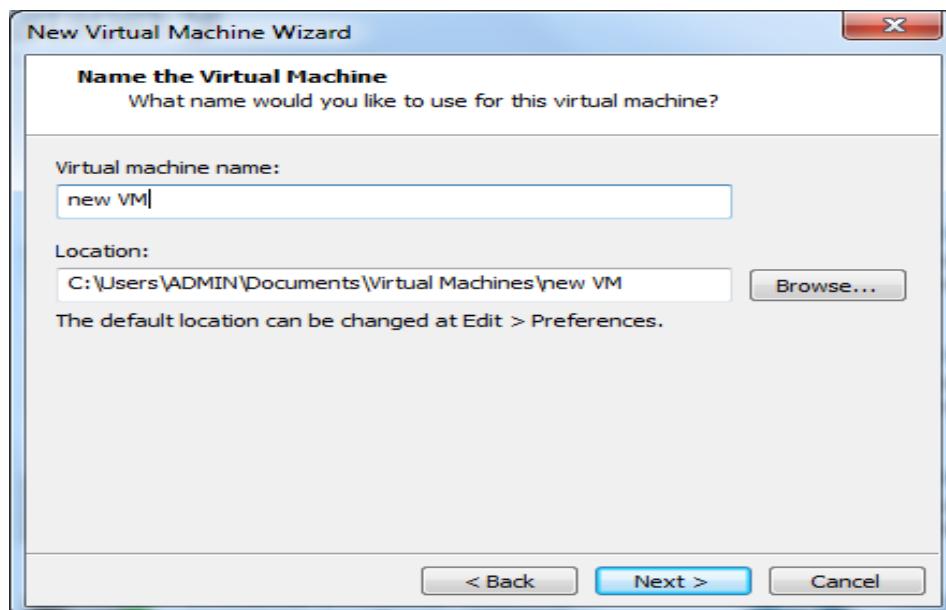
**Password** The password to use for an account with Administrator permissions on Windows operating systems other than Windows 2000. On Windows 2000, this is the password for the Administrator account. On Windows XP Home, an Administrator account without a password is created and you are automatically logged in to the guest operating system.

## Log on automatically (requires a password)

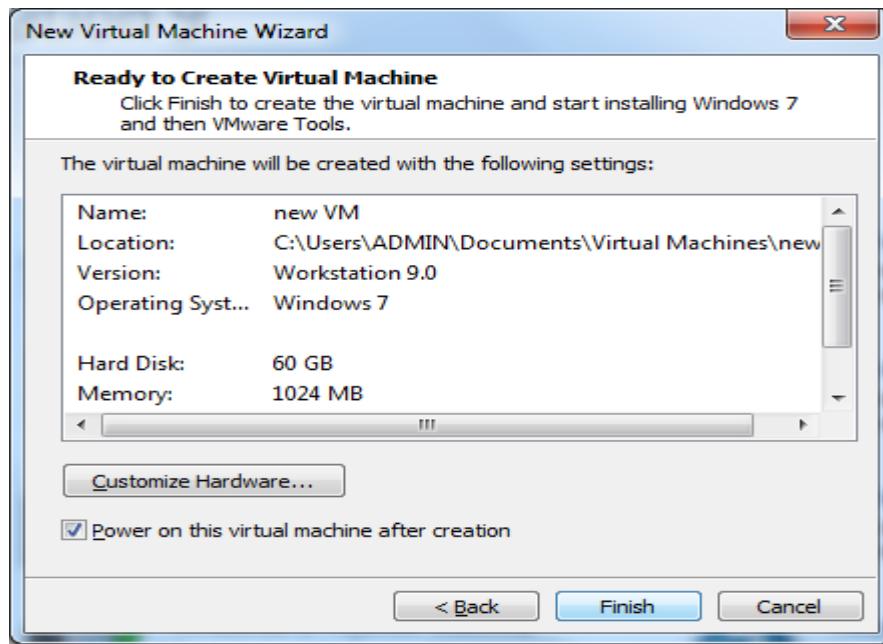
Save your login credentials and bypass the login dialog box when you power on the virtual machine. You must enter a name and password to use this feature.



Then next step is to create name for the new virtual machine and location should set where the virtual machine is to be stored.

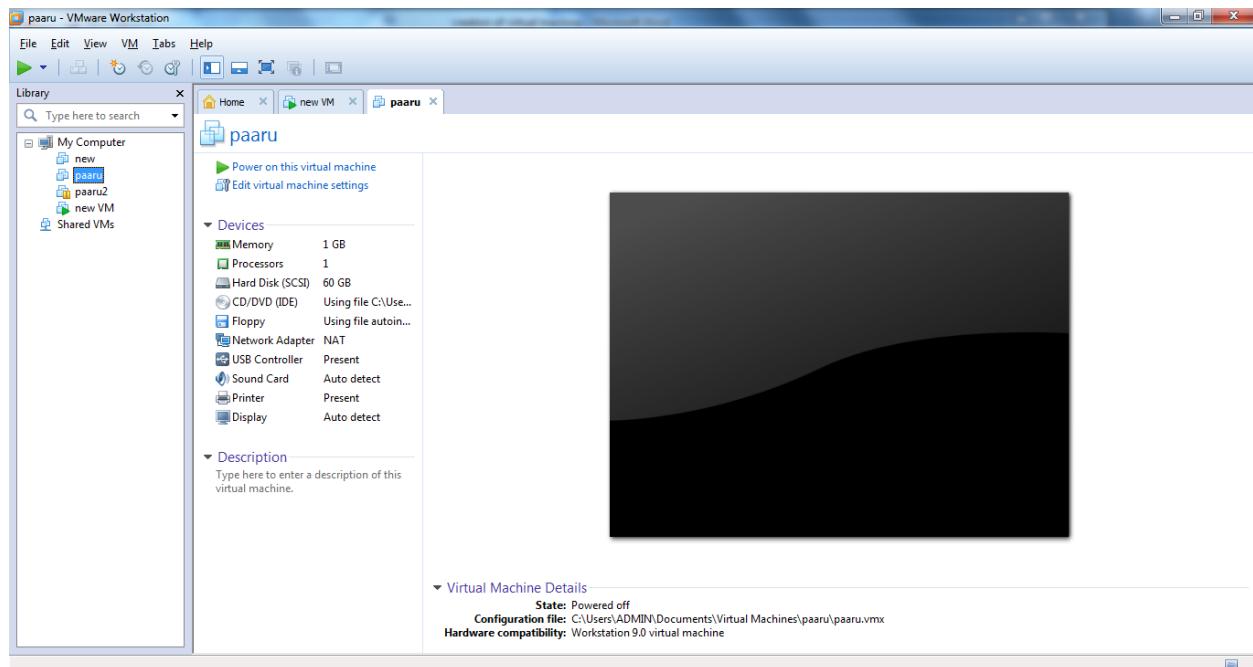


After completion of these steps the newly created virtual machine is ready to work with the specified system configuration.



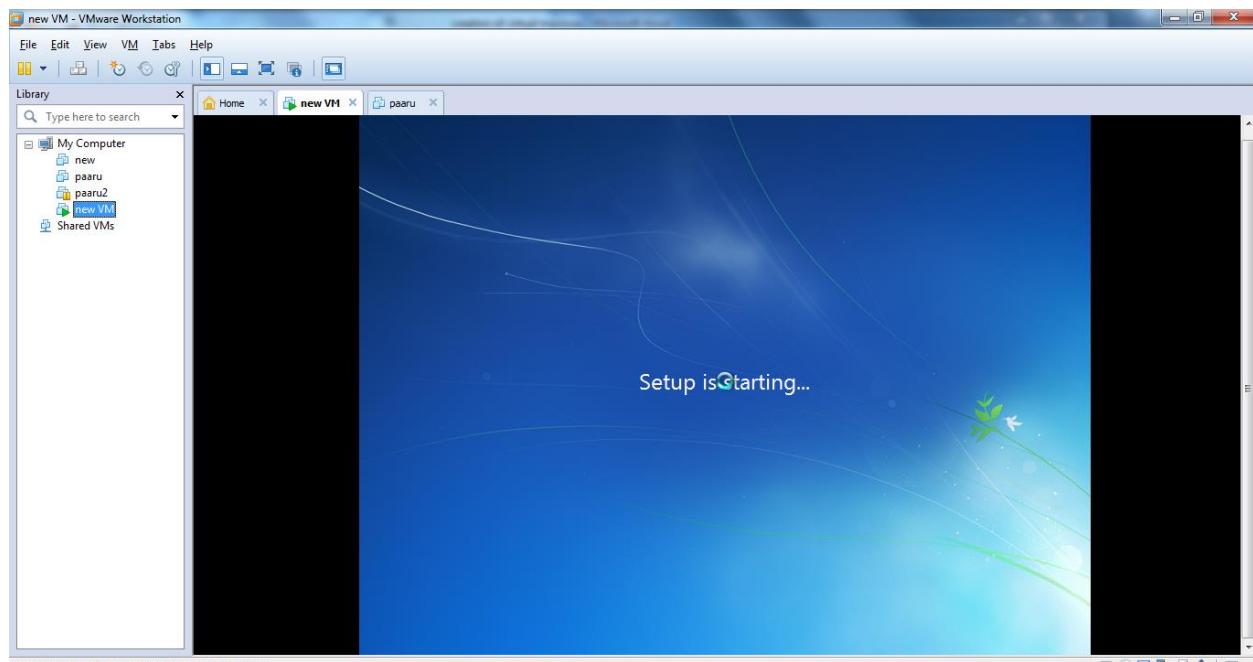
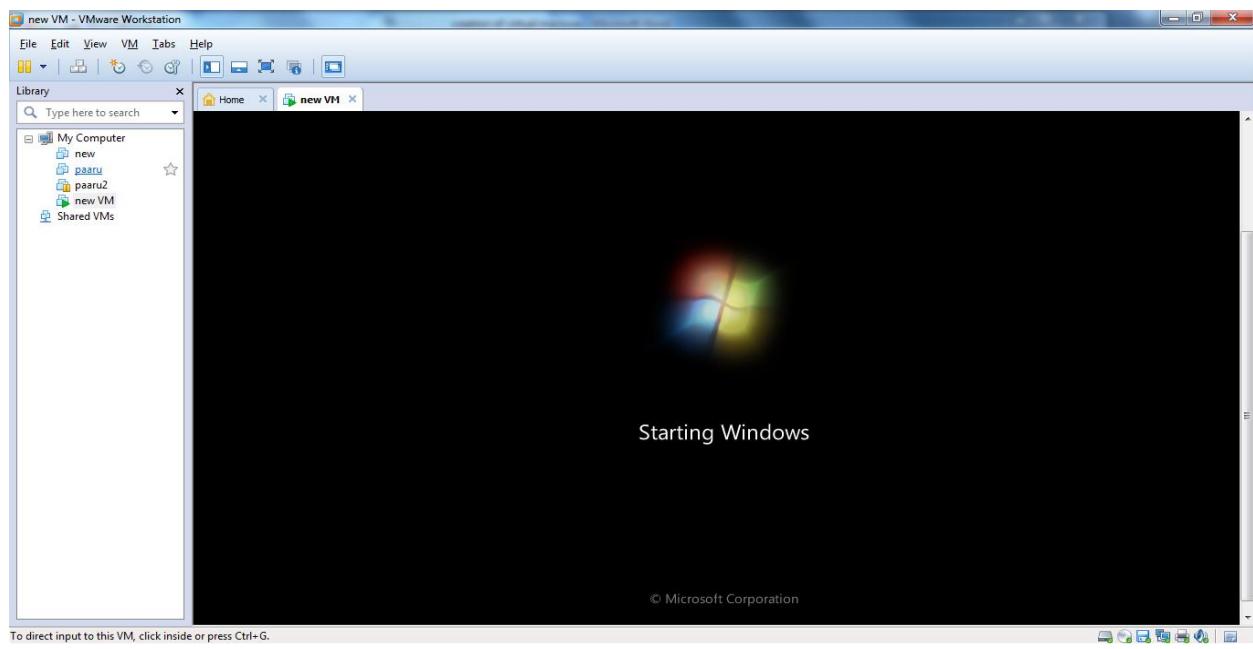
Click finish then new virtual machine will be created and it is ready for running. The created new virtual machine will be displayed on the leftside log of the VMware.

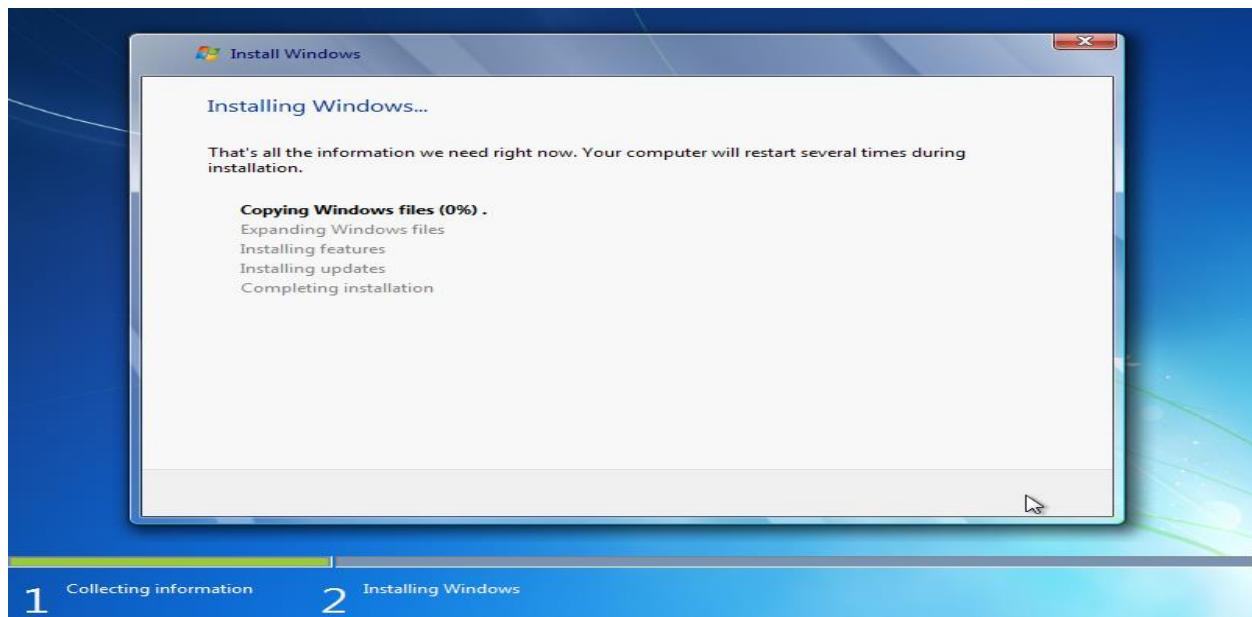
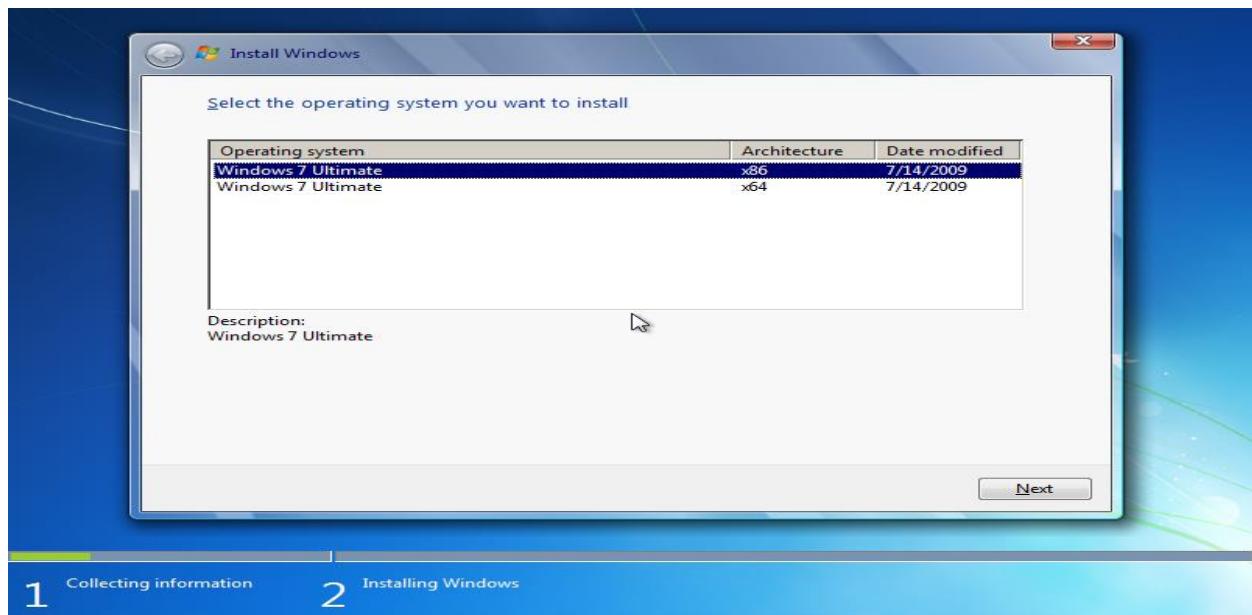
#### **4. CREATED NEW VIRTUAL MACHINE:**



#### **OUTPUT:**

Power on the newly created virtual machine. Virtual machine will get started with the booting sequence and loaded guest os will starting loading in the newly created virtual machine.





OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

## RESULT:

Thus the virtual machine has been created with different configuration and execution has been done successfully.

**EX. No 2**

**INSTALLING C COMPILER IN THE VIRTUAL MACHINE  
AND EXECUTION OF SAMPLE PROGRAM**

**AIM:**

To develop a C program using compiler in the virtual machine.

**ALGORITHM:**

**STEP 1:**Click the VMware Workstation.

**STEP 2:**Click the “Create a New Virtual Machine”option in Home Page.

**STEP 3:**Then,the VMware wizard automatically displayed on your screen.

**STEP 4:**Then,to click the custom option then click,next button.

**STEP 5:**In the browser,add a Ubuntu OS path,then click next button.

**STEP 6:**Then,the login page is opened,Enter the User name and Password must and confirm it and click next option.

**STEP 7:**To open the Terminal.(Shortcut-Ctrl+Alt+T)

**STEP 8:**Give the command in Terminal,**gedit**.

**STEP 9:**Open the Text Editor and type the simple C program.

**Example:**

```
#include<stdio.h>

Void main()
{
    printf("Hello World");
}
```

**STEP 10:** Compile the code using **gcc ex1.c**

**STEP 11:** Run the file by typing **./a.out**

**STEP 12:** Output will be displayed as "**Hello World**".

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

### **RESULT:**

Thus the above C program was developed using in C compiler in the virtual machine application executed successfully.

**EX. No 3**

**INSTALL GOOGLE APP ENGINE. CREATE A HELLO WORLD APP AND OTHER SIMPLE WEB APPLICATIONS USING PYTHON/JAVA.**

**AIM:**

To write a procedure on installation of google app engine and to create hello world app using java.

**INSTALLATION PROCEDURE:**

**Step 1:** Search google in google app engine and select the website page in google app engine in cloud platform.

A screenshot of a Google search results page. The search query "google app engine" is entered in the search bar. Below the search bar, there are filters for "All", "Images", "Books", "News", "Videos", and "More". To the right of the search bar are "Settings" and "Tools" buttons. The search results show approximately 95,00,00,000 results found in 0.66 seconds. The top result is a purple link titled "Google App Engine - Start Your GCP Free Trial" from "cloud.google.com/appengine". Below the link, there is a snippet of text: "Automatically Scale up with a Cloud Hosted App. Try Now for Free. Scalable, Cost Efficient VMs. Customizable for Your Workloads. Create Your Account Free. Google's infrastructure. 24x7 Phone Support. Per-Second Billing. Deploy In Minutes. Free Trial Google Cloud." There are two columns of text below the snippet: "App Engine Pricing" (Competitive cloud pricing that scales with your app's usage) and "Sign Up for Free" (Kick start your development. Get started with a free trial). There are also two more sections: "Just Add Code" (Offload concerns like scaling up or down to handle traffic to Google) and "Secure by Design Cloud" (Build on Google's secure infrastructure & global network).

[App Engine Application Platform | Google Cloud](#)

[cloud.google.com › appengine](#) ▾

**Google App Engine** lets app developers build scalable web and mobile back ends in any programming language on a fully managed serverless platform.

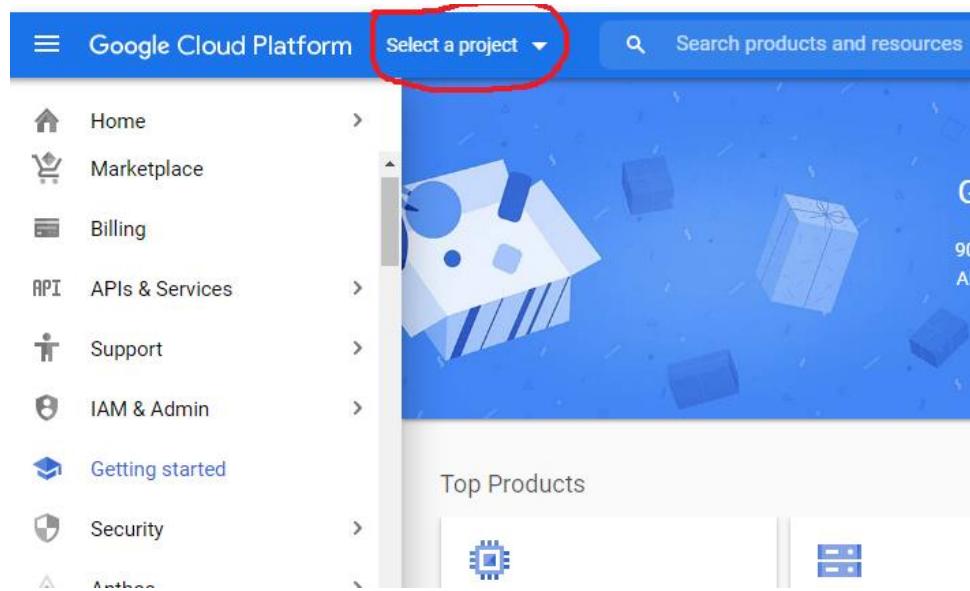
## Step 2: Login your account and click console button if you create new project.

The screenshot shows the Google Cloud App Engine landing page. At the top, there's a navigation bar with links for Google Cloud, Why Google, Solutions, Products, Pricing, and Getting Started. On the right side of the nav bar are links for Docs, Support, English (dropdown), and a redboxed 'Console' button. Below the nav bar, there's a search bar and buttons for Contact Sales and Get started for free. The main content area has a title 'App Engine' and a sub-section 'Serverless computing'. To the left is a sidebar with sections for App Engine (Benefits, Key features, Customers), Documentation (Try App Engine free), and Use cases. A callout box on the right says 'Scale your applications from zero to planet scale without having to manage' and includes a video thumbnail titled 'App Engine in 1 minute'.

## Step 3: Enter the country and agree the terms and services check box and select agree and continue.

The screenshot shows the Google Cloud Platform welcome screen. It features a sidebar with links for Home, Marketplace, Billing, APIs & Services, Support, IAM & Admin, Getting started, Security, Anthos, COMPUTE, and App Engine. The main area displays a 'Welcome Raj!' message, a 'Country' dropdown set to India, and a 'Terms of Service' section with a checked checkbox and a red circle around it. Below the checkbox is the text: 'I agree to the [Google Cloud Platform Terms of Service](#), and the terms of service of [any applicable services and APIs](#)'. At the bottom right of the modal is a large redboxed 'AGREE AND CONTINUE' button.

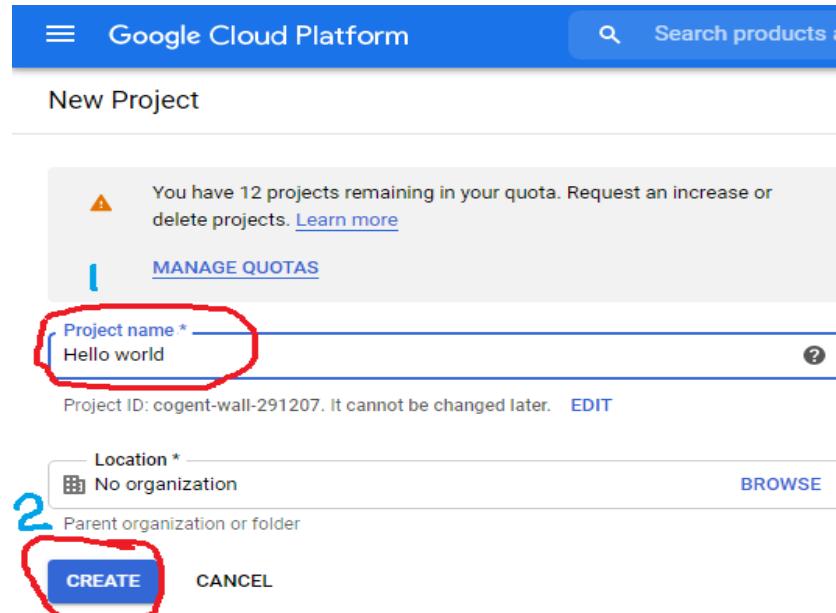
**Step 4:** Select the select project button in top of menu bar.



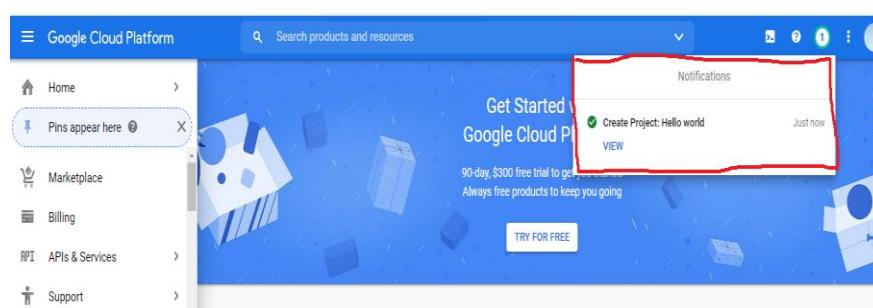
**Step 5:** Click new project.

A screenshot of the "Select a project" page. At the top left is the text "Select a project". To the right is a "NEW PROJECT" button with a star icon, which is also highlighted with a red box. Below this is a search bar with the placeholder "Search projects and folders". Underneath the search bar are two tabs: "RECENT" and "ALL", with "ALL" being underlined. A table follows, with columns for "Name" and "ID". There is one entry: "No organization" with an ID of "0".

**Step 6:** Enter the project name then click create button. Next show notification "create project hello world".



The screenshot shows the 'New Project' page in the Google Cloud Platform. At the top, there's a quota warning: 'You have 12 projects remaining in your quota. Request an increase or delete projects.' with a 'Learn more' link and a 'MANAGE QUOTAS' button. Below that is the 'Project name' field, which has a red border and contains 'Hello world'. To the right of the field is a question mark icon. Underneath the field, it says 'Project ID: cogent-wall-291207. It cannot be changed later.' with an 'EDIT' link. The 'Location' section shows 'No organization' with a 'BROWSE' button. Below that is a 'Parent organization or folder' section. At the bottom are 'CREATE' and 'CANCEL' buttons, with the 'CREATE' button also having a red border and a blue number '2' above it. The background shows a 'Get Started' banner for Google Cloud Platform.



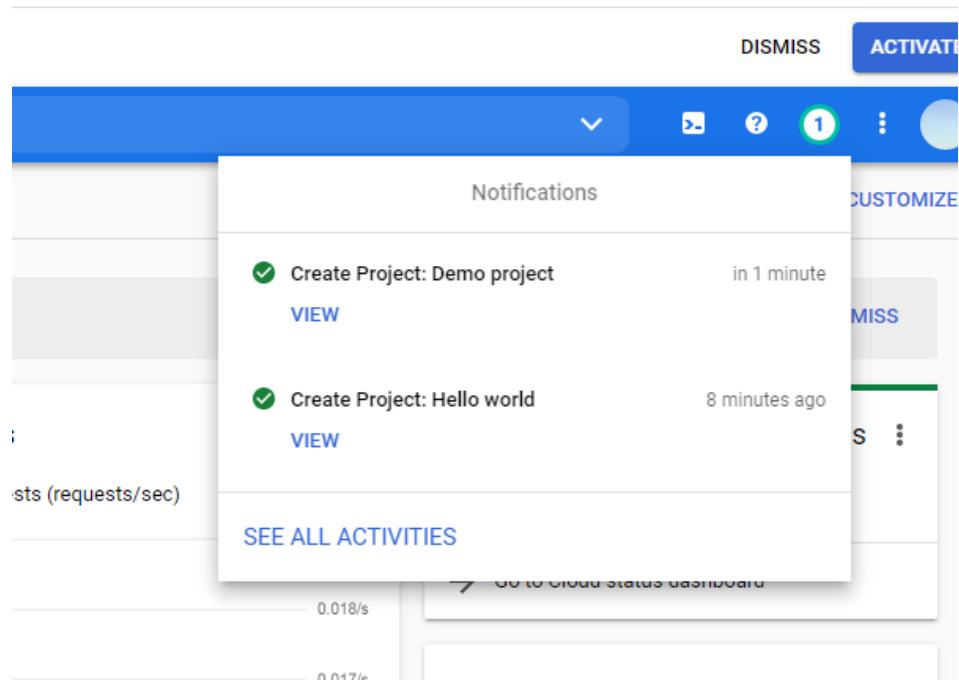
The screenshot shows the Google Cloud Platform dashboard. In the top right corner, there's a 'Notifications' box with a red border. It contains a single item: 'Create Project: Hello world' with a green checkmark and a 'VIEW' link. The rest of the dashboard includes a navigation menu on the left with options like Home, Marketplace, Billing, APIs & Services, and Support. The main area features a 'Get Started' banner for Google Cloud Platform.

**Step 7:** Go to Navigation menu and select Home icon then select dashboard. Now create another project in the top of the menu bar.

The screenshot shows the Google Cloud Platform dashboard. At the top, there is a blue header bar with the text "Your free trial is waiting: activate now to get \$300 credit to explore Google Cloud products. [Learn more](#)". Below the header is the "Google Cloud Platform" logo and a dropdown menu labeled "Hello world". A red box highlights this dropdown. To its right is a search bar with the placeholder "Search products and resources". The main content area has tabs for "DASHBOARD", "ACTIVITY", and "RECOMMENDATIONS". On the left, there is a sidebar with links: "Home", "Pins appear here", "Marketplace", "Billing", "APIs & Services", "Support", and "IAM & Admin". A red box highlights the "APIs & Services" link. On the right, there are two cards: "Project info" and "API APIs". The "Project info" card shows details: Project name: Hello world, Project ID: cogent-wall-291207, and Project number: 383336242443. The "API APIs" card shows requests per second: 1.0, 0.8, and 0.6, with a note: "No data is available for the selected API".

**Step 8:** Click new project button.Then get another notification to"create project: demo project".

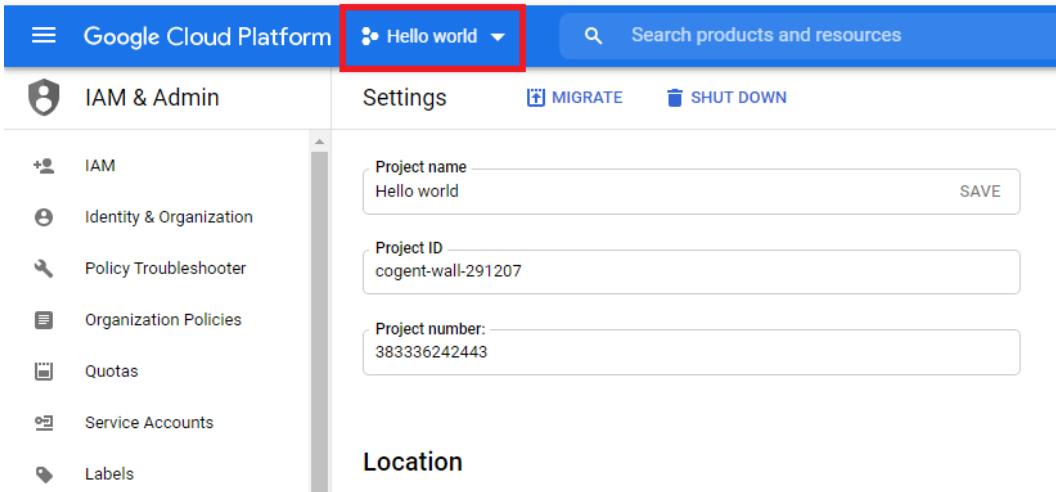
The screenshot shows a modal dialog titled "Select a project". At the top is a search bar with the placeholder "Search projects and folders". Below it are tabs for "RECENT" and "ALL". A table lists projects: "Hello world" (selected, indicated by a checkmark icon) and "cogent-wall-291207". At the bottom right of the dialog is a red box highlighting a "NEW PROJECT" button.



