EX.NO:7

Install Docker on your machine

Aim:

To Install Docker on Ubuntu system.

Procedure:

First, update your packages:

$ sudo apt update

Next, install docker with apt-get:

$ sudo apt install docker.io

Finally, verify that Docker is install**ed correctly:**

$ sudo docker run hello-world

2. Create your project

In order to create your first Docker application, I invite you to create a folder on your computer. It must contain the following two files:

A '*main.py*' file (python file that will contain the code to be executed).

A '*Dockerfile*' file (Docker file that will contain the necessary instructions to create the environment).

Normally you should have this folder architecture:

. ├── Dockerfile

└── main.py

0 directories, 2 files

3. Edit the Python file

Once your code is ready and the Dockerfile is written, all you have to do is create your image to contain your application.

$ docker build -t python-test .

The '*-t*' option allows you to define the name of your image. In our case we have chosen '*python-test*' but you can put what you want.

5. Run the Docker image

Once the image is created, your code is ready to be launched.

$ docker run python-test

You need to put the name of your image after '*docker run*'.

There you go, that's it. You should normally see "Docker is magic!" displayed in your terminal.

Code is available

If you want to retrieve the complete code to discover it easily or to execute it, I have put it at your disposal on my GitHub.

-> GitHub: Docker First Application example

Useful commands for Docker

List your images.

$ docker image ls

Delete a specific image.

$ docker image rm [image name]

Delete all existing images.

$ docker image rm $(docker images -a -q)

List all existing containers (running and not running).

$ docker ps -a

Stop a specific container.

$ docker stop [container name]

Stop all running containers.

$ docker stop $(docker ps -a -q)

Delete a specific container (only if stopped).

$ docker rm [container name]

Delete all containers (only if stopped).

$ docker rm $(docker ps -a -q)

Display logs of a container.

$ docker logs [container name]

Result:

Thus the Docker was installed successfully.

EX.NO:8

Run a container

Aim:

To write a procedure to run a container from docker.

Procedure:

Step 1: Get the sample application

If you have git, you can clone the repository for the sample application. Otherwise, you can download the sample application. Choose one of the following options.

Step 2: Explore the Dockerfile

To run your code in a container, the most fundamental thing you need is a Dockerfile. A Dockerfile describes what goes into a container. Open the sample application in your IDE and then open the Dockerfile to explore its contents. Note that this project already has a Dockerfile, but for your own projects you need to create a Dockerfile. A Dockerfile is simply a text file named Dockerfile with no file extension.

Step 3: Build your first image

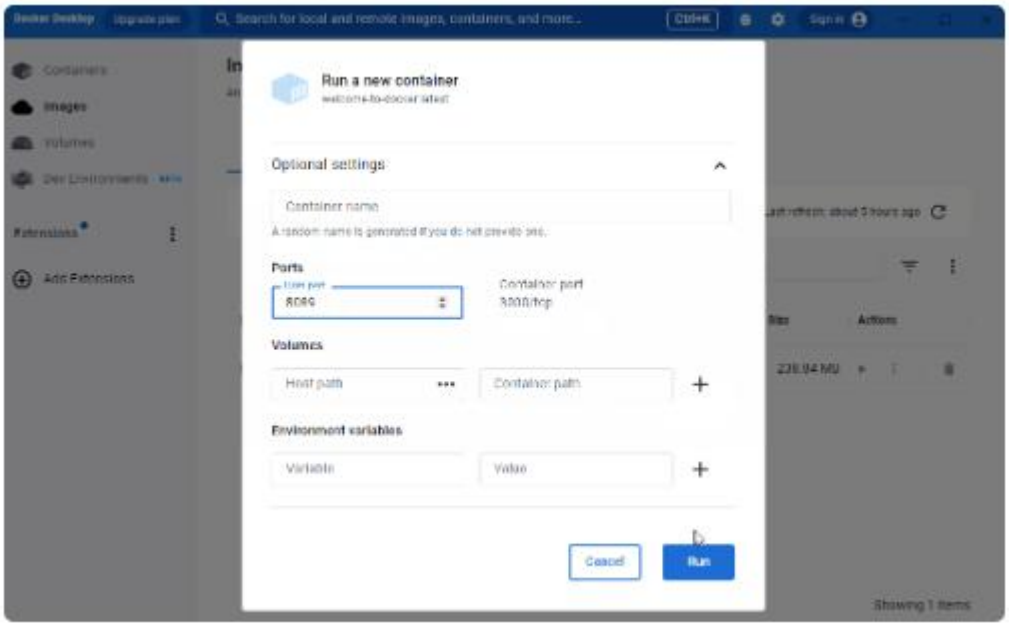
An image is like a static version of a container. You always need an image to run a container. Once you have a Dockerfile in your repository, run the following docker build command in the project folder to create an image.

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

Building the image may take some time. After your image is built, you can view your image in the **Images** tab in Docker Desktop.

Step 4: Run your container

To run your image as a container, go to the **Images** tab, and then select **Run** in the **Actions** column of your image. When the **Optional settings** appear, specify the **Host port** number 8089 and then select **Run**.



Result:

Thus the container from Docker hub was executed successfully.

CONTENT BEYOND SYLLABUS

1.Calculator webservice

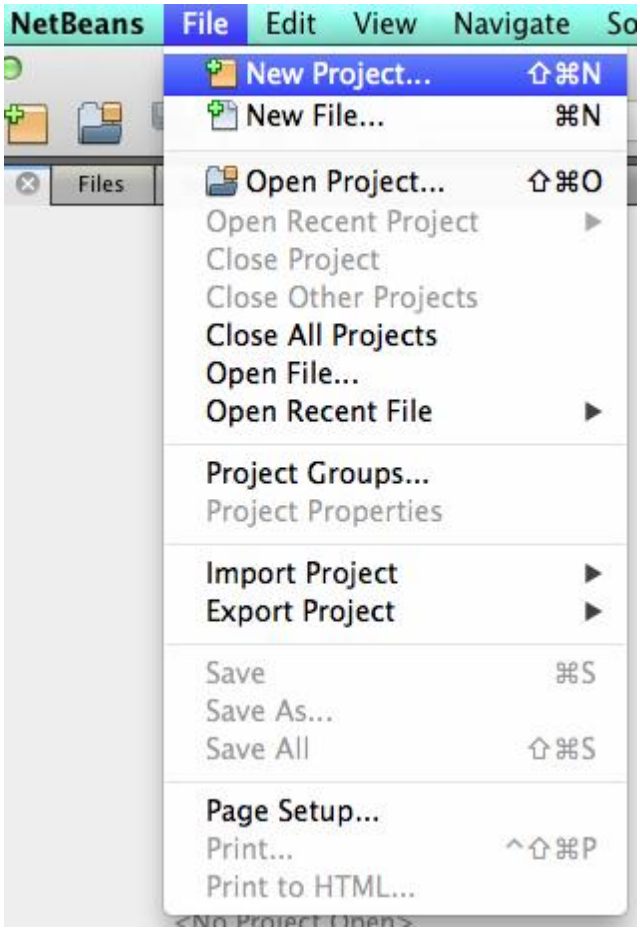
AIM:

