Terna Engineering College

Computer Engineering Department
Program: Sem VI

Course: Cloud Computing Lab (CSL603)

Faculty: Preeti Patil

Experiment No.3

A.1 Aim:
To study and Implement Infrastructure as a Service using AWS.

A.2 Prerequisite:
Understanding of Virtualization,Basics of Networking ,Basics of security and privacy.

A.3 Objective:
To demonstrate the steps to create and run virtual machines inside Public cloud platform.

A.3 Outcome: (LO2)
After successful completion of this experiment students will be able to demonstrate infrastructure as a service by launching EC2 instance.

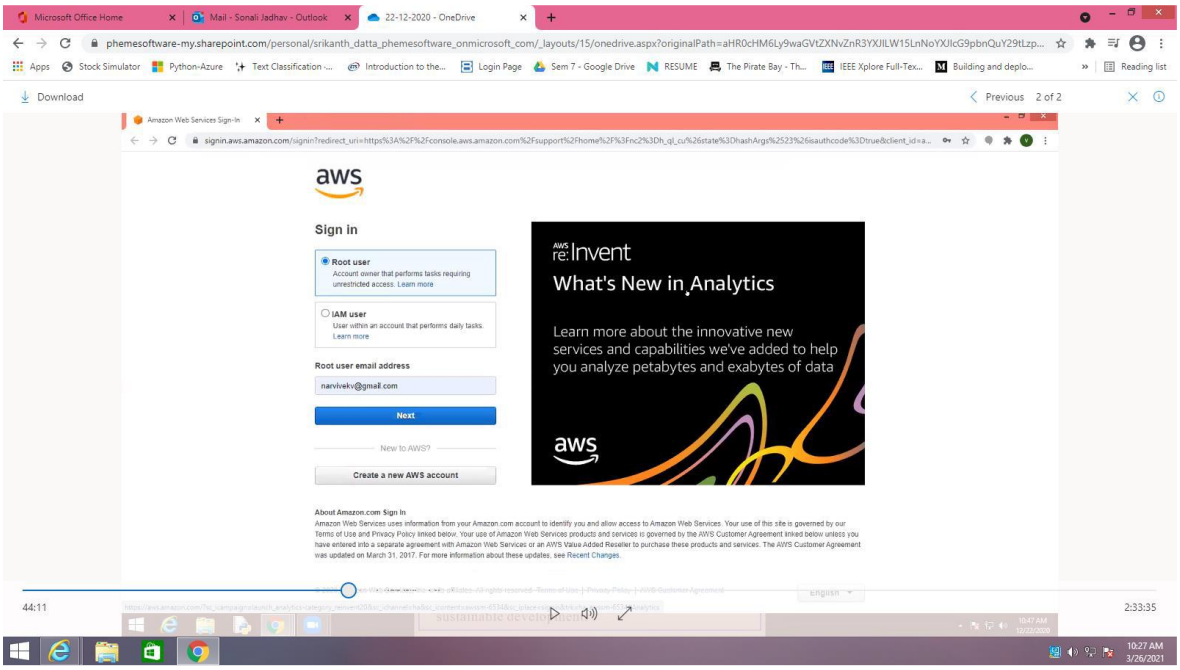
A.4 Theory:

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers and system administrators. Amazon EC2's simple web service interface allows obtaining and configuring capacity with minimal friction. It provides with complete control of computing resources and run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that actually use. Amazon EC2 provides developers and system administrators the tools to build failure resilient applications and isolate themselves from common failure scenarios. Auto Scaling allows scaling Amazon EC2 capacity up or down automatically according to conditions you define. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs. Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve even greater fault tolerance in your applications,