**Step 9:** Click go to project settings.

A screenshot of the Google Cloud Status dashboard. On the left, the "Project info" section shows the project name "Hello world", Project ID "cogent-wall-291207", and Project number "383336242443". Below this is a "ADD PEOPLE TO THIS PROJECT" button. At the bottom of this section is a button labeled "→ Go to project settings", which is highlighted with a red rectangular box. On the right, the "API APIs" section shows a timeline of request rates from 0.015/s to 0.019/s. The entire dashboard has a light gray background with dark blue header elements.

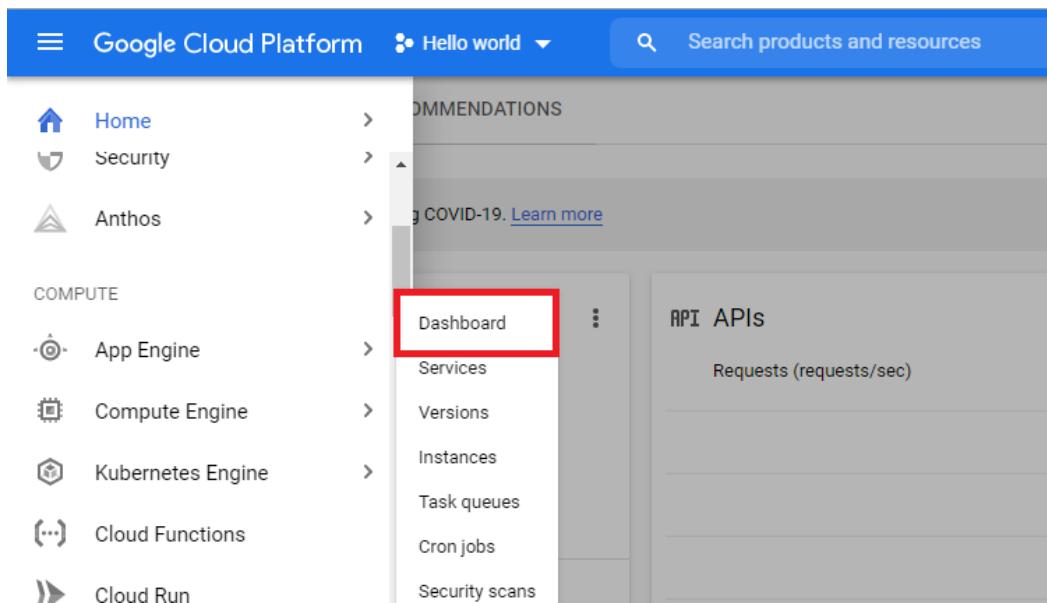
**Step 10:** Now click "hello world" in top of menu bar.



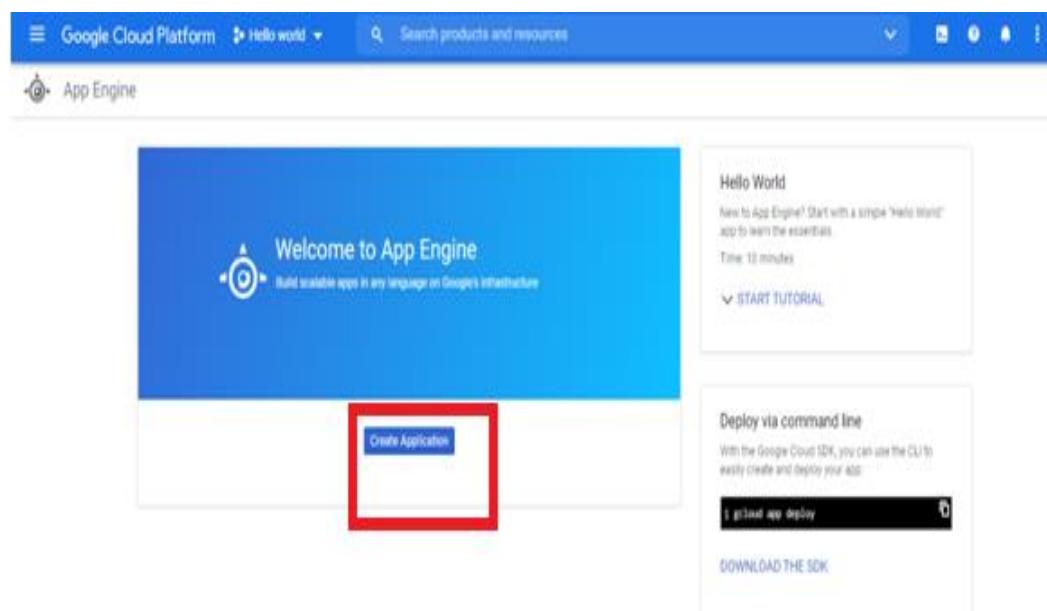
**Step 11:** Next select project 1.

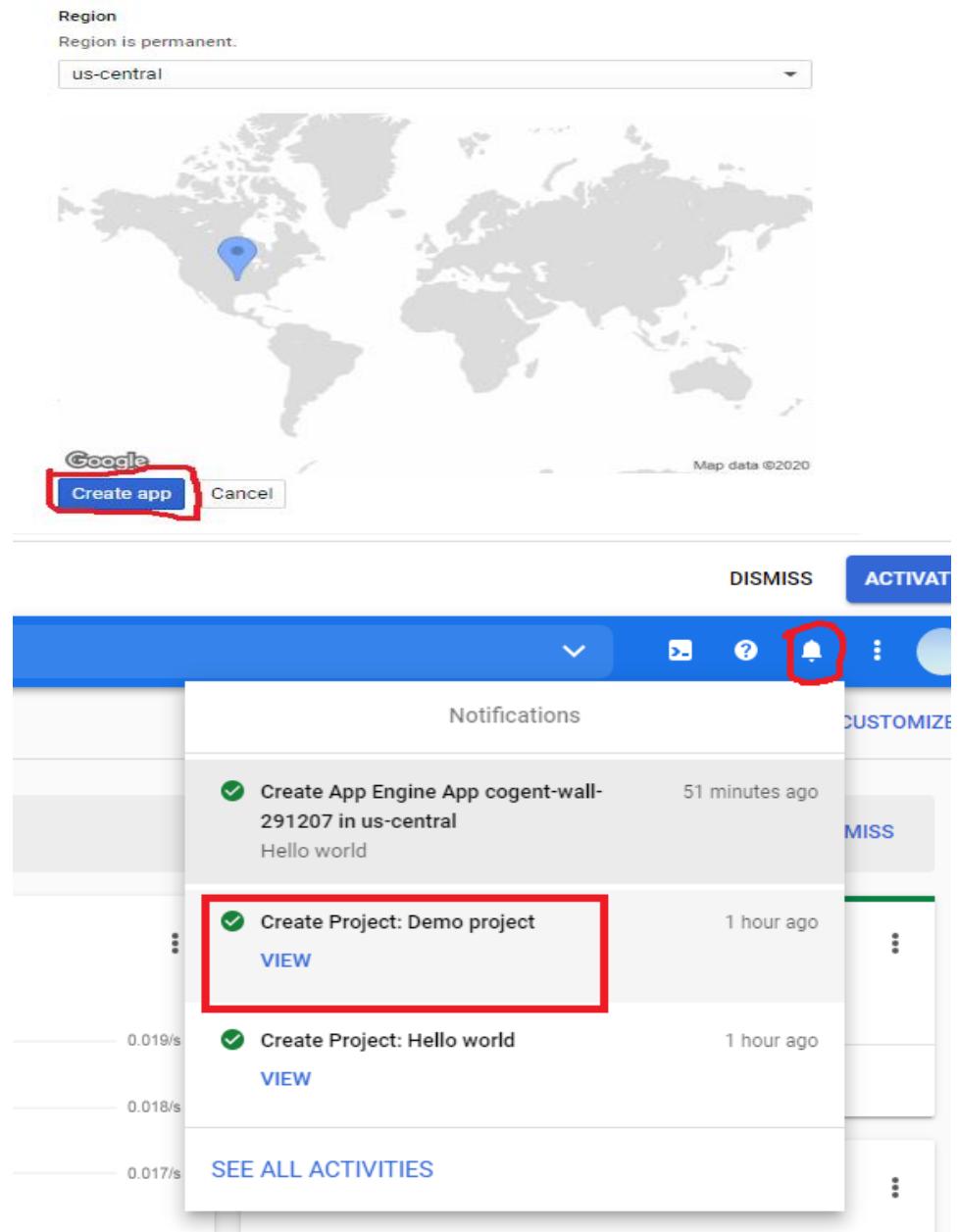
The screenshot shows a "Select a project" dialog. At the top, it says "Select a project" and has a "NEW PROJECT" button. Below is a search bar with the placeholder "Search projects and folders". Underneath, there are two tabs: "RECENT" (which is underlined) and "ALL". A table follows, with columns "Name" and "ID". It lists two projects: "Demo project" (ID: virtual-limiter-291207) and "Hello world" (ID: cogent-wall-291207). The "Hello world" row is highlighted with a red box around its entire row.

**Step 12:** Click Navigation Button in below select App Engine and Then Select Dashboard.

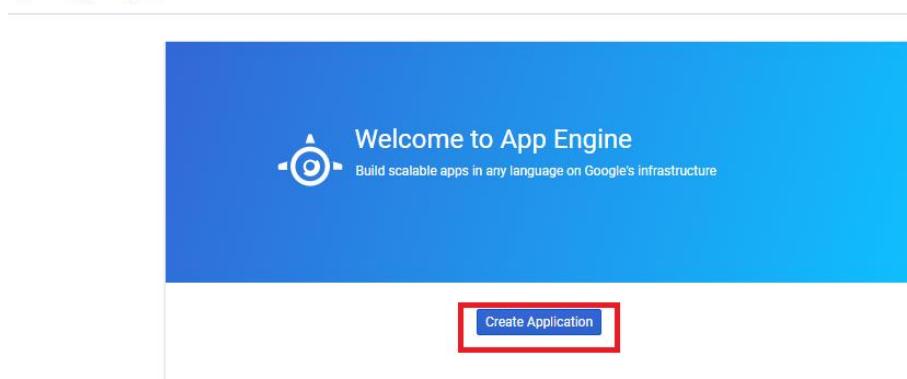
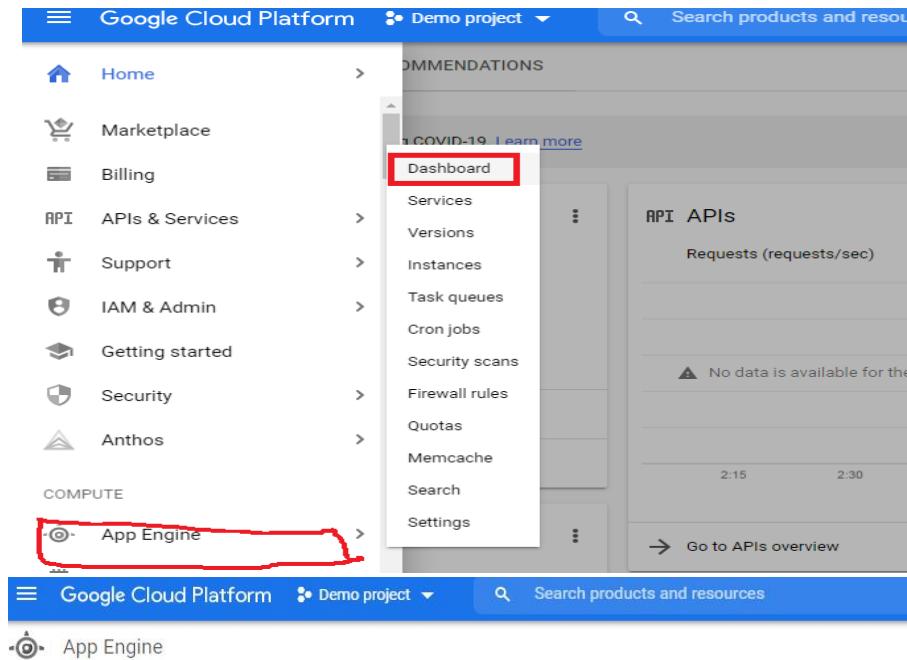


**Step13:** Next Step Click the "create Application". Then Click the "Create App". Now get Notification "create App Engine Cogent-Wall-291201 in Us-central".

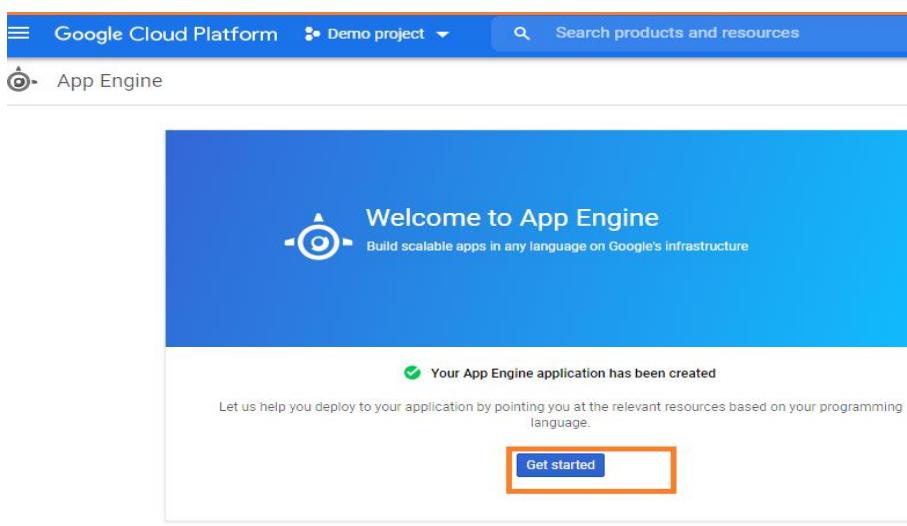


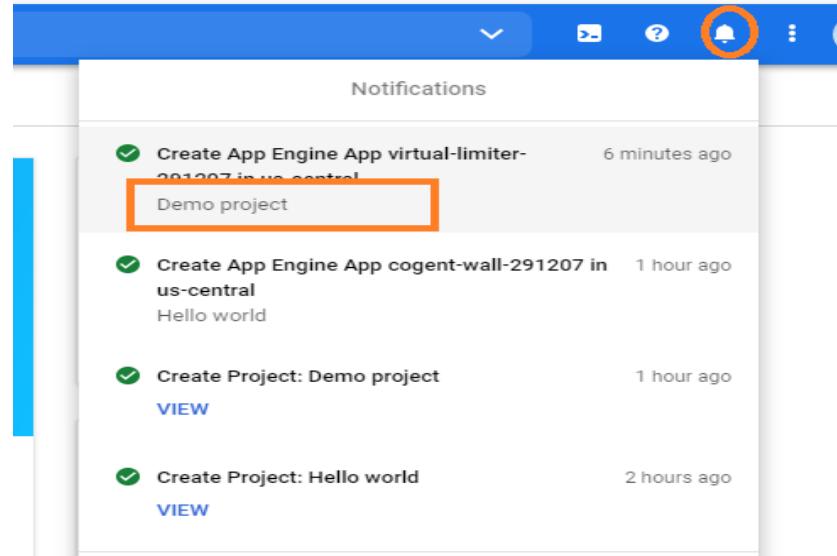


**Step 14:** Now done in step 12 And Step 13.

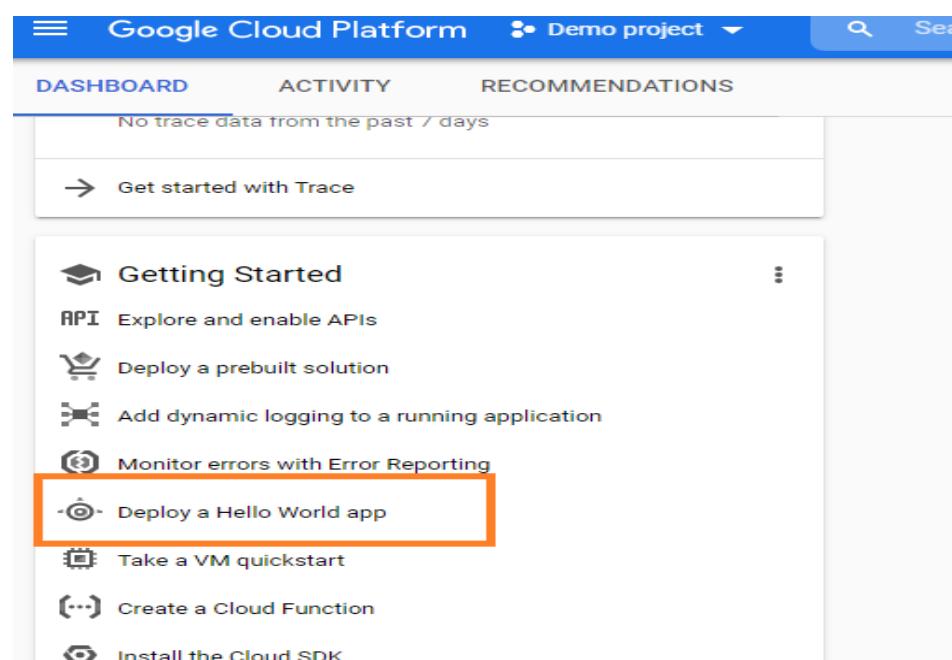


**Step 15:** Click Navigation Menu And Select App Engine . Then Click "Get Started".





**Step 16:** Click Bottom "Deploy a Hello World app ".Now Open the tutorial box in right side and Click Start Button.



ACTIVATE

LEARN Tutorial

App Engine Quickstart

Introduction

This tutorial shows you how to deploy a sample application to App Engine using the `gcloud` command.