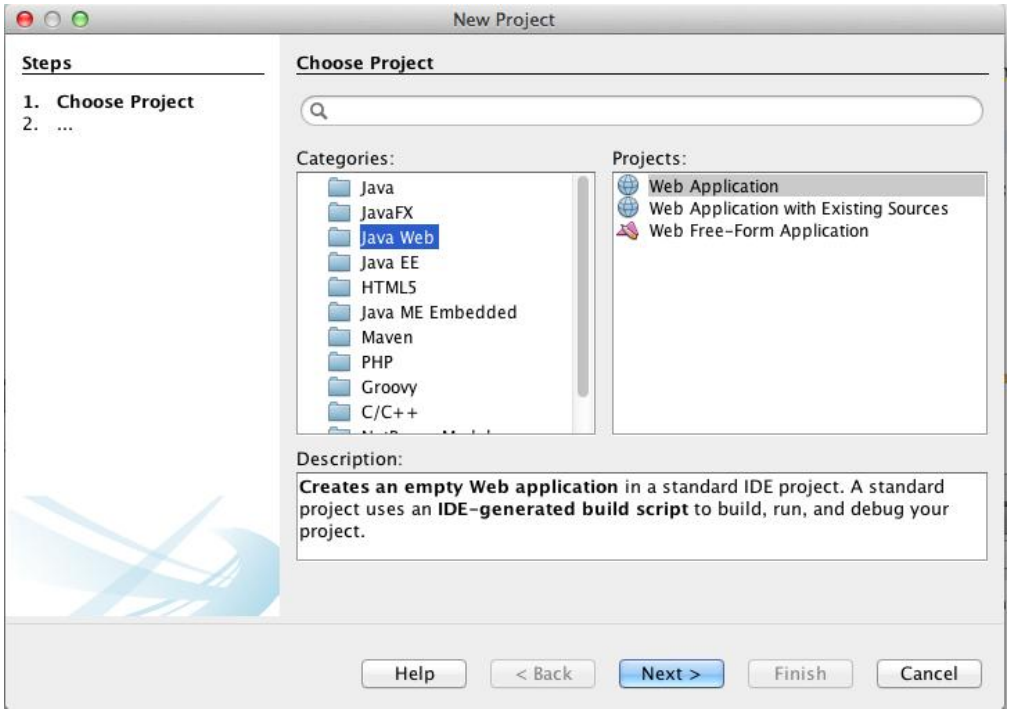
To create and deploy a calculator web service in Java using NetBeans IDE

Step 1: Create a Java Web Project

Open NetBeans IDE

Click on New Project and choose Java Web → Web Application

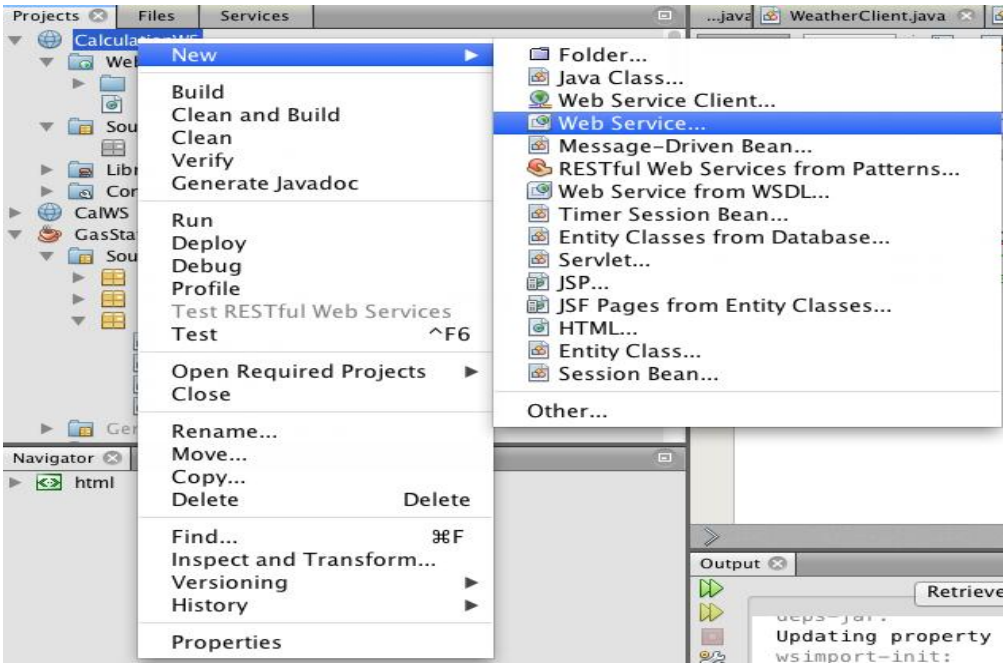




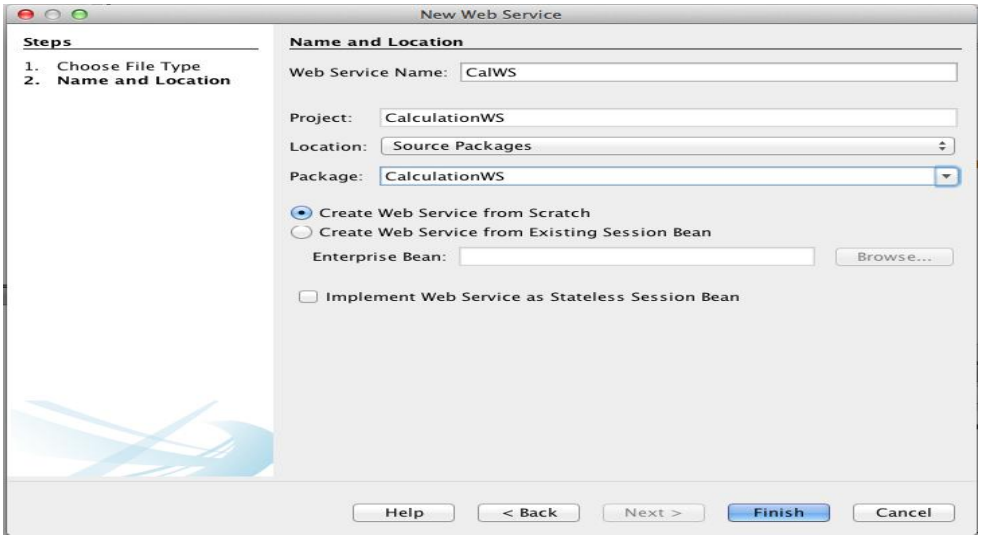
Enter the Project Name: CalculationWS, using the default settings and then click on "Finish".

Step 2: Create a Web Service

Now go to the Project Tree Structure on the left side of the window.
Right click on the project and select "New" and then choose "Web Service"



Specify web service name “CalWS” and package name “CalculationWS”. Click on “Finish”.