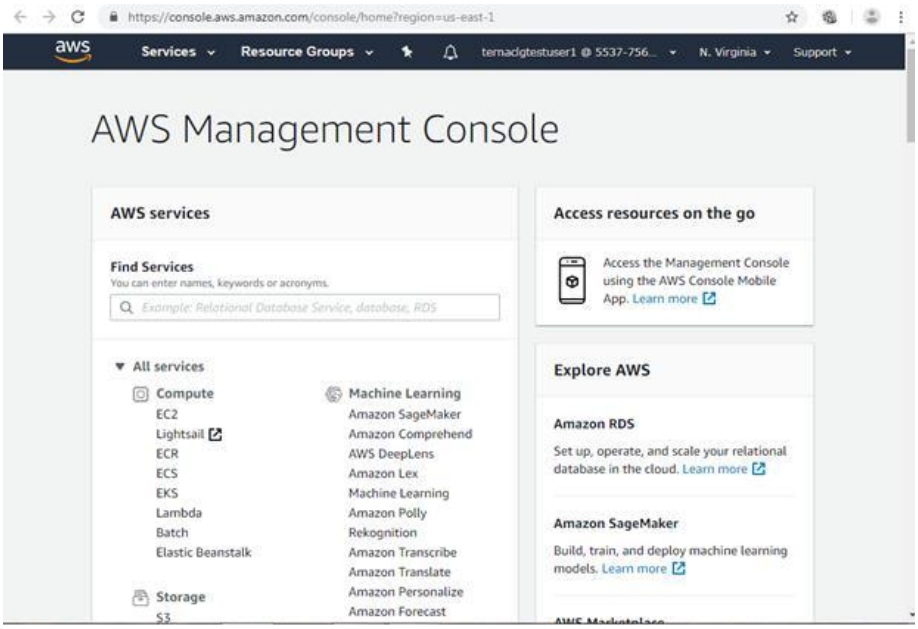
seamlessly providing the amount of load balancing capacity needed in response to incoming application traffic.

Following are the steps to launch EC2 instances.