Here are the steps you will be taking.

- **Create a project**  
Projects bundle code, VMs, and other resources together for easier development and monitoring.
- **Build and run your "Hello, world!" app**  
You will learn how to run your app using Cloud Shell, right in your browser. At the end, you'll deploy your app to the web using the `gcloud` command.
- **After the tutorial...**  
Your app will be real and you'll be able to experiment with it after you deploy, or you can remove it and start fresh.

Start

**Step 17:** Click the Activate Cloud Shell And Then open the bottom terminal box.Next activate" `git clone https://github.com/GoogleCloudPlatform/golang-samples`" and click enter key.Next Activate "git clone `https://github.com/GoogleCloudPlatform/golang-samples`" and click Next.

## Using Cloud Shell

Cloud Shell is a built-in command-line tool for the console. We're going to use Cloud Shell to deploy our app.

### Open Cloud Shell

Open Cloud Shell by clicking the 

**Activate Cloud Shell** button in the navigation bar in the upper-right corner of the console.

### Clone the sample code

Use Cloud Shell to clone and navigate to the "Hello World" code. The sample code is cloned from your project repository to the Cloud Shell.

Note: If the directory already exists, remove the previous files before cloning.

```
gleCloudPlatform/golang-samples
```

Then, switch to the tutorial directory:

```
$ cd \
  golang-samples/appengine/go
```

[Previous](#)

Step 2 of 8

[Next](#)

**Step 18:** Activate "cat helloworld.go" and Activate "cat app.yaml" And click start button. then type terminal "go run" and click the "Webpreview" in top terminal box and select "change port 8080".and click start.finally visit your app.

## Exploring the application

Enter the following command to view your application code:

```
$ cat helloworld.go
```



## Exploring your configuration

App Engine uses YAML files to specify a deployment's configuration. `app.yaml` files contain information about your application, like the runtime environment, URL handlers, and more.

Enter the following command to view your configuration file:

```
$ cat app.yaml
```



The syntax of this file is [YAML](#). For a complete list of configuration options, see the [app.yaml](#) reference.

[Previous](#)

Step 3 of 8

[Next](#)

The screenshot shows the Google Cloud Platform App Engine interface. At the top, there's a navigation bar with 'Google Cloud Platform' and 'Welcome KCET cuddalor'. A search bar says 'Search products and resources'. Below the navigation is a banner for 'Welcome to App Engine' with the subtext 'Build scalable apps in any language on Google's infrastructure'. A message indicates 'Your App Engine application has been created'. To the right, there's a 'START TUTORIAL' button and a section titled 'Deploy via command line' with instructions about using the Google Cloud SDK. The main area is a 'CLOUD SHELL' terminal window titled '(virtual-limiter-291207)'. It shows a connection warning: 'The connection to your Google Cloud Shell was lost.' The terminal history includes commands like 'git clone', 'cd', 'cat', 'gcloud app create', and 'gcloud app deploy'. The output ends with 'runtime: go112 # replace with go111 for Go 1.11'.

The screenshot shows a web browser with multiple tabs. One tab is open at the URL '8080-76cbde49-2ba8-4fce-acab-80975a2f1aea.asia-southeast1.cloudshell.dev/?authuser=0'. The page displays the text 'Hello, World!'. A tooltip for this text reads 'Cloud SDK Command Line' and 'Cloud SDK: Command Line cloud.google.com'.

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

## RESULT:

Thus the Installation of Google App Engine has been done and hello world app was created successfully using java.

**EX NO:4**

## **USE THE GAE LAUNCHER TO LAUNCH THE WEB APPLICATIONS**

### **AIM:**

To launch a setup for a web application in Google App Engine.

### **PROCEDURE:**

Google App Engine is a powerful platform that lets you build and run applications on Google's infrastructure whether you need to build a multi-tiered web application from scratch or host a static website. Here's a step-by-step guide to hosting your website on Google App Engine.

#### **(i) Creating a Google Cloud Platform project:**

To use Google's tools for your own site or app, you need to create a new project on Google Cloud Platform. This requires having a Google account.

1. Go to the [App Engine dashboard](#) on the Google Cloud Platform Console and press the *Create* button.
2. If you've not created a project before, you'll need to select whether you want to receive email updates or not, agree to the Terms of Service, and then you should be able to continue.
3. Enter a name for the project, edit your project ID and note it down.
  - Project Name: *GAE Sample Site*
  - Project ID: *gaesamplesite*
4. Click the *Create* button to create your project.

---

#### **(ii) Creating an application:**

Each Cloud Platform project can contain one App Engine application. Let's prepare an app for our project. We'll need a sample application to publish. If you've not got one to use, download and unzip this [sample app](#).

1. Have a look at the sample application's structure — the website folder contains your website content and *app.yaml* is your application configuration file.

- Your website content must go inside the website folder, and its landing page must be called index.html, but apart from that it can take whatever form you like.
  - The app.yaml file is a configuration file that tells App Engine how to map URLs to your static files. You don't need to edit it.
- 

### (iii) Publishing your application:

Now that we've got our project made and sample app files collected together, let's publish our app.

1. Open Google Cloud Shell.
2. Drag and drop the sample-app folder into the left pane of the code editor.
3. Run the following in the command line to select your project:

```
gcloudconfig set project gaesamplesite
```

4. Then run the following command to go to your app's directory:

```
cdsample-app
```

5. You are now ready to deploy your application, i.e. upload your app to App Engine:

```
gcloud app deploy
```

6. Enter a number to choose the region where you want your application located.
7. Enter Y to confirm.
8. Now navigate your browser to *your-project-id.appspot.com* to see your website online. For example, for the project ID *gaesamplesite*, go to *gaesamplesite.appspot.com*.

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

### **RESULT:**

Thus the procedure to launch Google App Engine executed successfully.

**EX NO: 5**

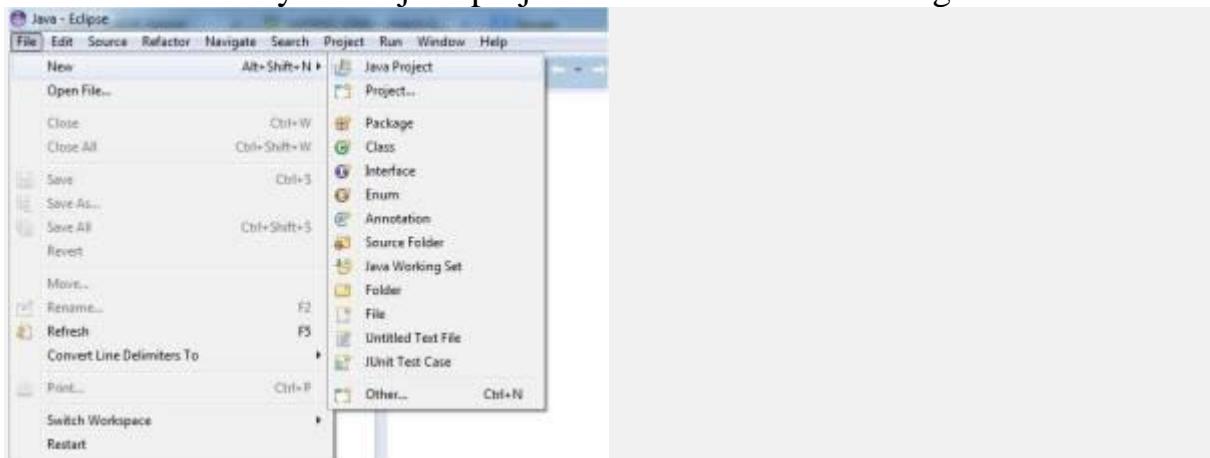
**SIMULATE A CLOUD SCENARIO USING CLOUDSIM AND RUN A SCHEDULING ALGORITHM THAT IS NOT PRESENT IN CLOUDSIM.**

**AIM:**

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

**PROCEDURE:****STEP BY STEP INSTALLATION OF CLOUD SIM INTO ECLIPSE:**

1. Open up Eclipse and go to Menu Section, then click File, keep on clicking New and finally select java project. It is shown as in the Figure1



Open eclipse and select java project

Open up Eclipse and Click on java project

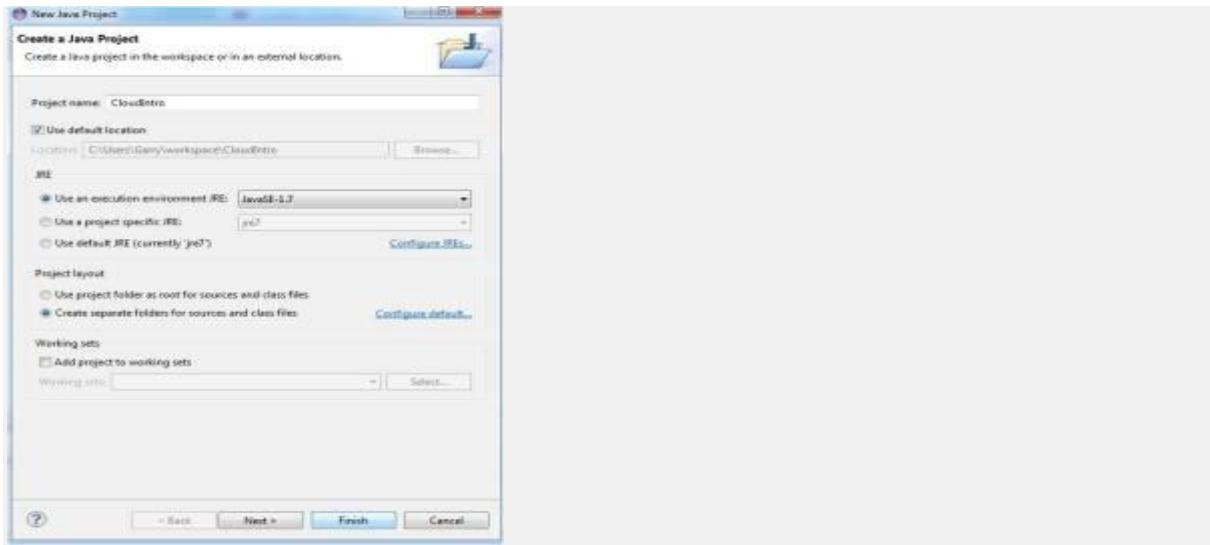
2. A new window will get open. Put a foot on to the following steps:-

**2.1.** Enter project name. (I have named it as CloudIntro)

**2.2** In the next line you will see the path where your project will be created as it as shown in the Figure2.

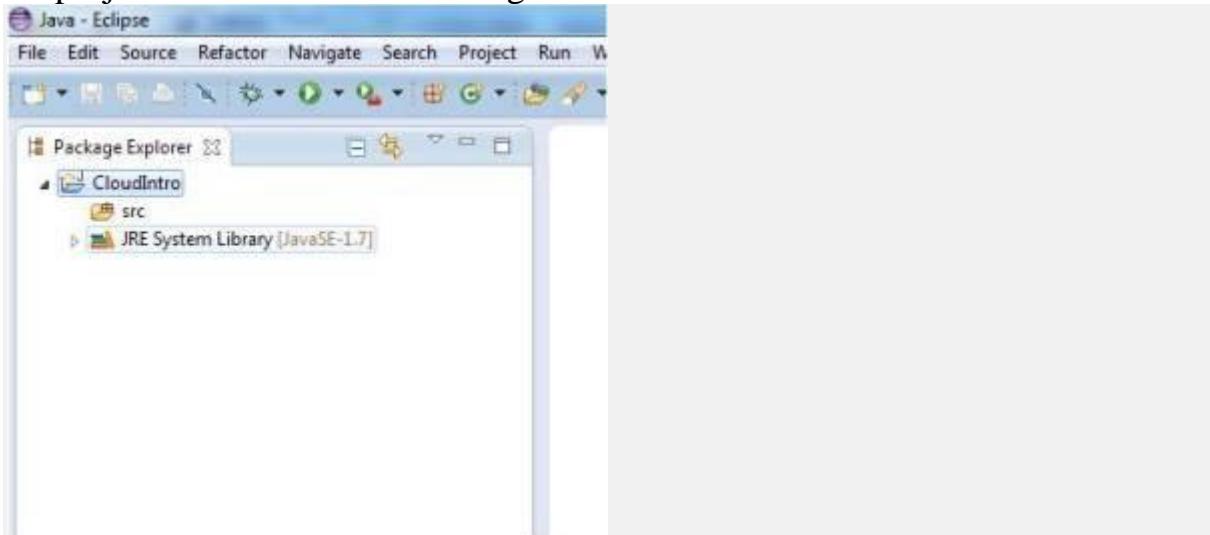
**2.3** Next You need to select the JRE environment.

**2.4** Finally Click Finish



Give project Name and select run time environment and Finish

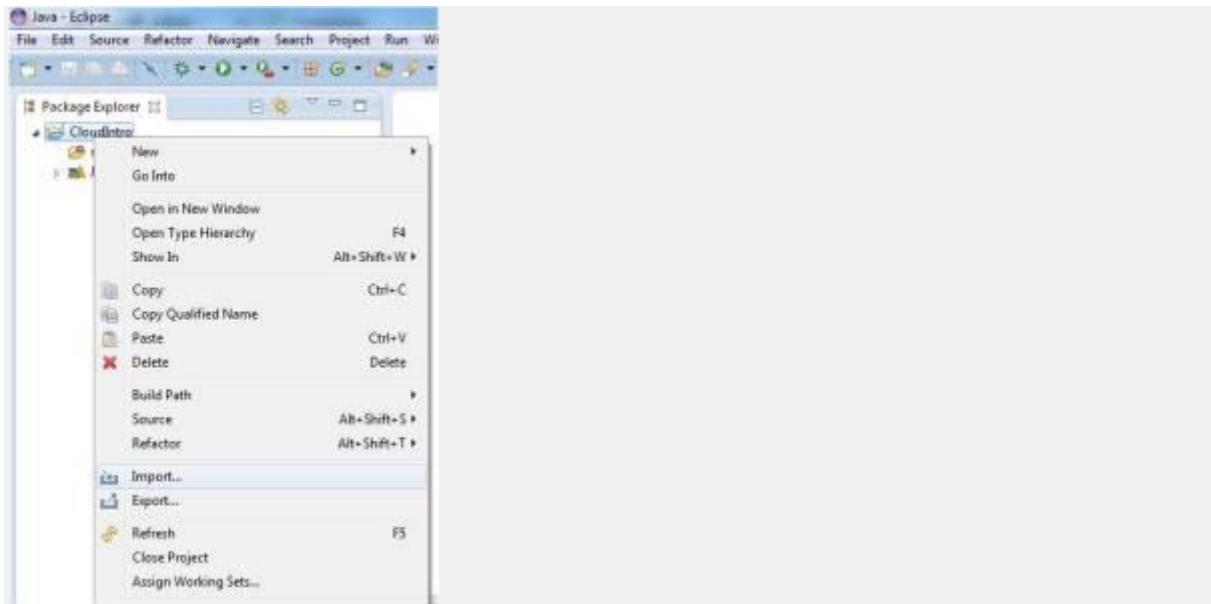
- Once you hit finish. An empty project named CloudIntro will be created in the project List as shown in the Figure3.