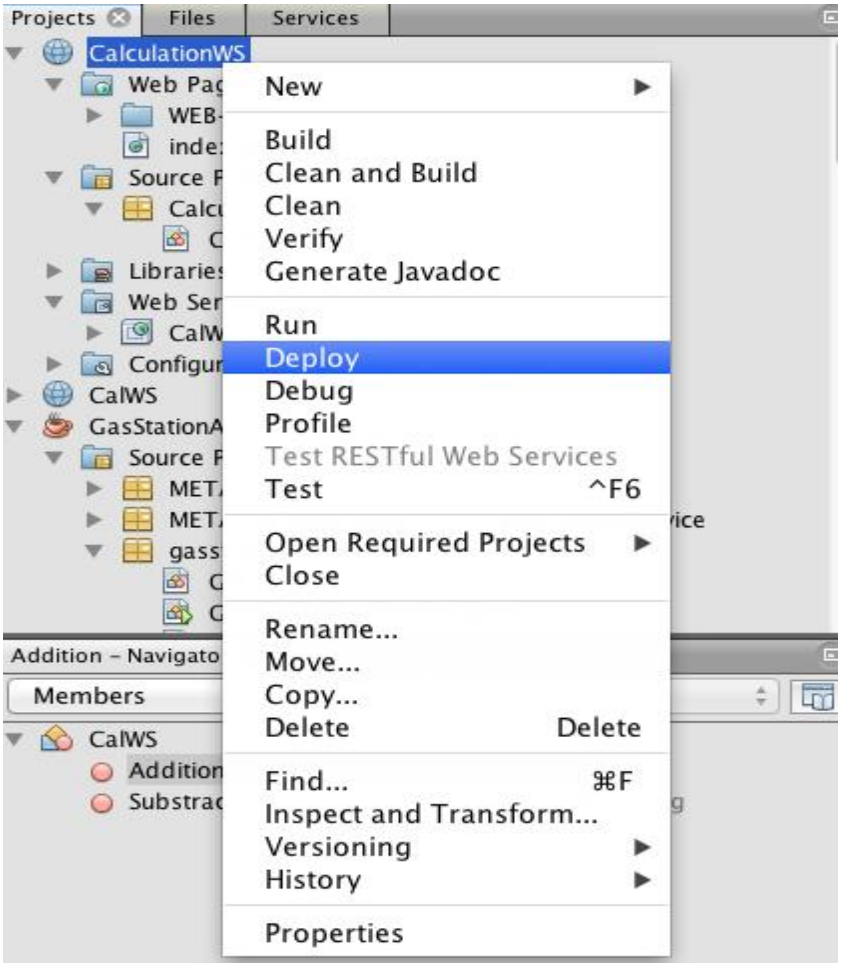
Open CalWS.java file, replace the original hello() function with the following code:

```
@WebMethod(operationName = "Addition")
public String Addition(@WebParam(name="value1") String value1,@WebParam(name = "value2") String value2 ) {
float value=Float.valueOf(value1)+Float.valueOf(value2);
return (Float.toString(value));
}
```

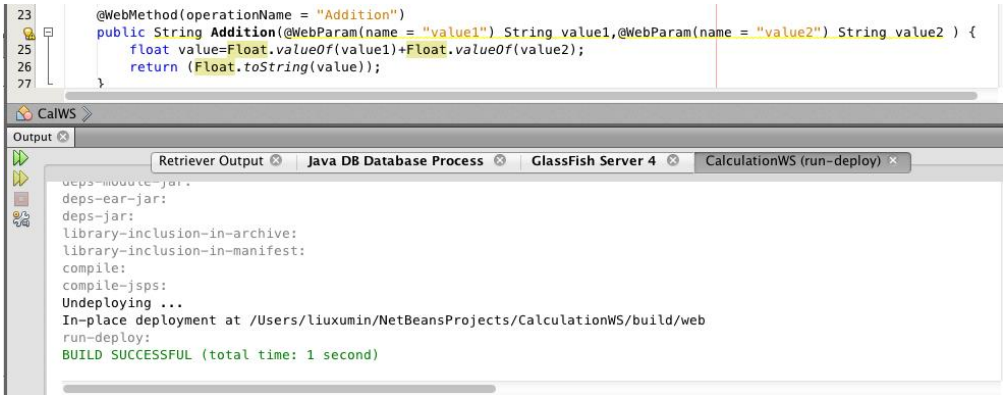
Now the web service is created.

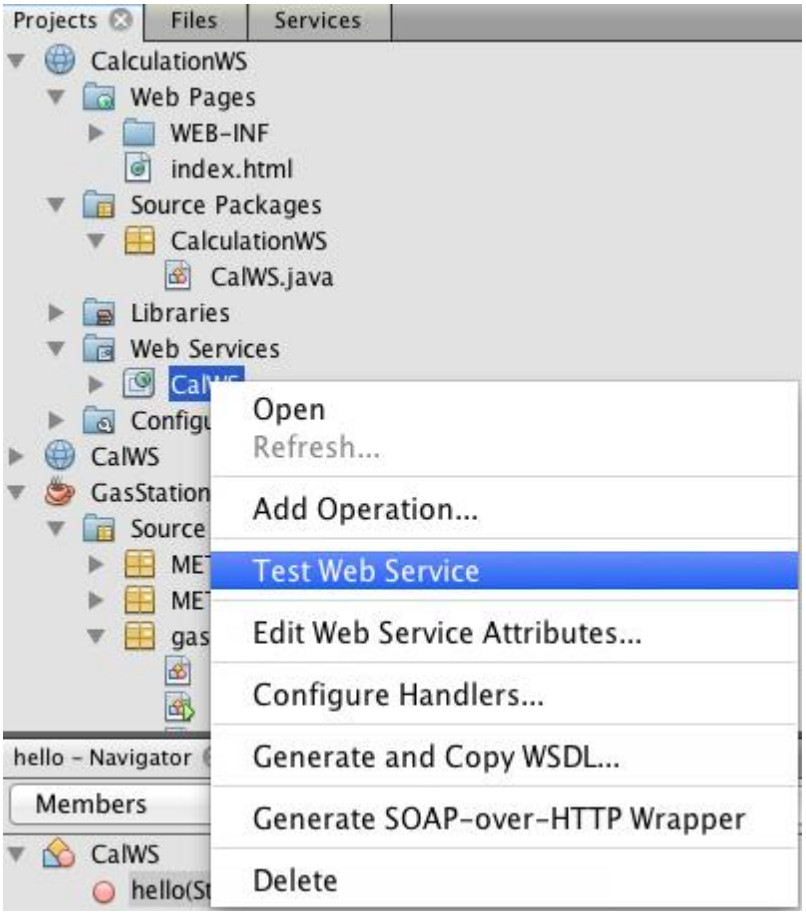
Step 3: Deploy and Test Web Service

Right click on the project and select “Deploy”



This is to deploy all the web services in this project. If success, you will see:





You will see:

Result:

Thus the calculator web service was executed successfully.