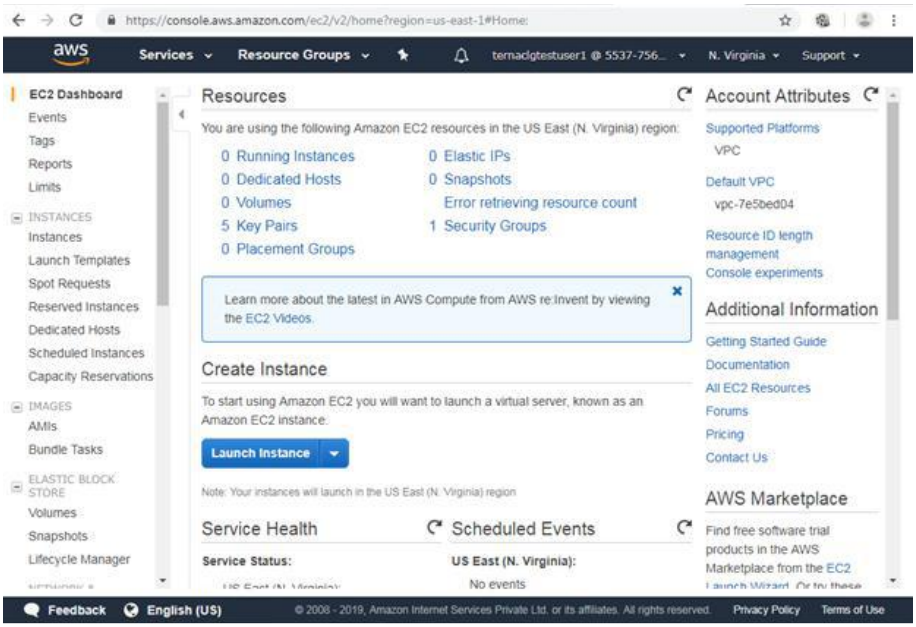
AWS account creation steps,



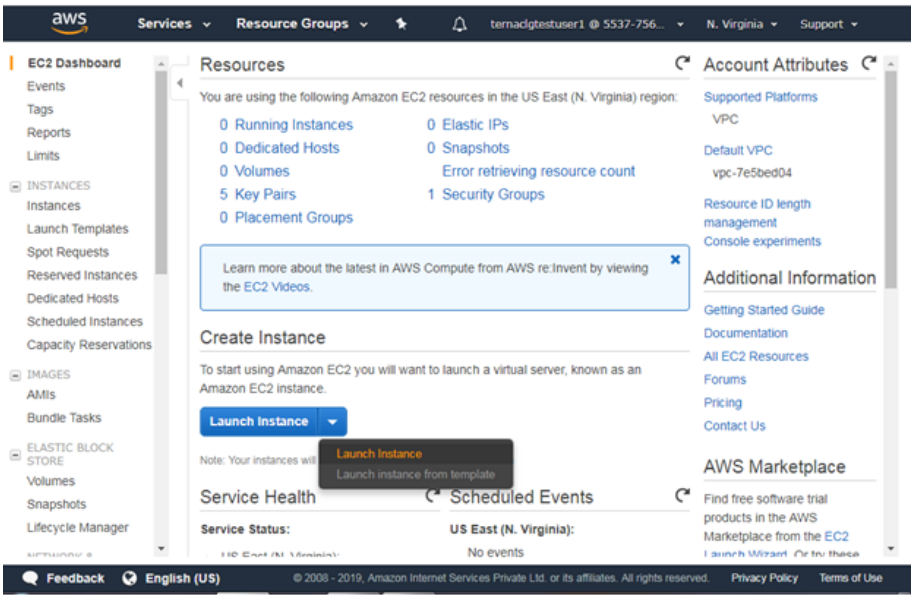
Step 1:Login



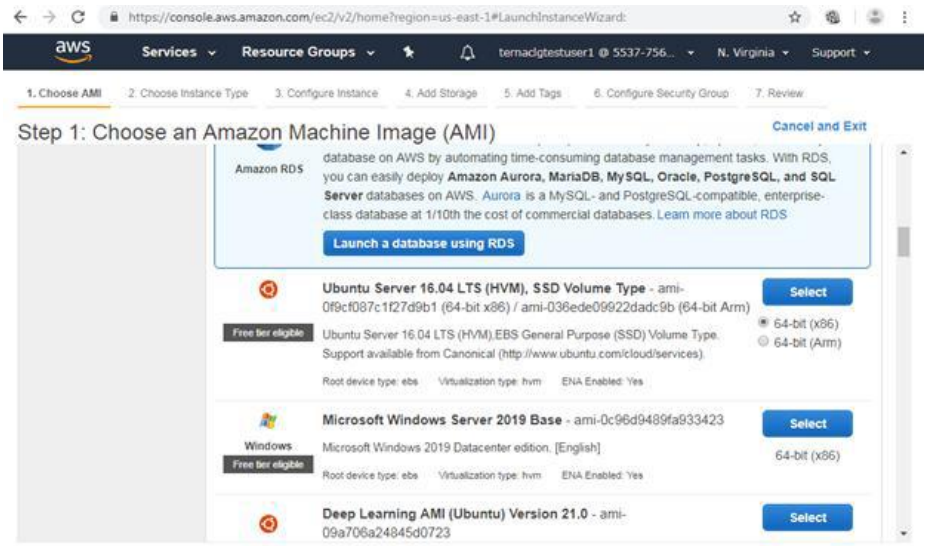
Step 2: Open AWS services