Project Folder Location

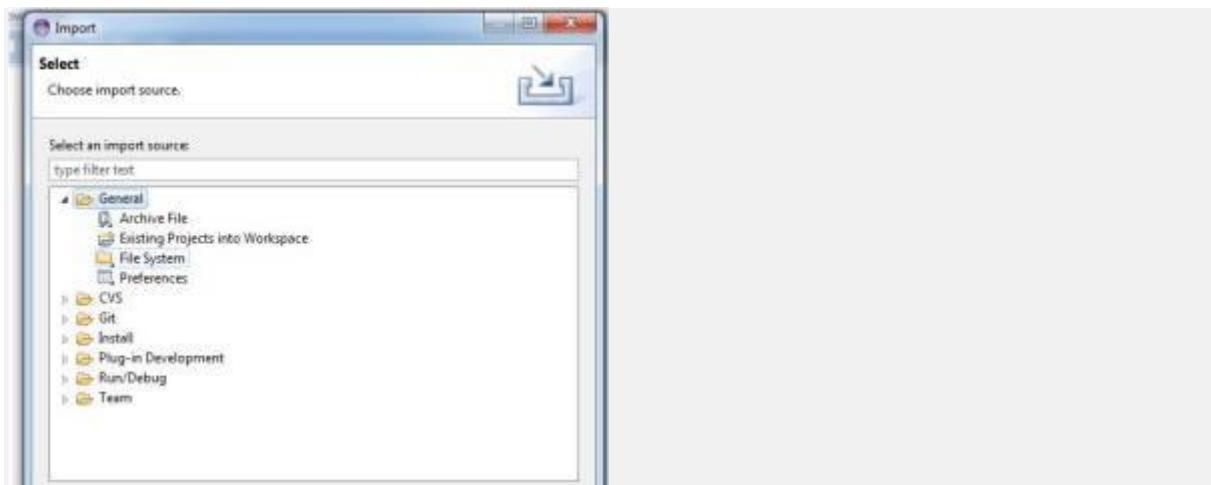
## REPORT THIS AD

- Next step is to go the project CloudIntro, right click on it. Click Import as shown in the Figure4.



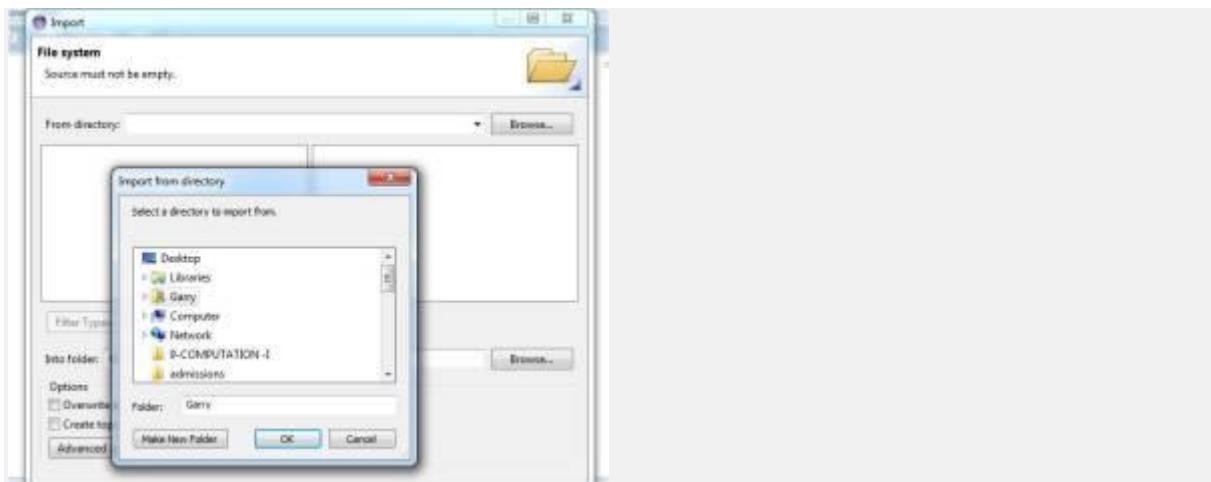
Import cloud sim tool files and subsequent folders

**5.** A new window will get open, now click File System as demonstrated in the Figure5.



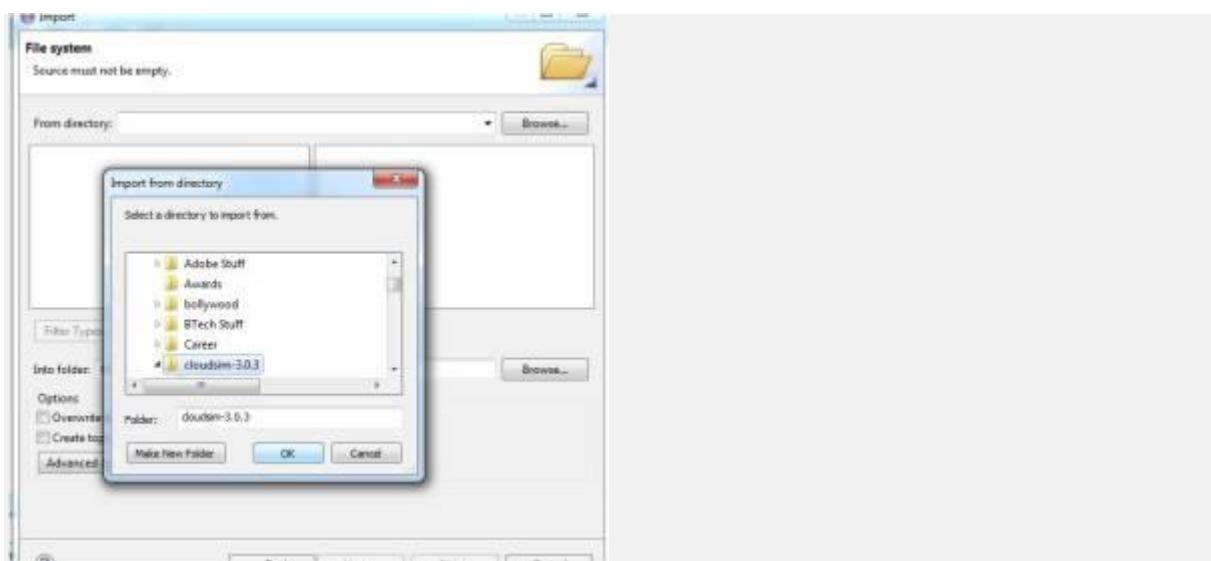
Next to select is File System

**6.** Next Step is to go to the directory where you have extracted your cloud sim tool. Figure6 is shown to guide you to get into the directory where your cloudsim folder is located.



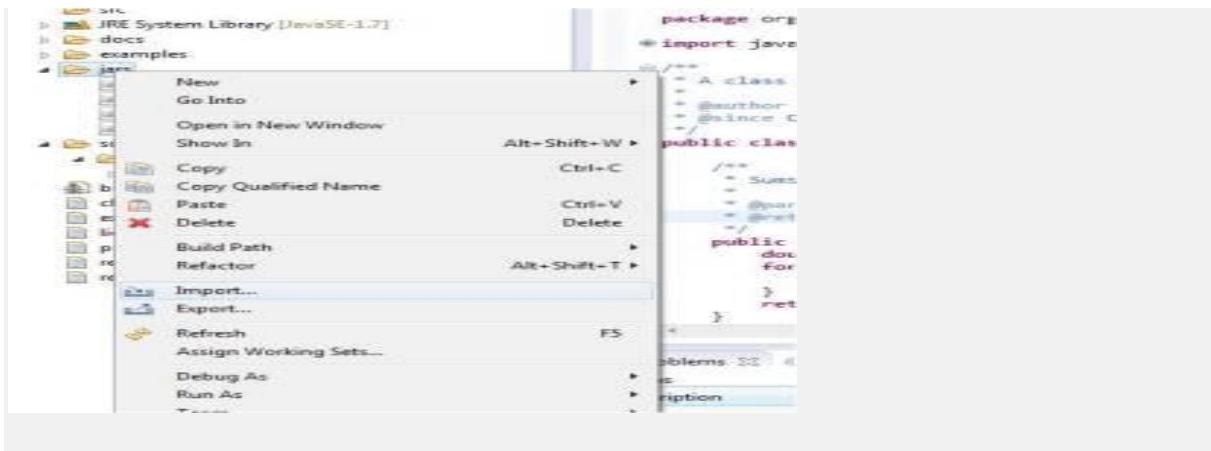
Go to Directory to select Cloudsim (My system searching)

## 7. Select the cloudsim and click Finish as shown in the Figure7.



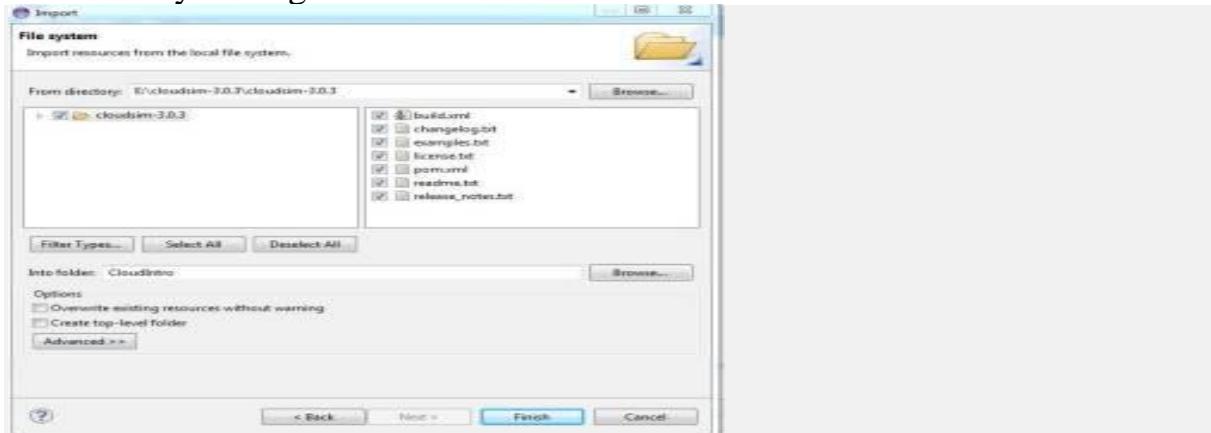
Select Cloudsim and Hit finish

## 8. Now go to the left side of the eclipse tool in the project bar. Go to jar and right click on it. Click import as shown in the Figure8.



Import jar files for math calculations

**9.** Now go to the folder where you have placed the downloaded and extracted file as described by point 8. Then all you have to do is select that jar file and hit finish as shown by the Figure9.



Import only jar

**10.** Finally the cloud sim is installed into your Eclipse environment.

## RESULT:

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

Thus the procedure to simulate a cloud scenario using cloudsim and run a scheduling algorithm that is not present in cloudsim was executed successfully.

**EX NO:6**

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

**AIM:**

To find a procedure for transferring files from one virtual machine to another virtual machine.

**PROCEDURE:**

**Preparing your Virtual Machine for the Move:**

1. Shut down the guest operating system and power off the virtual machine. If the virtual machine is suspended and its virtual disks are in persistent or nonpersistent mode, resume it, then shut down the guest operating system.
2. If your virtual machine is using disks in undoable mode, it is best to commit or discard the changes when the guest operating system shuts down. If you cannot commit or discard the changes to your disk, read [Considerations for Moving Disks in Undoable Mode](#).

**Note:** If your disks are using nonpersistent mode, you must also move the redo-log (.REDO) file to the new host computer. By default, it is located in your host operating system's temp directory.

3. Do one of the following:
  - o If you are moving the virtual machine to a new host and have a network connection between the original host machine and the new host, you are finished with the preparations on the original host. Otherwise, you need to have a way of moving the virtual disk (.vmdk) files from the virtual machine's directory to the new host. You could move them to a shared network directory, for example, or burn them to CD-ROMs if they are not too large.

Once you know how you are going to move the virtual machine, go to [Moving a Virtual Machine to a New Host Machine](#).

If you are moving this virtual machine to another directory on this host, then you are ready to make the move. Copy all the files in the virtual machine's original directory to the new location. If you stored any files in directories other

than the virtual machine directory, be sure to move them into a directory of the same name and same position relative to the location of the virtual machine. Start VMware Workstation and open the new virtual machine you just created. Choose **File > Open**, then browse to the virtual machine's configuration file. The virtual machine is added to the **Virtual Machine Name** list in the Workstation window.

### **Moving a Virtual Machine to a New Host Machine:**

1. Make sure VMware Workstation is installed and working correctly on the new host computer.
2. Locate the virtual disk files you are moving and copy them into the new virtual machine directory. Be sure to copy all the files in the virtual machine's original directory. If you stored any files in directories other than the virtual machine directory, be sure to move them into a directory of the same name and same position relative to the location of the virtual machine.

If, for some reason, you are *not* moving a file, make sure you do not have any relative or absolute paths pointing to file. Use the Configuration Editor and check to see if your virtual machine is pointing to the correct location for files you do not move.

Also, check to see you do not have any absolute paths pointing to any files you *are* moving.

To determine whether any files are using absolute or relative paths, use the Configuration Editor. Select each device. Also, look at the location of the redo-log file.

**Note:** If your virtual machine is using disks in undoable mode, it is best to commit or discard the changes when the guest operating system shuts down. If you cannot commit or discard the changes to your disk, read [Considerations for Moving Disks in Undoable Mode](#).

3. Start VMware Workstation and open the new virtual machine you just created. Choose **File > Open**, then browse to the virtual machine's configuration file. The virtual machine is added to the **Virtual Machine Name** list in the Workstation window.

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

### **RESULT:**

Thus the procedure for transferring files from one virtual machine to another virtual machine.

**EX NO:7**

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

**AIM:**

To write a word count program to demonstrate the use of map and reduce tasks.