Ex. 2 CREATING VIRTUAL MACHINES

Aim

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

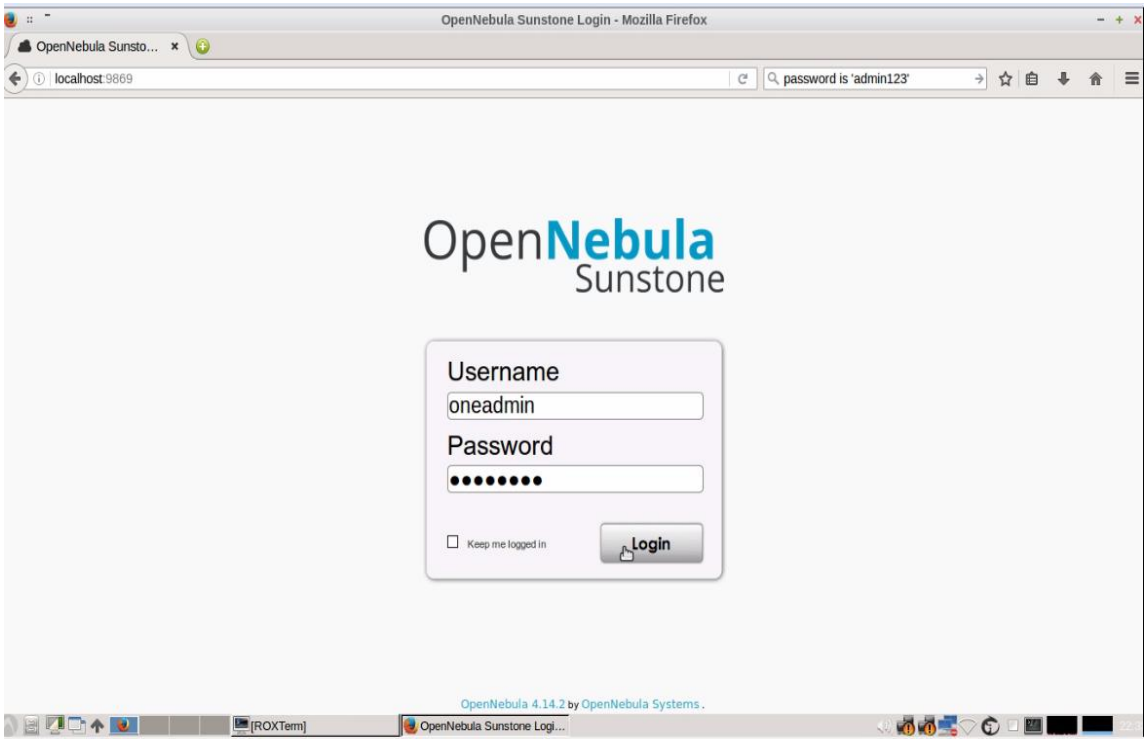
Procedure

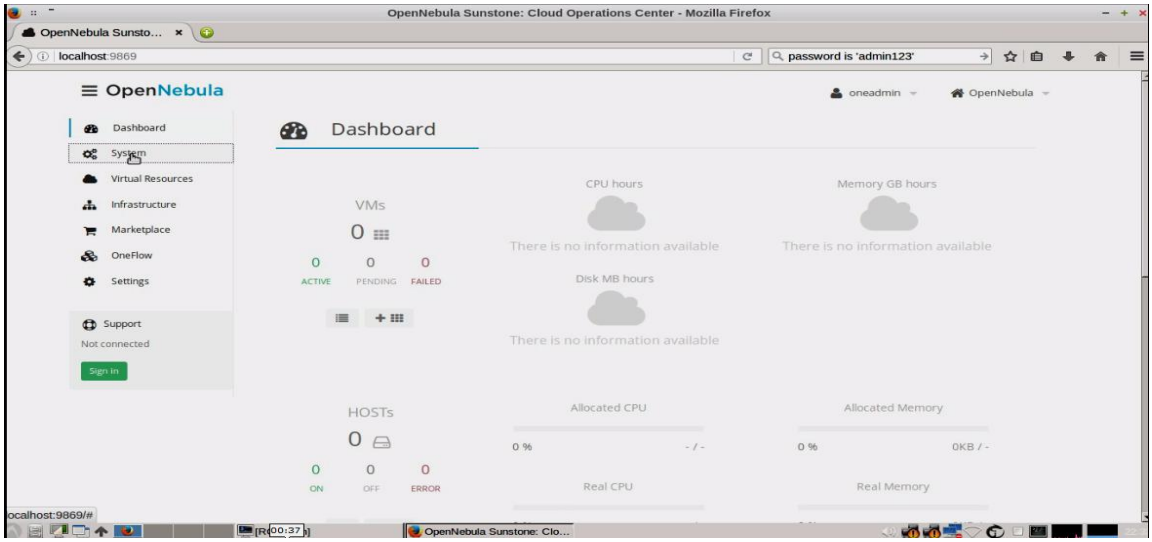
1. Check the service status for OpenNebula, OpenNebula Sunstone, and NFS(Network File System)

```
gcclab@gcc-server:~$ sudo bash
root@gcc-server:~# service opennebula status
 * one is running
root@gcc-server:~# service opennebula-sunstone status
 * sunstone-server is running
root@gcc-server:~# service nfs-kernel-server status
nfsd running
root@gcc-server:~#
```

   Incase if the above services are not running, restart them.

2. Log into OpenNebula sunstone web interface for managing OpenNebula cloud services.





3. Manage the OpenNebula using Command line interface(CLI) as 'oneadmin' user.

```
File Edit View Search Preferences Tabs Help
gcclab@gcc-server:~$ sudo bash
root@gcc-server:~# su - oneadmin
oneadmin@gcc-server:~$ pwd
/var/lib/one
oneadmin@gcc-server:~$ onehost list
  ID NAME            CLUSTER   RVM      ALLOCATED_CPU      ALLOCATED_MEM STAT

oneadmin@gcc-server:~$ onevnet list
  ID USER            GROUP       NAME              CLUSTER    BRIDGE   LEASES

oneadmin@gcc-server:~$ oneimage list
  ID USER       GROUP      NAME           DATASTORE     SIZE TYPE PER STAT RVMS

oneadmin@gcc-server:~$ onetemplate list
  ID USER           GROUP          NAME                                REGTIME

oneadmin@gcc-server:~$ onevm list
    ID USER     GROUP    NAME           STAT UCPU    UMEM HOST             TIME

oneadmin@gcc-server:~$ oneuser list
  ID NAME            GROUP      AUTH          VMS           MEMORY        CPU
   0 oneadmin        oneadmin   core            -              -          -
   1 serveradmin     oneadmin   server_c   0 /   -      0M /      -   0.0 /   -
oneadmin@gcc-server:~$ onegroup list
  ID NAME                          USERS       VMS           MEMORY        CPU
   0 oneadmin                          2         -              -          -
   1 users                             0   0 /   -      0M /      -   0.0 /   -
oneadmin@gcc-server:~$
oneadmin@gcc-server:~$ onehost create localhost -i kvm -v kvm -n dummy
ID: 0
oneadmin@gcc-server:~$ onehost list
  ID NAME            CLUSTER   RVM      ALLOCATED_CPU      ALLOCATED_MEM STAT
   0 localhost       -             0                -                - init
oneadmin@gcc-server:~$
```

5. Create Virtual Network using ‘onevnet’ command using (mynetwork.one) script file.

```
oneadmin@gcc-server:~$ cat mynetwork.one
NAME = "private"

BRIDGE = br0

AR = [
       TYPE = IP4,
       IP = 192.168.0.100,
       SIZE = 100
    ]
```

```
oneadmin@gcc-server:~$ onevnet list
  ID USER            GROUP         NAME                CLUSTER    BRIDGE   LEASES

oneadmin@gcc-server:~$ onevnet create mynetwork.one
ID: 0
oneadmin@gcc-server:~$ onevnet list
  ID USER            GROUP         NAME                CLUSTER    BRIDGE   LEASES
   0 oneadmin        oneadmin      private             -          br0           0
oneadmin@gcc-server:~$
```

5. Create Virtual disk image for CentOS 6.5(64 bit) using ‘oneimage’ command.

```
oneadmin@gcc-server:~$ oneimage create --name "CentOS-6.5_x86_64" --path "http://localhost/cloudimages/centos-6.5.qcow2.gz"
--driver qcow2 --datastore default
ID: 0
oneadmin@gcc-server:~$ oneimage list
  ID USER       GROUP      NAME           DATASTORE     SIZE TYPE PER STAT RVMS
   0 oneadmin   oneadmin   CentOS-6.5_x86_ default       267M OS    No rdy     0
oneadmin@gcc-server:~$
```

6. Create Virtual machine template for CentOS 6.5 (64 bit) using ‘onetemplate’ command.

```
ambiguous option: -nic
oneadmin@gcc-server:~$ onetemplate create --name "CentOS-6.5" --cpu 1 --vcpu 1 --memory 512 --arch x86_64 --disk "CentOS-6.5
_x86_64" --nic "private" --vnc --ssh
ID: 0
oneadmin@gcc-server:~$ onetemplate list
  ID USER            GROUP         NAME                                REGTIME
   0 oneadmin        oneadmin      CentOS-6.5                  06/27 00:58:27
```