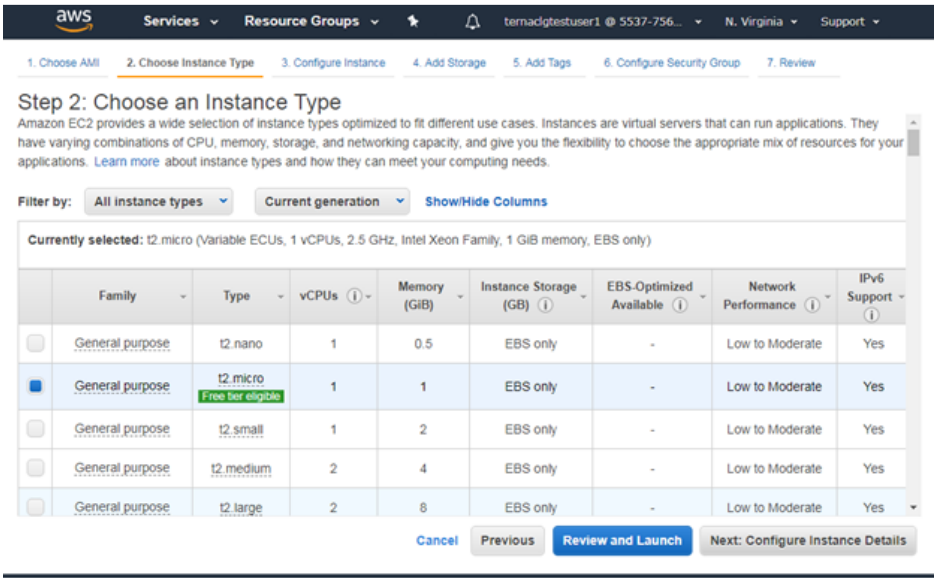
Step 3: Open EC2



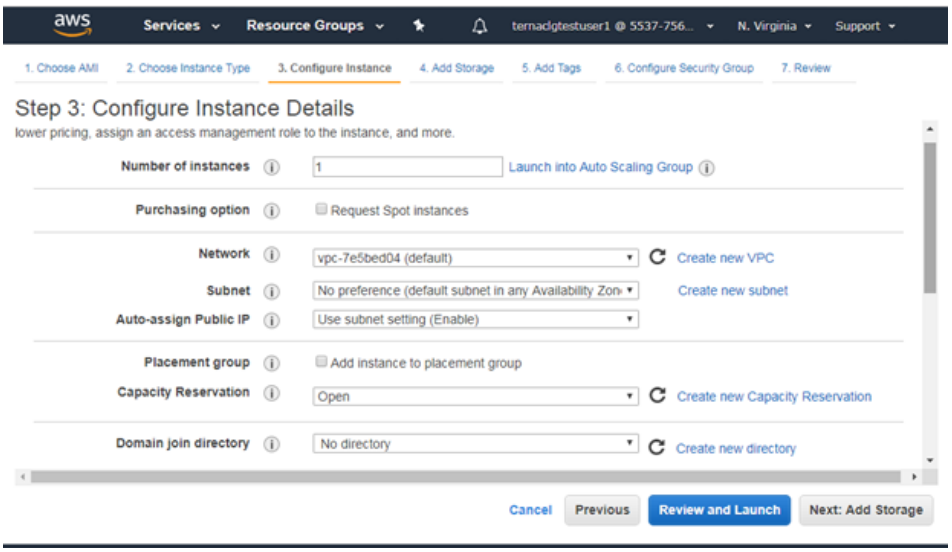
Step 4:Select Launch Instance



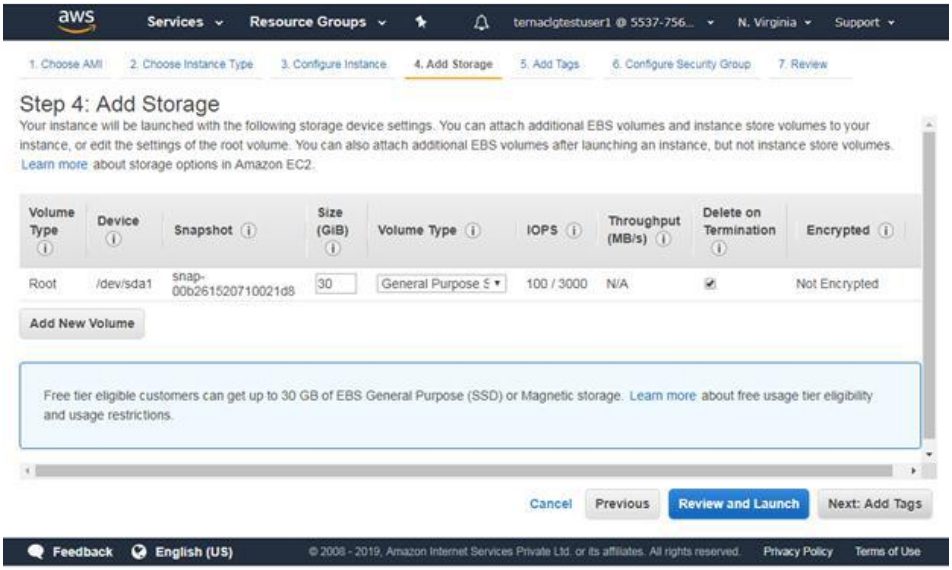
Step 5: Select Amazon Linux