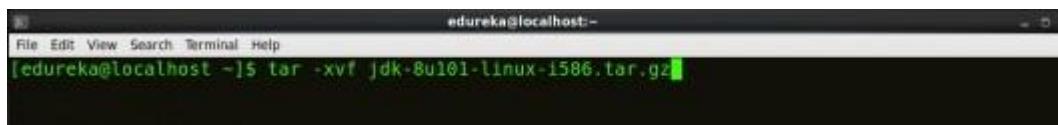
**PROCEDURE:**

**Install Hadoop**

**Step 1:** [Click here](#) to 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



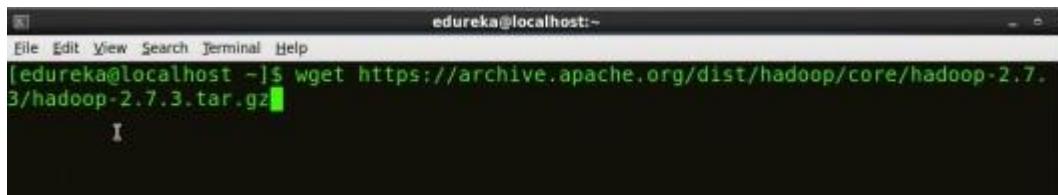
```
edureka@localhost:~$ tar -xvf jdk-8u101-linux-i586.tar.gz
```

*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>

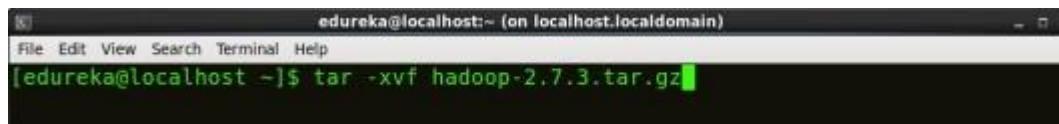


```
edureka@localhost:~$ wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.3/hadoop-2.7.3.tar.gz
```

*Fig: Hadoop Installation – Downloading Hadoop*

**Step 4:** Extract the Hadoop tar File.

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



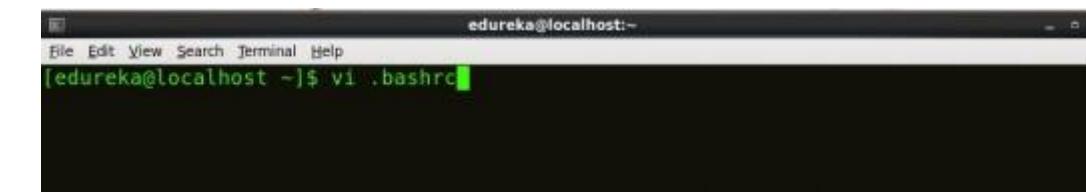
```
edureka@localhost:~ (on localhost.localdomain)
File Edit View Search Terminal Help
[edureka@localhost ~]$ tar -xvf hadoop-2.7.3.tar.gz
```

*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:~
File Edit View Search Terminal Help
[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

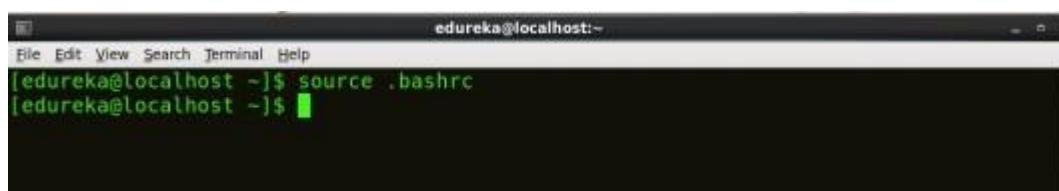
# Set JAVA_HOME
export JAVA_HOME=/home/edureka/jdk1.8.0_181
export PATH=/home/edureka/jdk1.8.0_181/bin:$PATH
```

*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.

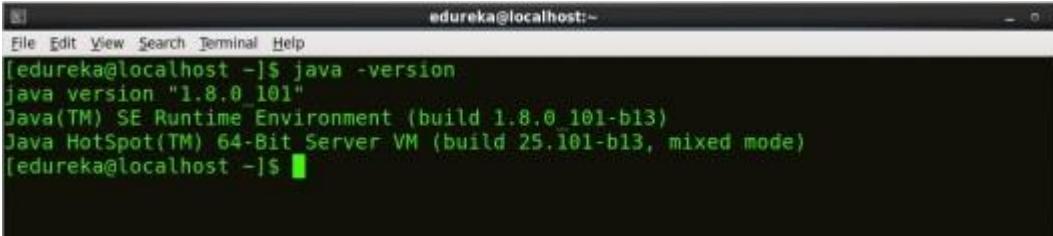
**Command:** source .bashrc



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



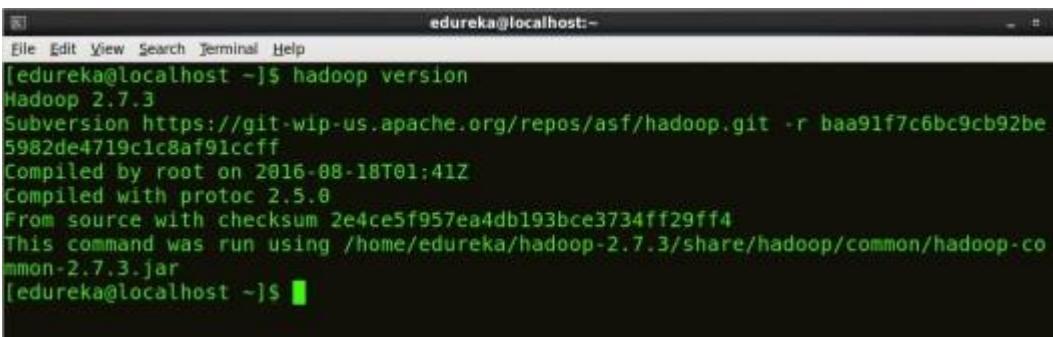
```
edureka@localhost:~$ java -version
java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)
[edureka@localhost ~]$
```

A screenshot of a terminal window titled 'edureka@localhost:~'. The window shows the command 'java -version' being run and its output. The output indicates Java version 1.8.0\_101, Java(TM) SE Runtime Environment, and Java HotSpot(TM) 64-Bit Server VM.

*Fig: Hadoop Installation – Checking Java Version*

**Command:** 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 ~]$
```

A screenshot of a terminal window titled 'edureka@localhost:~'. The window shows the command 'hadoop version' being run and its output. The output displays the Hadoop version as 2.7.3, the subversion URL, compilation details, and the checksum.

*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 ~]$ 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
```

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 NameNode runs in the cluster. It contains configuration settings of Hadoop core such as I/O settings that are common to HDFS & MapReduce.

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

```
edureka@localhost ~]$ 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

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

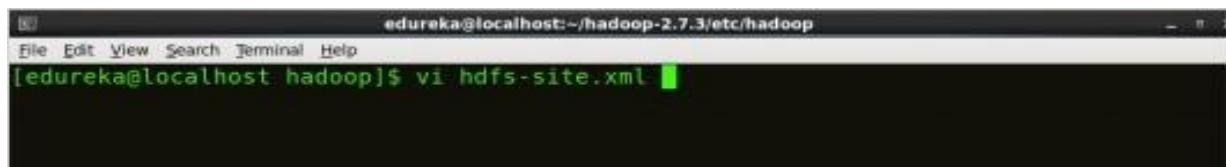
```

1           <?xml version="1.0" encoding="UTF-8"?>
2   <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3       <configuration>
4           <property>
5               <name>fs.default.name</name>
6               <value>hdfs://localhost:9000</value>
7           </property>
8       </configuration>

```

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

**Command:** 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>

```

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

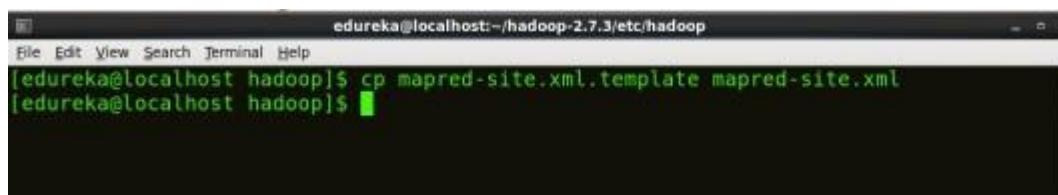
inside configuration tag

*mapred-site.xml* contains configuration settings of MapReduce 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.

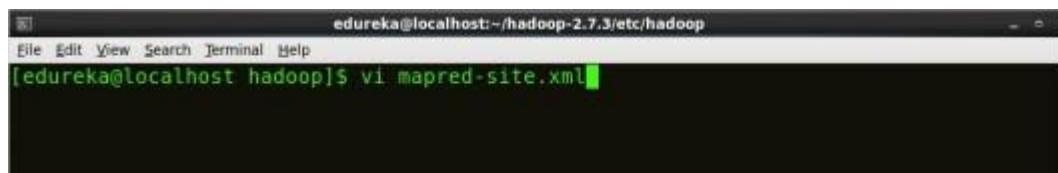
In some cases, *mapred-site.xml* file is not available. So, we have to create the *mapred-site.xml* file using *mapred-site.xml* template.

**Command:** cp *mapred-site.xml.template* *mapred-site.xml*

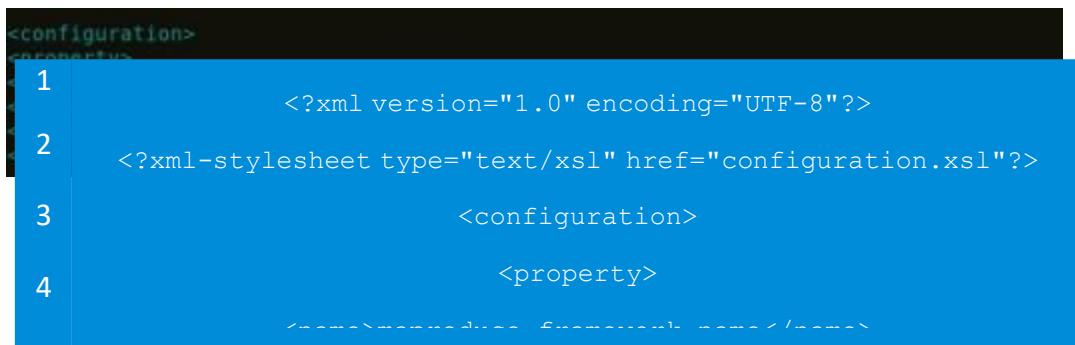
**Command:** vi *mapred-site.xml*.



```
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
```



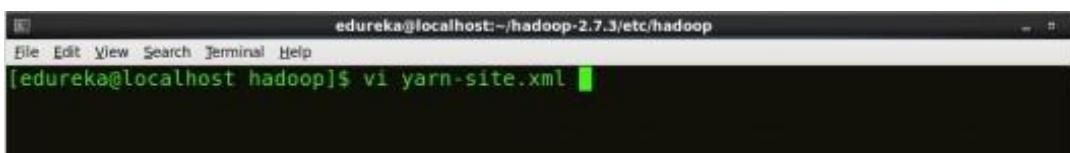
```
<configuration>
<property>
 1   <?xml version="1.0" encoding="UTF-8"?>
 2   <?xmlstylesheet type="text/xsl" href="configuration.xsl"?>
 3   <configuration>
 4     <property>
```

Fig: Hadoop Installation – Configuring *mapred-site.xml*

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

*yarn-site.xml* contains configuration settings of ResourceManager and NodeManager like application memory management size, the operation needed on program & algorithm, etc.

**Command:** 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>
```

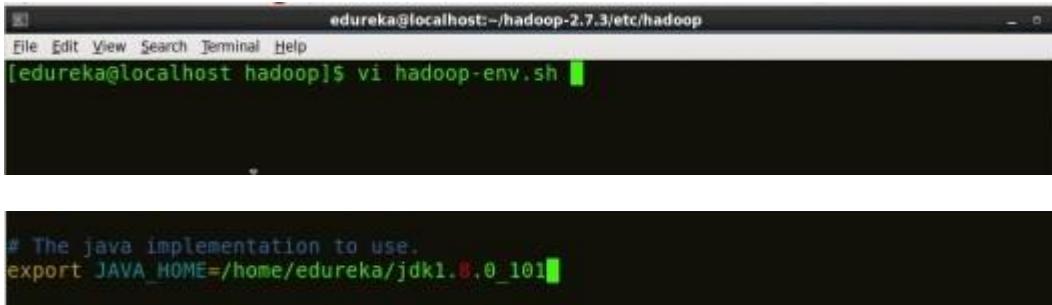
Fig: Hadoop Installation – Configuring *yarn-site.xml*

```
1
2           <?xml version="1.0">
3
4           <configuration>
5
6               <property>
7
8                   <name>yarn.nodemanager.aux-services</name>
9
10                  <value>mapreduce_shuffle</value>
11
12                  </property>
13
14                  <property>
```

**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 runHadoop like Java home path, etc.

**Command:** vi *hadoop-env.sh*



The terminal window shows the command `vi hadoop-env.sh` being run in the directory `/etc/hadoop`. The file contains the following configuration:

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

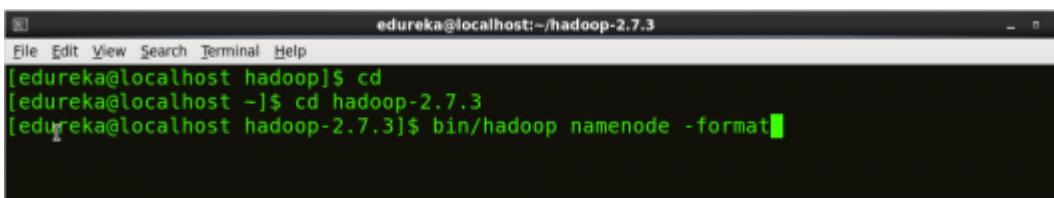
*Fig: Hadoop Installation – Configuring hadoop-env.sh*

**Step 12:** Go to Hadoop home directory and format the NameNode.

**Command:** `cd`

**Command:** `cd hadoop-2.7.3`

**Command:** `bin/hadoop namenode -format`



The terminal window shows the commands `cd`, `cd hadoop-2.7.3`, and `bin/hadoop namenode -format` being run sequentially.

*Fig: Hadoop Installation – Formatting NameNode*

This formats the HDFS via NameNode. 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 filesystem. You will lose all your data stored in the HDFS.

**Step 13:** Once the NameNode 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

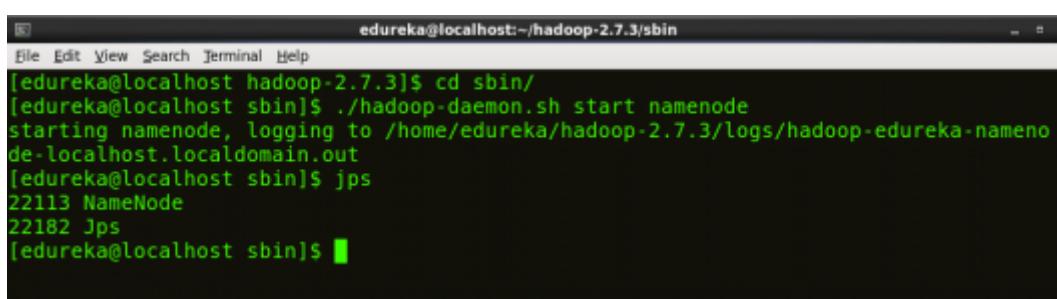
The above command is a combination of **start-dfs.sh**, **start-yarn.sh** & **mr-jobhistory-daemon.sh**

Or you can run all the services individually as below:

### Start NameNode:

The NameNode 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 namenode



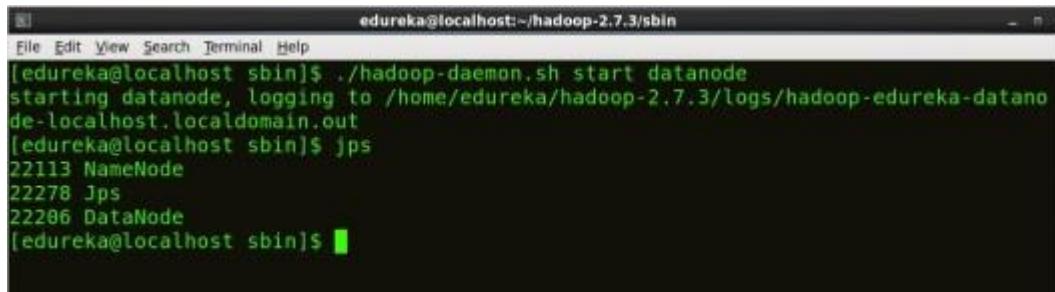
```
edureka@localhost:~/hadoop-2.7.3/sbin
File Edit View Search Terminal Help
[edureka@localhost hadoop-2.7.3]$ cd sbin/
[edureka@localhost sbin]$ ./hadoop-daemon.sh start namenode
starting namenode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-namenode-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22182 Jps
[edureka@localhost sbin]$
```

*Fig: Hadoop Installation – Starting NameNode*

### Start DataNode:

On startup, a DataNode connects to the Namenode and it responds to the requests from the Namenode for different operations.

**Command:** ./hadoop-daemon.sh start datanode



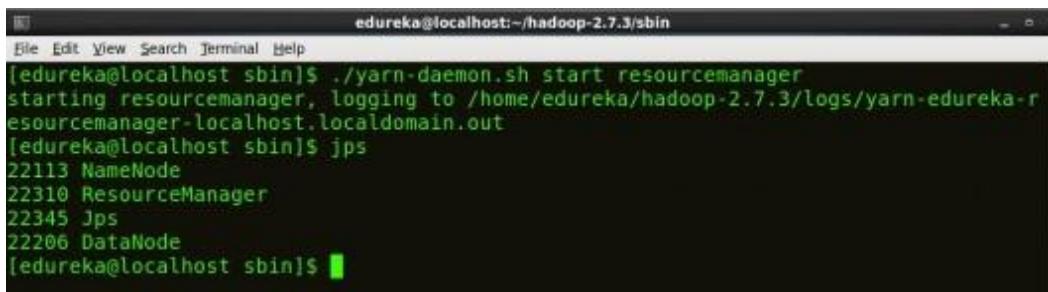
```
edureka@localhost sbin$ ./hadoop-daemon.sh start datanode
starting datanode, logging to /home/edureka/hadoop-2.7.3/logs/hadoop-edureka-datanode-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22278 Jps
22206 DataNode
[edureka@localhost sbin]$
```

Fig: Hadoop Installation – Starting DataNode

### Start ResourceManager:

ResourceManager 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 NodeManagers and the each application's ApplicationMaster.