7. Update the Template for ‘oneadmin’ user with the SSH public key using ‘oneuser’ command and Sunstone web interface.

```
oneadmin@gcc-server:~$ cat /var/lib/one/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCpZ7VExltM+8w36OsQZdzBsINiIRTBqU6934vS2wIRZvhjzT4RO6QS314gG3K0ghFk4cVAlS8ykMMjqW11G0L
IIqMYaKUYOG4oWfiB2hkeQoGGJCPhjMzsz3RKXkOsn/bzgo2iYXldiCTVLaj5d+c8ZxXHIErCK0K3AM2JYoeN/iR88nP6h8vCdJwaahpcysggpKyHTAsJ+TBaXF
8TGhVH9W0AAw6qM/OA2+FNKqCnR+b57KI7fXeBBVc/MckJfjI5PQXm+ZDrKa2LtFV9L5f71VvOmc8YWIBmDfZ2Bx/FcHuCEphq7Sh8WLNrLuqNW+Kf9lRcr33DBY
IROm9w2B root@gcc-server
oneadmin@gcc-server:~$ EDITOR=vi oneuser update oneadmin
Broken pipe
```

```
File Edit View Search Preferences Tabs Help
TOKEN_PASSWORD="b7e1a8486cca3d6070f88c6f79a5c4cac99c424a"
SSH_PUBLIC_KEY="ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCpZ7VExltM+8w36OsQZdzBsINiIRTBqU6934vS2wIRZvhjzT4RO6QS314gG3K0ghFk4cVA
lS8ykMMjqW11G0LtIIqMYaKUYOG4oWfiB2hkeQoGGJCPhjMzsz3RKXkOsn/bzgo2iYXldiCTVLaj5d+c8ZxXHIErCK0K3AM2JYoeN/iR88nP6h8vCdJwaahpcysg
gpKyHTAsJ+TBaXFl3TGhVH9W0AAw6qM/OA2+FNKqCnR+b57KI7fXeBBVc/MckJfjI5PQXm+ZDrKa2LtFV9L5f71VvOmc8YWIBmDfZ2Bx/FcHuCEphq7Sh8WLNrL
qNW+Kf9lRcr33DBYIROm9w2B root@gcc-server"
```

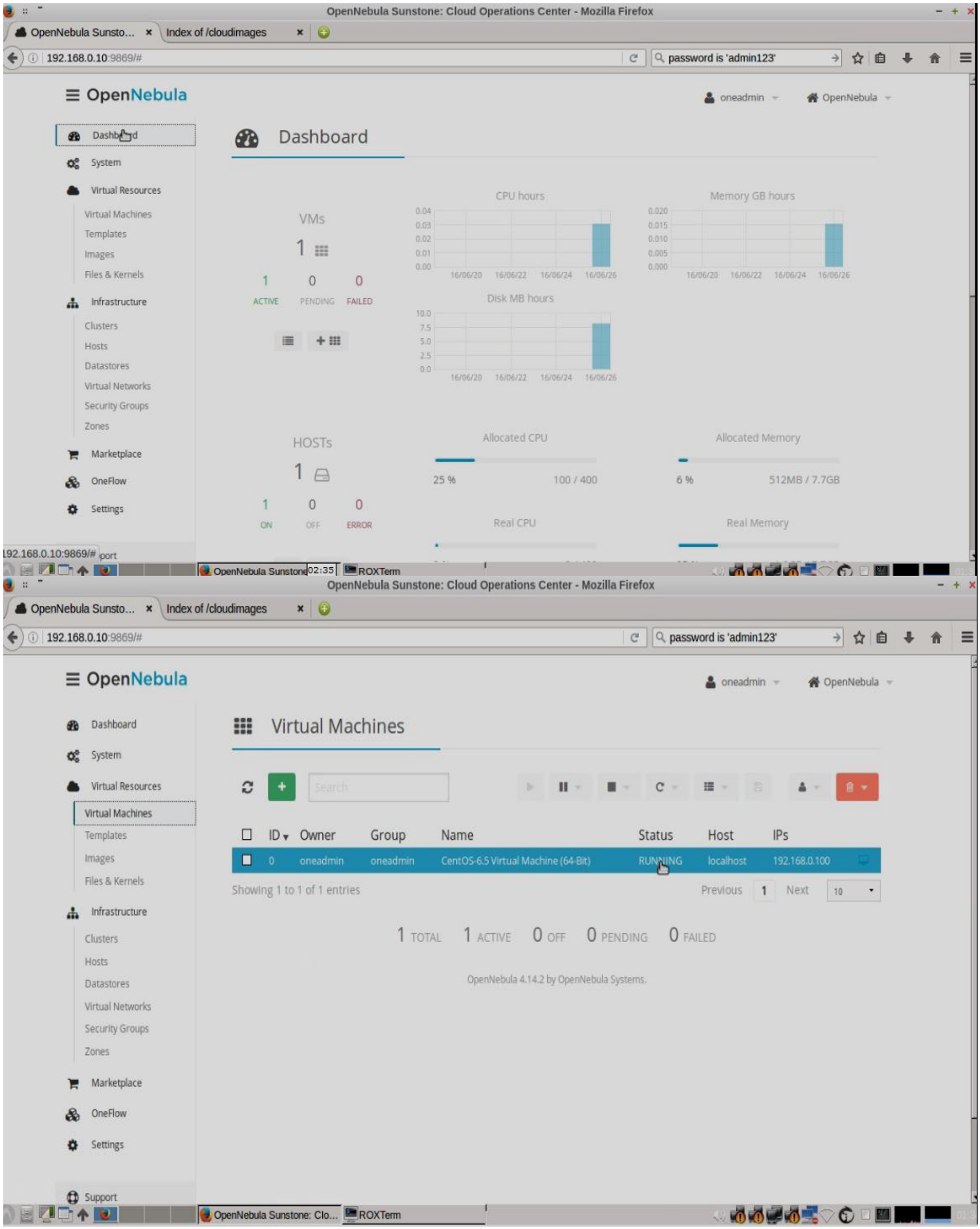
8. Create VM for CentOS 6.5 (64 bit) from the existing CentOS 6.5 template using ‘onetemplate’ command.

```
File Edit View Search Preferences Tabs Help
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME

oneadmin@gcc-server:~$ onetemplate instantiate "CentOS-6.5"  --name "CentOS-6.5 Virtual Machine (64-Bit)"
VM ID: 0
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS-6.5 Virt pend    0      0K              0d 00h00
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS-6.5 Virt prol    0      0K localhost    0d 00h00
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS-6.5 Virt runn    0      0K localhost    0d 00h00
oneadmin@gcc-server:~$
oneadmin@gcc-server:~$ onevm show 0
VIRTUAL MACHINE 0 INFORMATION
ID                  : 0
NAME                : CentOS-6.5 Virtual Machine (64-Bit)
USER                : oneadmin
GROUP               : oneadmin
STATE               : ACTIVE
LCM_STATE           : RUNNING
RESCHED             : No
HOST                : localhost
CLUSTER ID          : -1
CLUSTER             : default
START TIME          : 06/27 01:32:03
END TIME            : -
DEPLOY ID           : one-0

VIRTUAL MACHINE MONITORING
CPU                 : 0.0
MEMORY              : 512M
NETTX               : 0K
NETRX               : 10K

PERMISSIONS
```



Result

Thus the procedure to run the virtual machine of different configuration and to check how many virtual machines can be utilized at particular time was executed successfully.

Ex. 3 ATTACHING A VIRTUAL BLOCK

Aim

To find the procedure to attach virtual block to the virtual machine and to check whether it holds the data even after the release of the virtual machine.