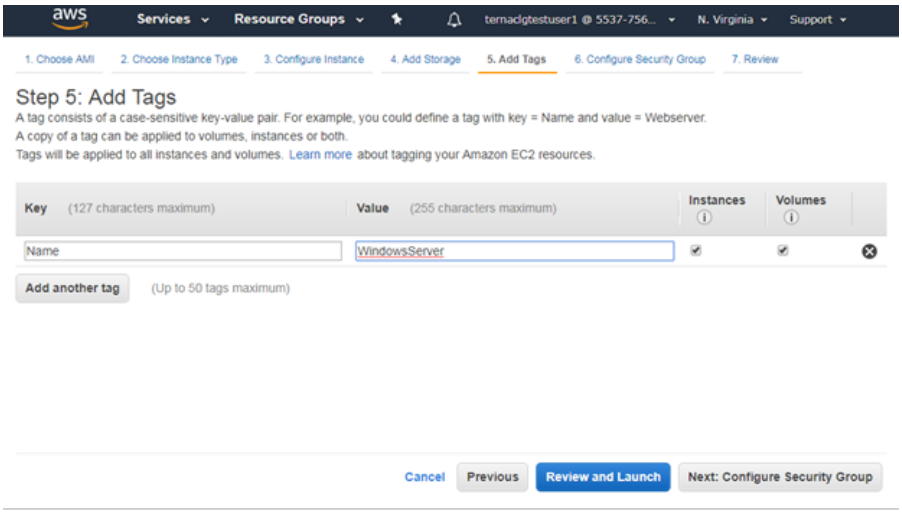
Step 6: Select Instance Type



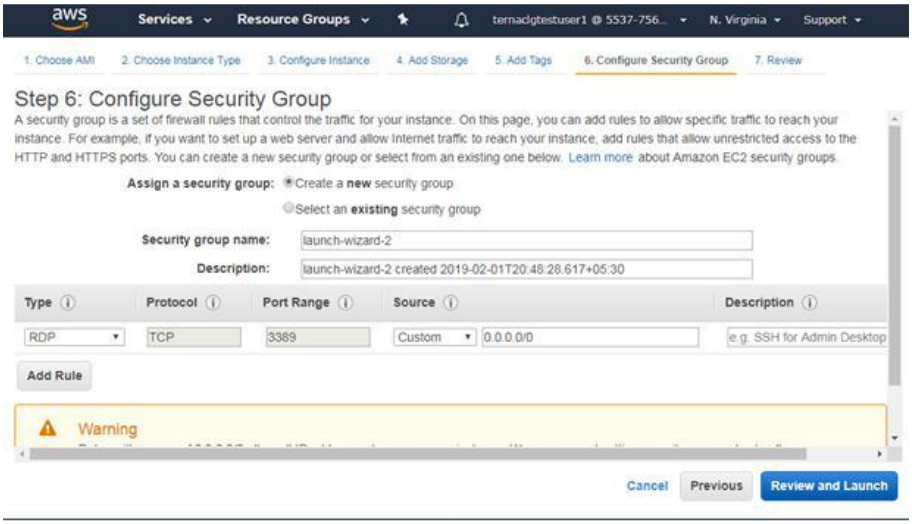
Step 7: Select Next: Add storage



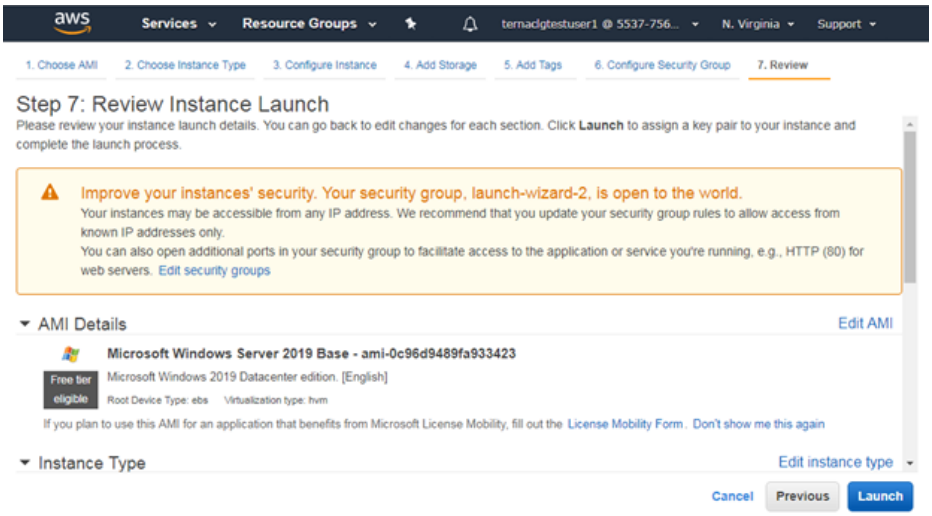