**Command:** ./yarn-daemon.sh start resourcemanager



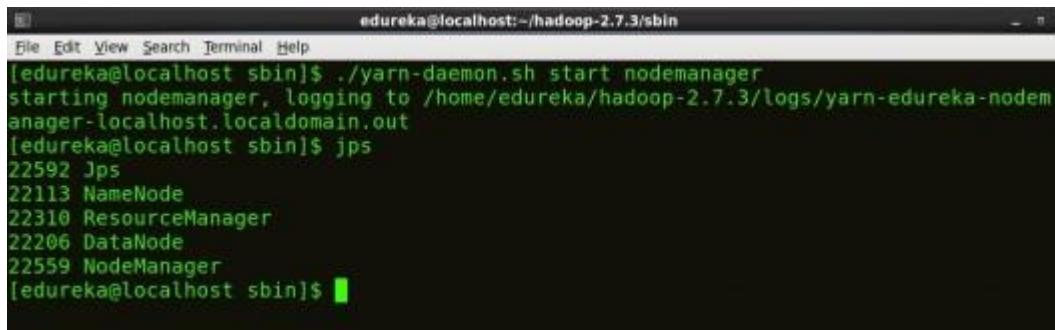
```
edureka@localhost sbin$ ./yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /home/edureka/hadoop-2.7.3/logs/yarn-edureka-resourcemanager-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22113 NameNode
22310 ResourceManager
22345 Jps
22206 DataNode
[edureka@localhost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

Fig: Hadoop Installation – Starting ResourceManager

### Start NodeManager:

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

**Command:** ./yarn-daemon.sh start nodemanager