Procedure

1. Use CentOS 6.5(64-Bit) virtual machine to check for the presence of Block device.

```
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS 6.5 (64- runn  0.0      0K localhost     0d 01h42
```

```
oneadmin@gcc-server:~$
oneadmin@gcc-server:~$ onevm show 0
VIRTUAL MACHINE 0 INFORMATION
ID                  : 0
NAME                : CentOS 6.5 (64-Bit) Virtual Machine
USER                : oneadmin
GROUP               : oneadmin
STATE               : ACTIVE
LCM_STATE           : RUNNING
RESCHED             : No
HOST                : localhost
CLUSTER ID          : -1
CLUSTER             : default
START TIME          : 07/09 12:37:41
END TIME            : -
DEPLOY ID           : one-0

VIRTUAL MACHINE MONITORING
CPU                 : 100.9
MEMORY              : 1024M
NETTX               : 0K
NETRX               : 5K

PERMISSIONS
OWNER               : um-
GROUP               : ---
oneadmin@gcc-server:~$ ssh root@192.168.0.100
Warning: Permanently added '192.168.0.100' (RSA) to the list of known hosts.
Last login: Sat Jul  9 08:45:48 2016 from 192.168.0.10
[root@localhost ~]# cat /etc/issue
CentOS release 6.5 (Final)
Kernel \r on an \m
```

2. Now we are inside the CentOS system. The block sda is for OS. There is no disk named sdb. So create a DATABLOCK (Virtual Block) and attach it to this CentOS 6.5 Virtual machine as /dev/sdb.

```
[root@localhost ~]#
[root@localhost ~]# fdisk -l /dev/sda

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000c55ce

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          64      512000   83  Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2              64        1306     9972736   8e  Linux LVM
[root@localhost ~]#
```

3. The virtual block cannot be created when the virtual machine is running. So exit and poweroff the virtual machine before attaching the virual block.

```
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS 6.5 (64- poff  0.0      0K localhost    0d 01h47
```

4. Create a DATABLOCK (Virtual Block) using ‘oneimage’

oneimage create –d 1 –name data –type DATABLOCK –size 20G –fstype
ext4

```
oneadmin@gcc-server:~$ oneimage create -d 1 --name data  --type  DATABLOCK  --size 20G  --fstype ext4
ID: 5
oneadmin@gcc-server:~$ oneimage list
  ID USER       GROUP      NAME            DATASTORE     SIZE TYPE PER STAT RVMS
   0 oneadmin   oneadmin   CentOS-6.5_x86_ default      267M OS    No used    1
   1 oneadmin   oneadmin   Ubuntu-14.04_am default      287M OS    No used    1
   2 oneadmin   oneadmin   Debian-7_amd64  default      167M OS    No used    1
   3 oneadmin   oneadmin   Devuan-1.0.0_am default      686M OS    No used    1
   5 oneadmin   oneadmin   data            default       20G DB    No rdy     0
```

Now, we have created a DATABLOCK named ‘data’ of size ‘20G’.

5. Now attach the DATABLOCK to the CentOS 6.5 (64-bit) virtual machine using ‘onevm’ command.

onevm disk –attach 0 - - image data

6. The DATABLOCK of 20GB size is attached to CentOS. Now power on the virtual machines and create partition/format/mount.

```
oneadmin@gcc-server:~$ onevm resume 0
oneadmin@gcc-server:~$ onevm list
  ID USER     GROUP    NAME            STAT UCPU    UMEM HOST             TIME
   0 oneadmin oneadmin CentOS 6.5 (64- runn  0.0      0K localhost    0d 01h52
```

```
[root@localhost ~]# fdisk -l /dev/sda

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000c55ce

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          64      512000   83  Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2              64        1306     9972736   8e  Linux LVM
[root@localhost ~]# fdisk -l /dev/sdb

Disk /dev/sdb: 21.5 GB, 21474836992 bytes
255 heads, 63 sectors/track, 2610 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

7. Partition the DATABLOCK using #fdisk /dev/sdb then press ‘p’ and then ‘n’ for new partition with number

1.

```
[root@localhost ~]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x0a21ffe6.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
         switch off the mode (command 'c') and change display units to
         sectors (command 'u').

Command (m for help): p

Disk /dev/sdb: 21.5 GB, 21474836992 bytes
255 heads, 63 sectors/track, 2610 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0a21ffe6

   Device Boot      Start         End      Blocks   Id  System

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-2610, default 1): 1
Last cylinder, +cylinders or +size{K,M,G} (1-2610, default 2610):
```

8. Partition the DATABLOCK using the command

#mkfs.ext4 /dev/sdb/

9. Now the DATABLOCk is ready for use. So mount it before using it with the command

#mount /dev/sdb1 /data