```
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-nodemanager-localhost.localdomain.out
[edureka@localhost sbin]$ jps
22592 Jps
22113 NameNode
22310 ResourceManager
22206 DataNode
22559 NodeManager
[edureka@localhost sbin]$
```

*Fig: Hadoop Installation – Starting NodeManager*

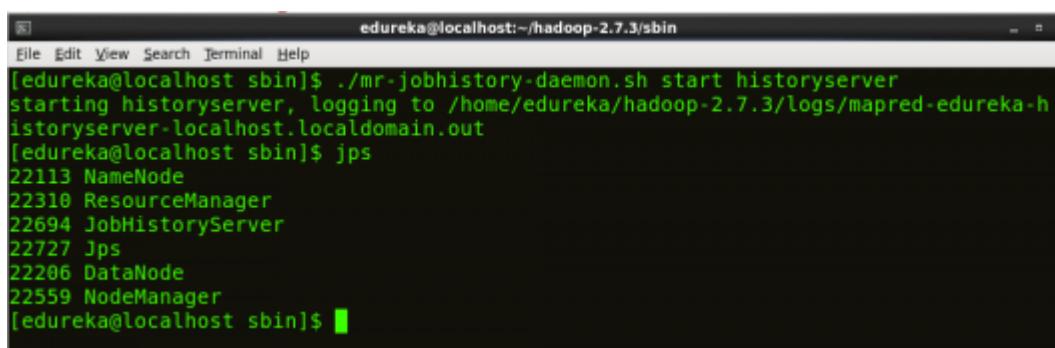
### **Start JobHistoryServer:**

JobHistoryServer is responsible for servicing all job history related requests from client.

**Command:** ./mr-jobhistory-daemon.sh start historyserver

**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-historyserver-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 NameNode interface.

The screenshot shows a Mozilla Firefox window titled "Namenode information - Mozilla Firefox". The address bar displays the URL "http://localhost:50070/dfshealth.html#tab-overview". The page content is titled "Overview 'localhost:9000' (active)". Below this, there is a table with the following data:

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

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

## RESULT:

Thus the Word Count Program to demonstrate the use of Map and Reduce Tasks was executed successfully.

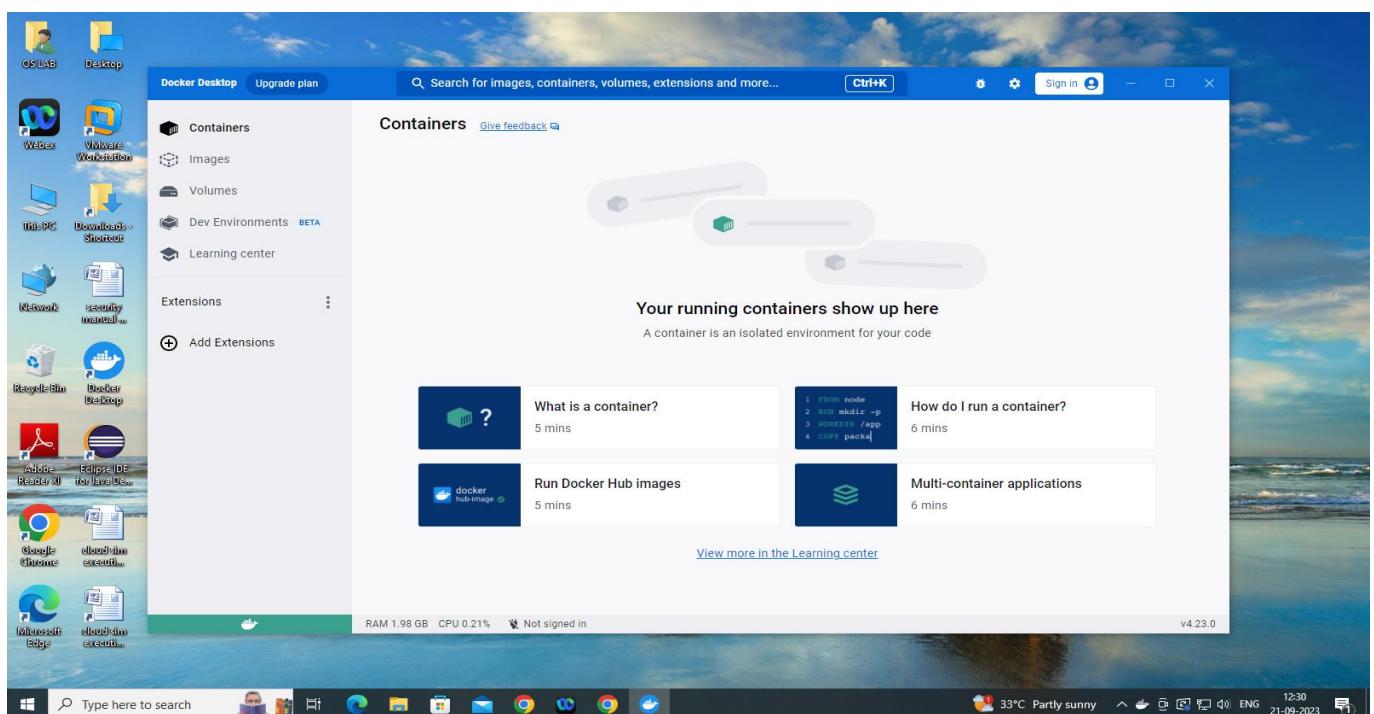
<b>EX. NO : 8</b>	<b>CREATING AND EXECUTING YOUR FIRST CONTAINER USING DOCKER</b>
-------------------	---

**AIM:**

To creating and executing your first container using Docker.

**PROCEDURE:**

- 1) To open the Google Chrome.
- 2) Type the text “**Docker for Windows 10**”.
- 3) Open the website “**Install Docker Desktop on Windows**”.
- 4) Press the option for “**Docker Desktop for Windows**”.
- 5) Then, download the Docker Engine for the Windows OS.
- 6) Then, complete the Installation procedure for Docker.
- 7) Next, Open the Command Prompt then type the command as **wsl --update**.
- 8) To open the Docker Engine.



- 9) Then, click “**What is a container ?**”

## (i) Containers on Docker Desktop

The screenshot shows the Docker Desktop interface. On the left, there's a sidebar with icons for Containers, Images, Volumes, and Networks. The main area is titled 'Containers' with a search bar at the top. It displays a single container named 'welcome-to-docker' with the image 'docker/welcome-to-docker:latest'. The container is listed as 'Running' with 0% CPU usage and 9.52MB memory usage. To the right, a 'What is a container?' panel provides introductory steps. At the bottom, a taskbar shows various application icons and the system tray.

## (ii) View the frontend

This screenshot shows the 'View the frontend' step in the Docker Desktop learning center. The main interface is identical to the previous one, showing the 'welcome-to-docker' container. The 'What is a container?' panel now includes a callout pointing to the 'Port(s)' column in the table, which lists '8088:80'. A mouse cursor is hovering over this link. The taskbar at the bottom remains the same.

### (iii) Explore your container

The screenshot shows the Docker Desktop interface. On the left, there's a sidebar with icons for Containers, Images, Volumes, and Compose. The main area is titled 'Containers' with a search bar and a chart showing CPU and memory usage. A table lists one container:

Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
welcome-to-docker d1caf79c49a8	docker/welcome-to-docker:latest	Running	0%	8088:80	32 seconds ago	[Stop, More]

To the right, a sidebar titled 'What is a container?' provides information and links to 'Explore your container', 'Stop your container', and 'What's next'. The taskbar at the bottom shows various application icons and the system tray.

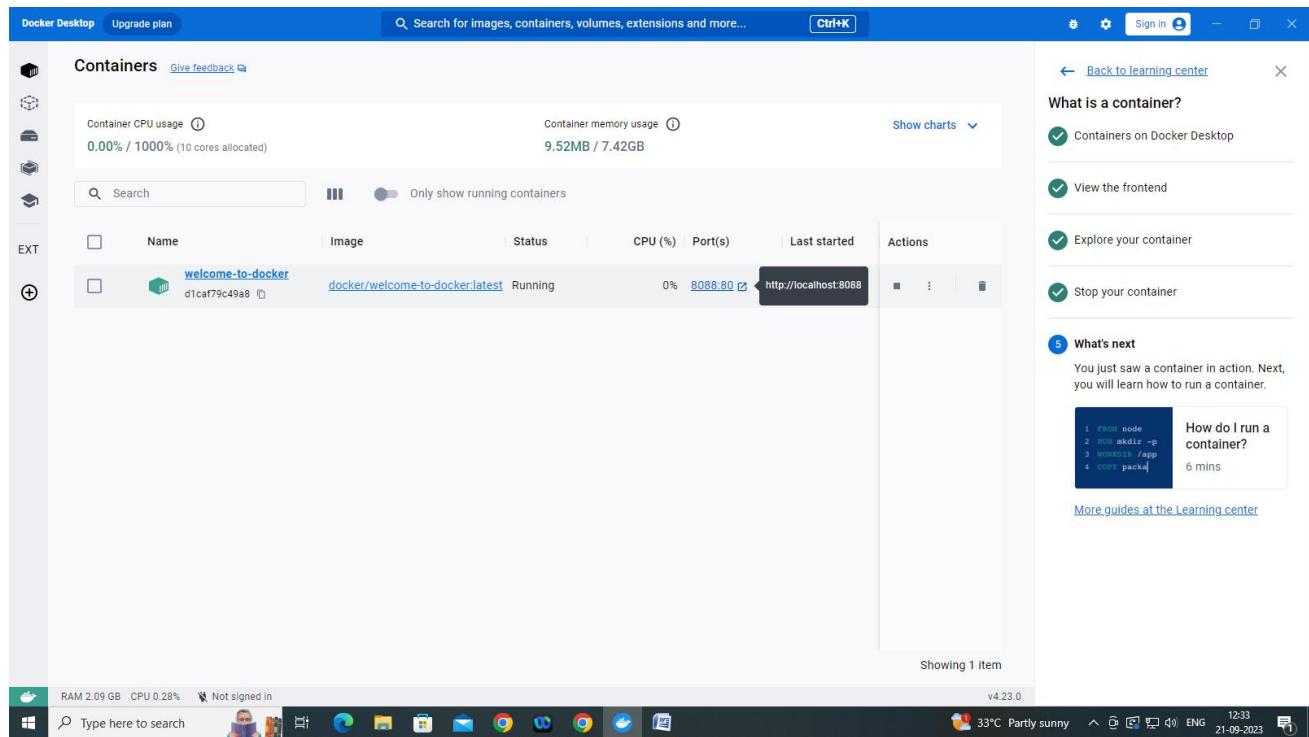
### (iv) Stop your container

This screenshot is identical to the previous one, except the container status has changed from 'Running' to 'Stopped'. The table now shows:

Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
welcome-to-docker d1caf79c49a8	docker/welcome-to-docker:latest	Stopped	0%	8088:80	42 seconds ago	[Start, More]

The sidebar and taskbar remain the same.

## 10) Then, click “How do I run a container ?”

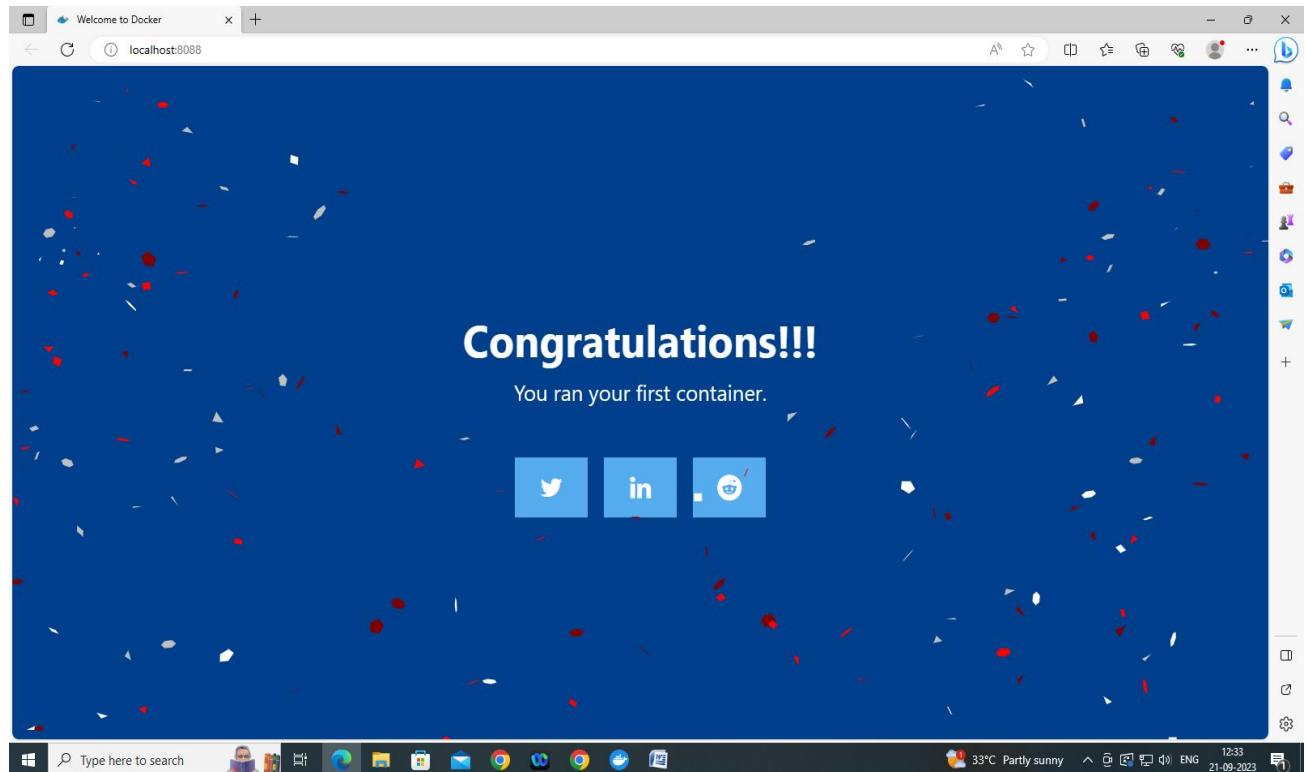


## The steps as follows:

- (i) Images are used to run containers
- (ii) Get the sample application
- (iii) Verify your Dockerfile
- (iv) Build your first image
- (v) Run your container
- (vi) View the Frontend

## OUTPUT:

- (i) Finally, we have successfully run the first container.
- (ii) The output will be displayed as below ,



OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

## RESULT:

Thus,to create the first container using Docker was executed successfully.

**EX. NO : 9**

## **RUN A CONTAINER FROM DOCKER HUB**

### **AIM:**

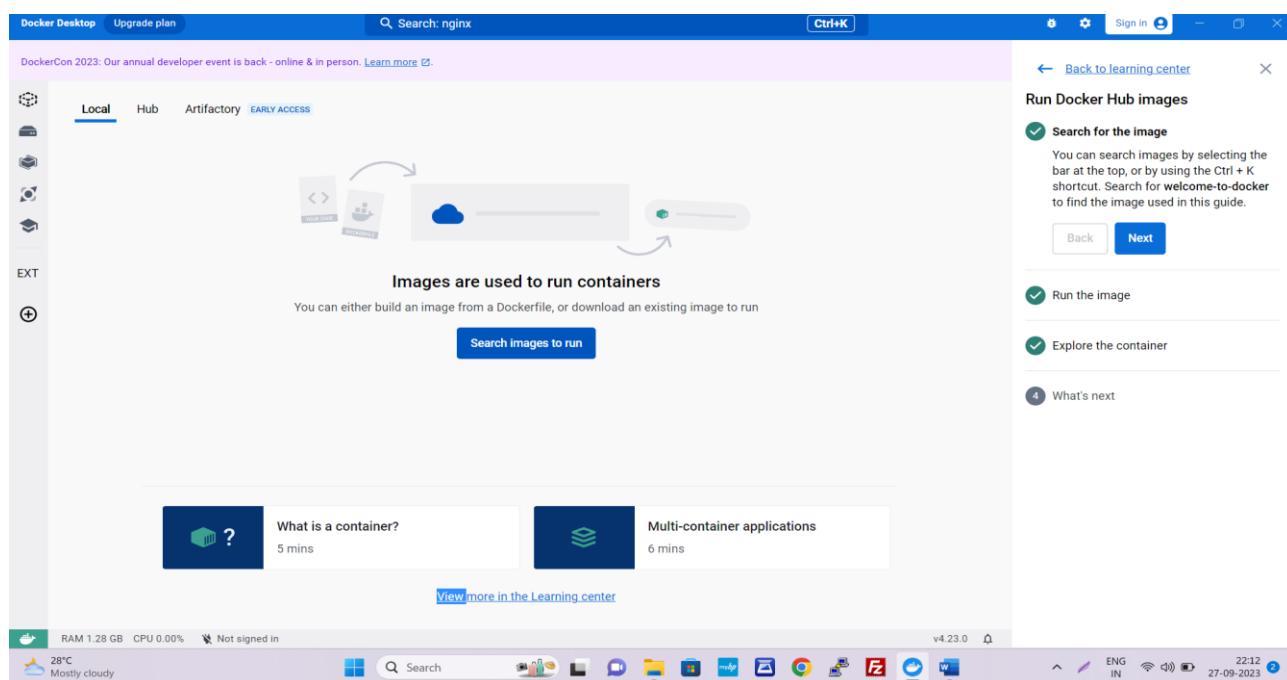
To run a Container from Docker Hub.

### **PROCEDURE:**

- 1) To open the Docker Engine.
- 2) Then,choose “**Run Docker Hub images**”.
- 3) The below mentioned steps as follows as to run a container from Docker Hub,

#### **STEP 1: Search for the image**

- (i) You can search images by selecting the bar at the top,or by using the **Ctrl+k** shortcut.
- (ii)Search for **welcome-to-docker** to find the image used in this guide.

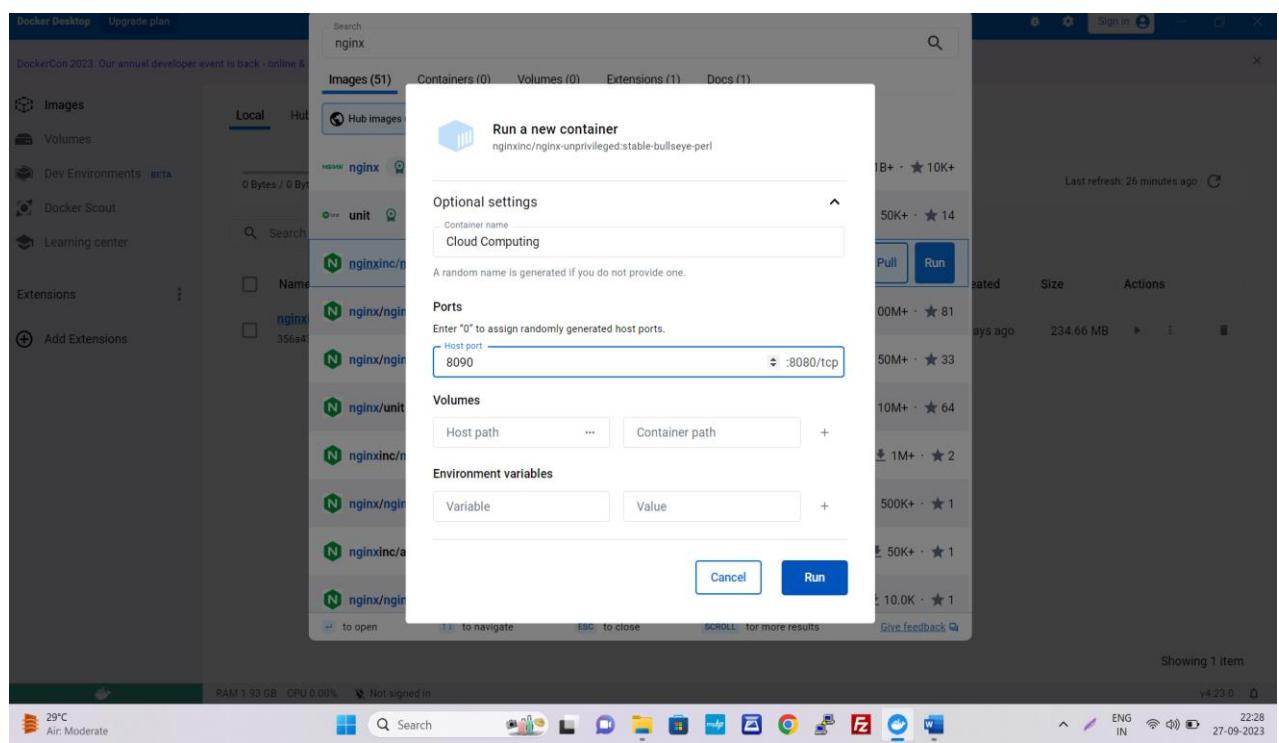
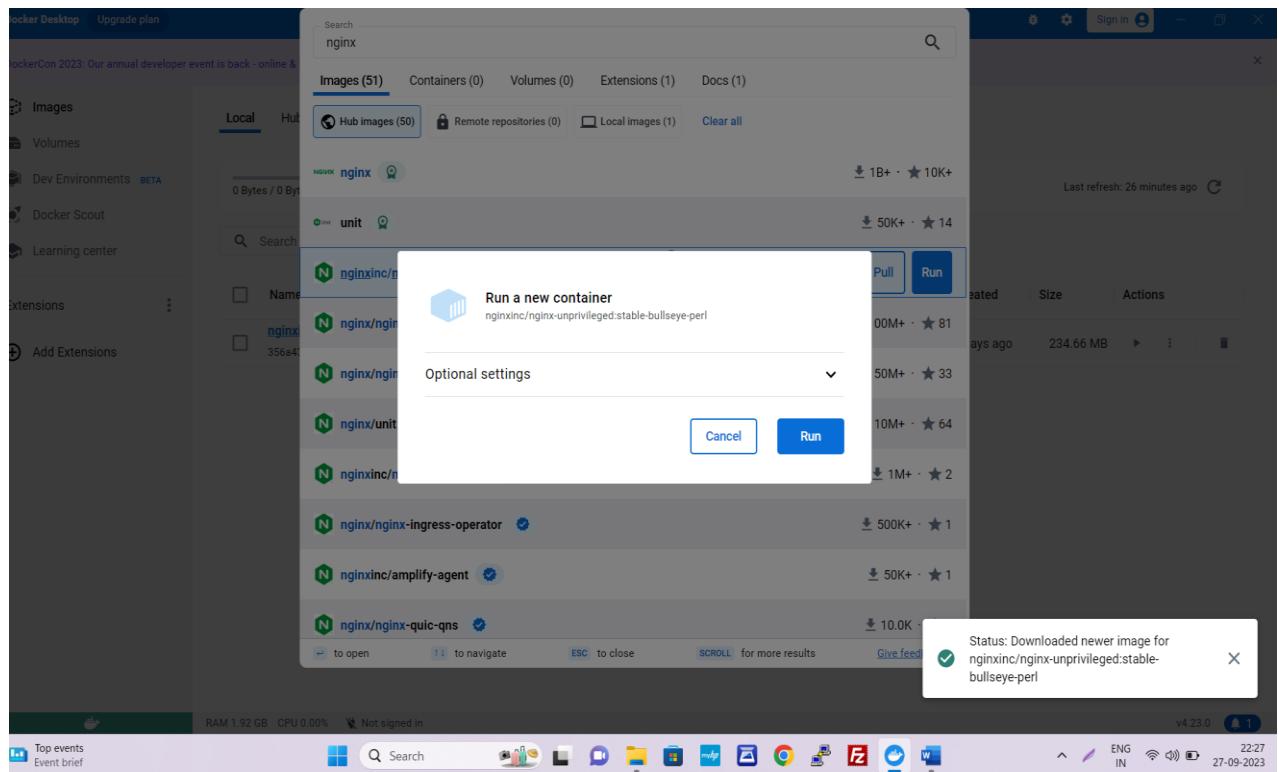


## STEP 2: Run the image

- (i) Select **Run**, when the **Optional Settings** appear, specify the **Host Port** number **8090** and then select **Run**.
- (ii) You can also select **View on Hub** to learn more about an image.

The screenshot shows the Docker Desktop interface. On the left, the 'Containers' section lists a single running container named 'welcome-to-docker'. On the right, a 'Walkthrough' panel titled 'Run Docker Hub images' provides instructions for running an image. It includes a step to 'Run the image' with a note about specifying a host port (8090) and a 'Optional settings' dialog. Below this, there's a 'Explore the container' section and a 'What's next' link. The taskbar at the bottom shows system status and a notification for v4.23.0.

This screenshot shows the Docker Desktop interface with the search bar set to 'nginx'. The 'Images (50)' section displays various nginx-related Docker images from the Docker Hub. A 'Run Docker Hub images' walkthrough is overlaid on the right, showing the 'Run the image' step with the host port set to 8090. The taskbar at the bottom shows system status and a notification for v4.23.0.



## STEP 3: Explore the Container

- (i) Go to the **Container** tab in Docker Desktop to view the container.

The screenshot shows the Docker Desktop interface with the 'Containers' tab selected. The main pane displays a single running container named 'welcome-to-docker' with the image 'docker/welcome-to-docker'. The container's status is 'Running', it has consumed 0% CPU and 9.62MB of memory, and was last started 3 minutes ago. A search bar and a filter for 'Only show running containers' are visible. To the right, a 'Learning center' sidebar provides links to 'Run Docker Hub images', 'Explore the container' (which is currently selected), and 'Walkthroughs'.

The screenshot shows the Docker Desktop interface with the 'Images' tab selected. The left sidebar includes 'Containers', 'Images' (which is selected), 'Volumes', 'Dev Environments BETA', 'Docker Scout', and 'Learning center'. Under 'Images', the 'Local' tab is active, showing two images: 'nginxinc/nginx-unprivileged' (stable-bullseye-perl) and 'docker/welcome-to-docker' (latest). Both images are unused and were created 3 days ago and 3 months ago respectively. The total size is 234.66 MB. A search bar and a refresh button are at the top of the images list. The bottom of the screen shows a taskbar with various application icons and system status.

Docker Desktop Upgrade plan

Search for images, containers, volumes, extensions and more... Ctrl+K

Containers Images Volumes Dev Environments BETA

Docker Scout Learning center

Extensions Add Extensions

nginxinc/nginx-unprivileged:stable-bullseye-perl  
356a43768d46

CREATED 3 days ago SIZE 234.66 MB Run Delete

Vulnerabilities

Give feedback

Layers (25)

Layer	Content	Size
0	ADD file:a1398394375faab8dd9e1e8d584eea96c750fb5...	74.76 MB
1	CMD ["bash"]	0 B
2	LABEL maintainer=NGINX Docker Maintainers <docker-m...	0 B
3	ENV NGINX_VERSION=1.25.2	0 B
4	ENV NJS_VERSION=0.8.0	0 B
5	ENV PKG_RELEASE=1~bookworm	0 B
6	ARG UID=101	0 B
7	ARG GID=101	0 B
8	RUN [2] UID=101 GID=101 /bin/sh -c set -x && groupadd ...	111.86 MB
9	RUN [2] UID=101 GID=101 /bin/sh -c sed -i 's,listen 80,listen 443, ...'	9.58 KB
10	COPY docker-entrypoint.sh / # buildkit	1.62 KB
11	COPY 10-listen-on-ipv6-by-default.sh /docker-entrypoint...	2.13 KB
12	COPY 15-local-resolvers.envsh /docker-entrypoint.d # bui...	276 B
13	COPY 20-envsubst-on-templates.sh /docker-entrypoint.d ...	3 KB

Analyzing image Starting analysis

RAM 1.55 GB CPU 0.00% Not signed in v4.23.0

29°C Partly cloudy Search

Docker Desktop Upgrade plan

Search: nginx Ctrl+K

Images Volumes Dev Environments BETA

Docker Scout Learning center

Image hierarchy

FROM debian:12-slim, 12.1-slim, bookworm-20230919-slim, bo...

ALL nginxinc/nginx-unprivileged:stable-bullseye-perl

Vulnerabilities (34) Packages (239)

Give feedback

FROM debian:12-slim

DOCKER OFFICIAL IMAGE

created 8 days ago View in GitHub

Packages: 126

nginxinc/nginx-unprivileged:stable-bullseye-perl

Packages: 239

Layers (25)

Layer	Content	Size
0	ADD file:a1398394375faab8dd9e1e8d584eea96c750fb5...	74.76 MB
1	CMD ["bash"]	0 B
2	LABEL maintainer=NGINX Docker Maintainers <docker-m...	0 B
3	ENV NGINX_VERSION=1.25.2	0 B
4	ENV NJS_VERSION=0.8.0	0 B
5	ENV PKG_RELEASE=1~bookworm	0 B
6	ARG UID=101	0 B
7	ARG GID=101	0 B
8	RUN [2] UID=101 GID=101 /bin/sh -c set -x && groupadd ...	111.86 MB
9	RUN [2] UID=101 GID=101 /bin/sh -c sed -i 's,listen 80,listen 443, ...'	9.58 KB
10	COPY docker-entrypoint.sh / # buildkit	1.62 KB

RAM 1.98 GB CPU 0.35% Not signed in v4.23.0

29°C Partly cloudy Search

Docker Desktop Upgrade plan

Search: nginx

DockerCon 2023: Our annual developer event is back - online & in person. [Learn more](#)

Images (2) Vulnerabilities (34) Packages (239)

Image hierarchy

Layers (25)

Package Vulnerabilities

RAM 0.00 GB CPU 0.00% Not signed in v4.23.0

29°C Partly cloudy

22:33 27-09-2023

Package	Vulnerabilities
> debian/tiff 4.5.0-6	0 H 0 M 6 L
> debian/openldap 2.5.13+dfsg-5	0 H 0 M 4 L
> debian/tar 1.34+dfsg-1.2	0 H 0 M 2 L
> debian/shadow 1.4.13+dfsg1-1	0 H 0 M 2 L
> debian/perl 5.36.0-7	0 H 0 M 2 L
> debian/openssl 3.0.9-1	0 H 0 M 2 L
> debian/glibc 2.36-9+deb12u1	0 H 0 M 2 L
> debian/util-linux 2.38.1-5	0 H 0 M 1 I

Docker Desktop Upgrade plan

Search: nginx

DockerCon 2023: Our annual developer event is back - online & in person. [Learn more](#)

Images (2) Vulnerabilities (34) Packages (239)

Image hierarchy

Layers (25)

Package

RAM 0.00 GB CPU 0.00% Not signed in v4.23.0

29°C Partly cloudy

22:34 27-09-2023

Package	Fixes
> debian/abseil 20220623.1-1	0 H 0 M 1 I
> debian/acl 2.3.1-3	
> debian/adduser 3.134	
> debian/aom 3.6.0-1	
> debian/apt 2.6.1	
> debian/attr 1:2.5.1-4	
> debian/audit 1:3.0.9-1	
> debian/base-files 12.4+deb12u1	

OUTPUT	05	
VIVA	05	
RECORD	05	
TOTAL	15	

## **RESULT:**

Thus ,from the Docker Hub to create a new container and the Container was runned successfully.