```
[root@localhost ~]#
[root@localhost ~]# mount /dev/sdb1  /data
[root@localhost ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-lv_root
                      8.5G  841M  7.2G  11% /
tmpfs                 499M     0  499M   0% /dev/shm
/dev/sda1             485M   54M  407M  12% /boot
/dev/sdb1              20G  172M   19G   1% /data
[root@localhost ~]#
[root@localhost ~]# tune2fs -m 0 /dev/sdb1
tune2fs 1.41.12 (17-May-2010)
Setting reserved blocks percentage to 0% (0 blocks)
[root@localhost ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-lv_root
                      8.5G  841M  7.2G  11% /
tmpfs                 499M     0  499M   0% /dev/shm
/dev/sda1             485M   54M  407M  12% /boot
/dev/sdb1              20G  172M   20G   1% /data
[root@localhost ~]#
```

Result

Thus procedure to attach virtual block to the virtual machine and to check whether it holds the data even after the release of the virtual machine was executed successfully.

EX:4. Find procedure to install storage controller and interact with it.

Aim:

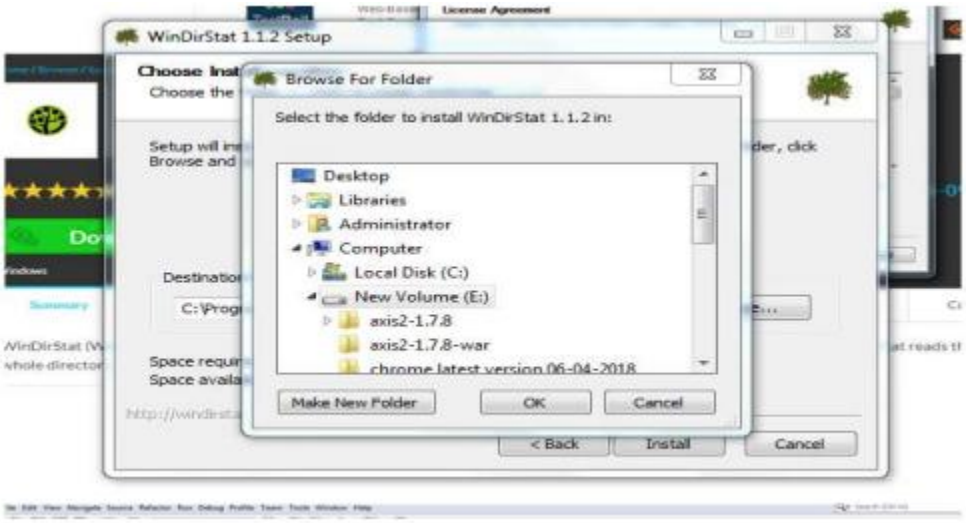
To install storage controller and interact with it. Algorithm:

Step1:Goto https://sourceforge.net/projects/windirstat/files/

Step2: download and Run the Winstart1_1_2

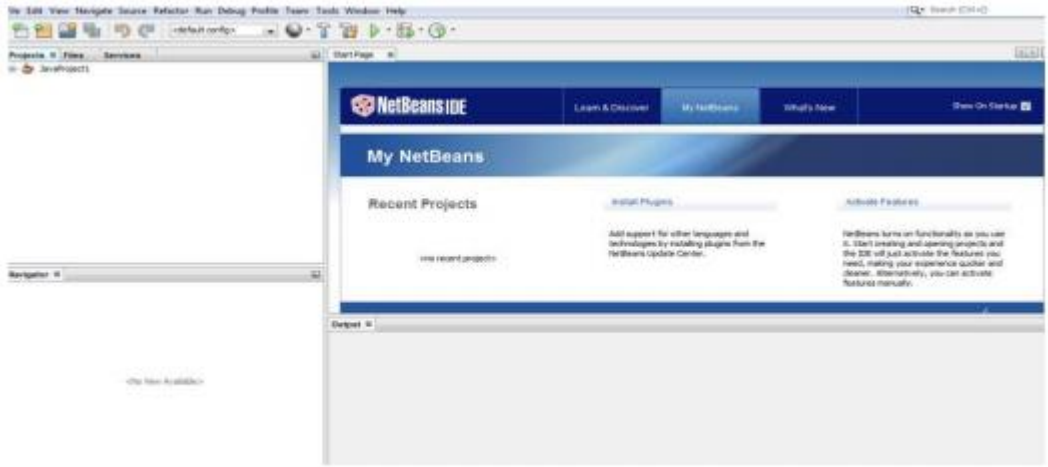
Step4: choose the E:drive and Do the Installation Step5:Open IDE-

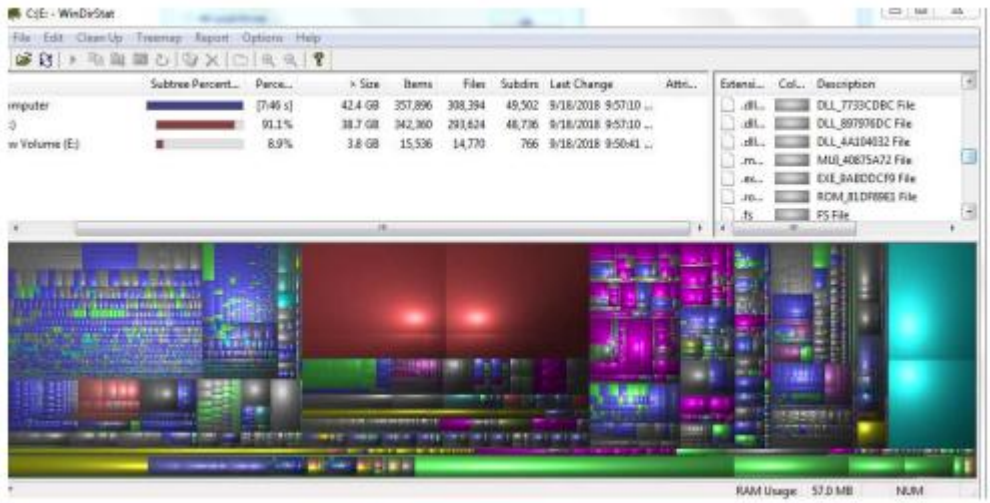
◊SELECT New Project

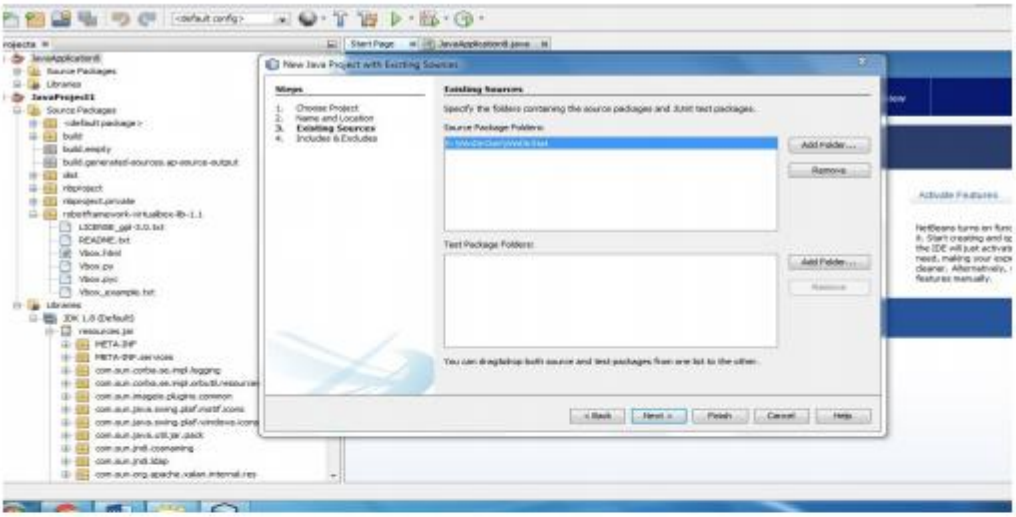


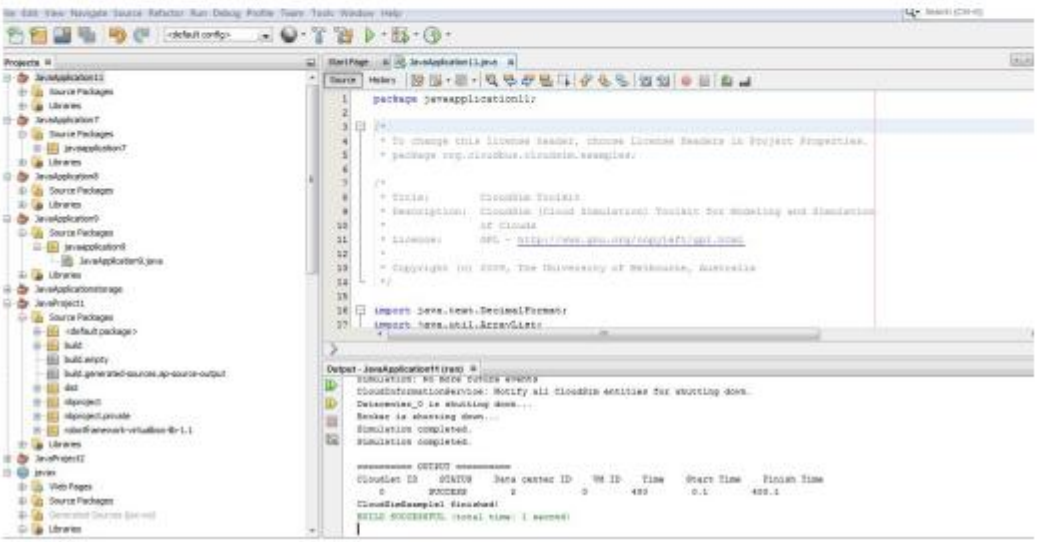
Step4: choose the E:drive and Do the Installation

Step5:Open IDE-◊SELECT New Project









Result:

Thus the storage controller has been implemented successfully.

VIVA QUESTIONS AND ANSWERS

1.Define Cloud Computing with example.

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

2.What is the working principle of Cloud Computing?

The cloud is a collection of computers and servers that are publicly accessible via the Internet. This hardware is typically owned and operated by a third party on a consolidated basis in one or more data center locations. The machines can run any combination of operating systems.

3.What are the advantages and disadvantages of Cloud Computing?

Advantages
Lower-Cost Computers for Users
Improved Performance
Lower IT Infrastructure Costs
Fewer Maintenance Issues
Lower Software Costs
Instant Software Updates
Increased Computing Power
Unlimited Storage Capacity
Increased Data Safety
Improved Compatibility Between Operating Systems
Improved Document Format Compatibility
Easier Group Collaboration
Universal Access to Documents
Latest Version Availability
Removes the Tether to Specific Devices
Disadvantages
Requires a Constant Internet Connection
Doesn't Work Well with Low-Speed Connections
Can Be Slow
Features Might Be Limited
Stored Data Might Not Be Secure
If the Cloud Loses Your Data, You're Screwed

4.What is distributed system?

A *distributed system* is a software system in which components located on networked computers communicate and coordinate their actions by

passing messages. The components interact with each other in order to achieve a common goal.
Three significant characteristics of distributed systems are:
Concurrency of components
Lack of a global clock
Independent failure of components

5.What is grid computing?
Grid Computing enables virtuals organizations to share geographically distributed resources as they pursue common goals, assuming the absence of central location, central control, omniscience, and an existing trust relationship.

6.What are the business areas needs in Grid computing?
Life Sciences
Financial services
Higher Education
Engineering Services
Government
Collaborative games

7.List out the Grid Applications:
Application partitioning that involves breaking the problem into discrete pieces
Discovery and scheduling of tasks and workflow
Data communications distributing the problem data where and when it is required
Provisioning and distributing application codes to specific system nodes
Autonomic features such as self-configuration, self-optimization, self-recovery and self
management

8.List some grid computing toolkits and frameworks?
Globus Toolkit Globus Resource Allocation Manager(GRAM)
Grid Security Infrastructure(GSI)
Information Services
Legion, Condor and Condor-G
NIMROD, UNICORE, NMI.

9.What are Desktop Grids?
These are grids that leverage the compute resources of desktop computers.

Because of the true (but unfortunate) ubiquity of Microsoft® Windows® operating system in corporations, desktop grids are assumed to apply to the Windows environment. The Mac OS™ environment is supported by a limited number of vendors.

10.What are Server Grids?

Some corporations, while adopting Grid Computing , keep it limited to server resources that are within the purview of the IT department. Special servers, in some cases, are bought solely for the purpose of creating an internal "utility grid" with resources made available to various departments. No desktops are included in server grids. These usually run some flavor of the Unix/Linux
operating system.

11.Define Opennebula.

OpenNebula is an open source management tool that helps virtualized data centers oversee private clouds, public clouds and hybrid clouds. ... OpenNebula is vendor neutral, as well as platform- and API-agnostic. It can use KVM, Xen or VMware hypervisors.

12.Define Eclipse.

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment.

13.Define Netbeans.

NetBeans is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages. NetBeans is also referred to as a platform of modular components used for developing Java desktop applications.

14.Define Apache Tomcat.

Apache Tomcat (or Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run."

15.What is private cloud?

The *private cloud* is built within the domain of an intranet owned by a single organization.

Therefore, they are client owned and managed. Their access is limited to the owning clients and their partners. Their deployment was not meant to sell capacity over the Internet through publicly accessible interfaces. Private clouds give local users a flexible and agile private infrastructure to run service workloads within their administrative domains.

16.What is public cloud?

A *public cloud* is built over the Internet, which can be accessed by any user who has paid for the service. Public clouds are owned by service providers. They are accessed by subscription. Many companies have built public clouds, namely Google App Engine, Amazon AWS, Microsoft Azure, IBM Blue Cloud, and Salesforce Force.com. These are commercial providers that offer a publicly accessible remote interface for creating and managing VM instances within their proprietary infrastructure.

17. What is hybrid cloud?

A *hybrid cloud* is built with both public and private clouds, Private clouds can also support a *hybrid cloud* model by supplementing local infrastructure with computing capacity from an external public cloud. For example, the *research compute cloud* (RC2) is a private cloud built by IBM.

18.What is a Community Cloud ?

A community cloud in computing is a collaborative effort in which infrastructure is shared between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.), whether managed internally or by a third-party and hosted internally or externally. This is controlled and used by a group of organizations that have shared interest. The costs are spread over fewer users than a public cloud (but more than a private cloud

19.Define IaaS?

The IaaS layer offers storage and infrastructure resources that is needed to deliver the Cloud services. It only comprises of the infrastructure or physical resource. Top IaaS Cloud Computing Companies: Amazon (EC2), Rackspace, GoGrid, Microsoft, Terremark and Google.

20.Define PaaS?

PaaS provides the combination of both, infrastructure and application. Hence, organisations using PaaS don’t have to worry for infrastructure nor for services. Top PaaS Cloud Computing
Companies: Salesforce.com, Google, Concur Technologies, Ariba, Unisys and Cisco..

21.Define SaaS?

In the SaaS layer, the Cloud service provider hosts the software upon their servers. It can be defined as a in model in which applications and softwares are hosted upon the server and made available to customers over a network. Top SaaS Cloud Computing Companies: Amazon Web Services, AppScale, CA Technologies, Engine Yard, Salesforce and Windows Azure.

22.What is meant by virtualization?

Virtualizationisacomputerarchitecturetechnologybywhichmultiplevirtualm achines (VMs)are multipl exedin the same hardwar emachine.Theideaof VMs canbe dated back to the 1960s.The purpose of a VM is to enhance resource sharing by many users and improve computer performance interms of resource utilization and application flexibility.

23.What are the implementation levels of virtualization?

The virtualization types are following

1. OS-level virtualization
2. ISA level virtualization
3. User-ApplicationLevel virtualization
4. hardware level virtualization
5. library level virtualization

24.List the requirements of VMM?

There are three requirements for a VMM. First, a VMM should provide an environment for programs which is essentially identical to the original machine. Second, programs run in this environment should show, at worst, only minor decreases in speed. Third, a VMM should be in complete control of the system resources.

25.Explain Host OS and Guest OS?

A comparison of the differences between a host system, a guest system, and a virtual machine within a virtual infrastructure. A host system (host operating system) would be the primary & first installed operating system. If you are using a bare metal Virtualization platform like Hyper-V or ESX, there really isn't a host operating system besides the Hypervisor. If you are using a Type-2 Hypervisor like VMware Server or Virtual Server, the host operating system is whatever operating system those applications are installed into. A guest system (guest operating system) is a virtual guest or virtual machine (VM) that is installed under the host operating system. The guests are the VMs that you run in your virtualization platform.

26.Write the steps for live VM migration?

The five steps for live VM migration is

Stage 0: *Pre-Migration*

Active VM on Host A

Alternate physical host may be preselected for migration

Block devices mirrored and free resources maintained

Stage 1: *Reservation*

Initialize a container on the target host

Stage 2: *Iterative pre-copy*

Enable shadow paging

Copy dirty pages in successive rounds.

Stage 3: *Stop and copy*

Suspend VM on host A

Generate ARP to redirect traffic to Host B

Synchronize all remaining VM state to Host B

Stage 4: *Commitment*

VM state on Host A is released

Stage 5: *Activation*

VM starts on Host B

Connects to local devicesResumes normal operation

27.Define Globus Toolkit: Grid Computing Middleware

Globus is open source grid software that addresses the most challenging problmes in distributed resources sharing. The Globus Toolkit includes software services and libraries for distributed security, resource management, monitoring and discovery, and data management.

28.Define Blocks in HDFS

A disk has a block size, which is the minimum amount of data that it can read or write. Filesystems for a single disk build on this by dealing with data in blocks, which are an integral multiple of the disk block size. Filesystem blocks are typically a few kilobytes in size, while disk blocks are normally 512 bytes. This is generally transparent to the filesystem user who is simply reading or writing a file—of whatever length.