Step 8:Select Next: Add Tags



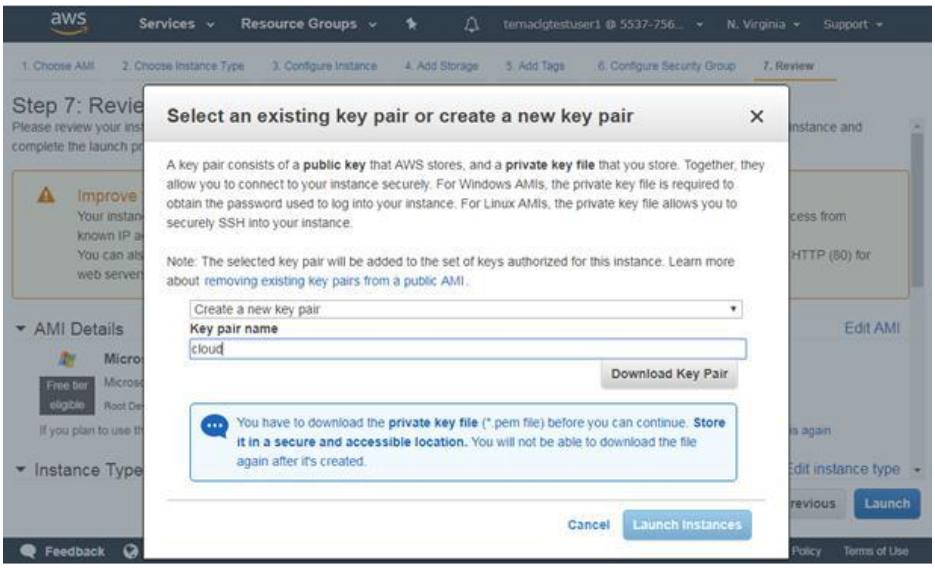
Step 9: Select Next: Configure Security Group



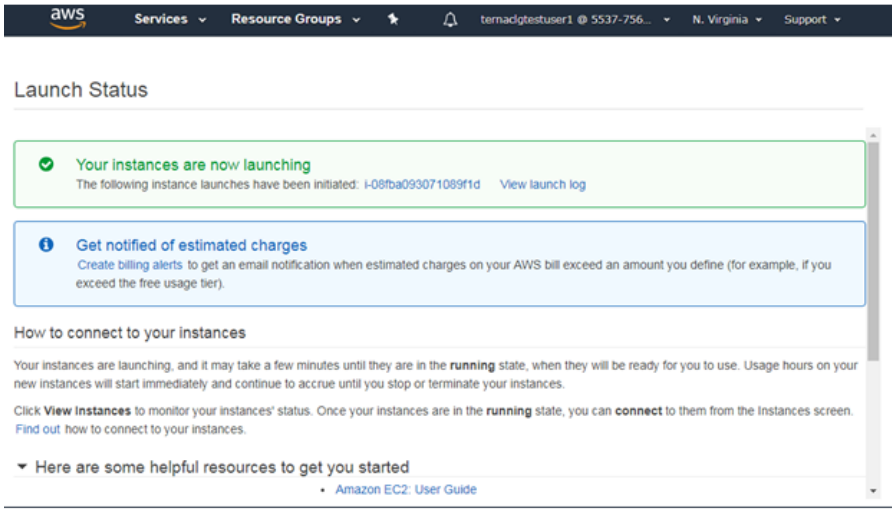
Step 10: Select Review and launch

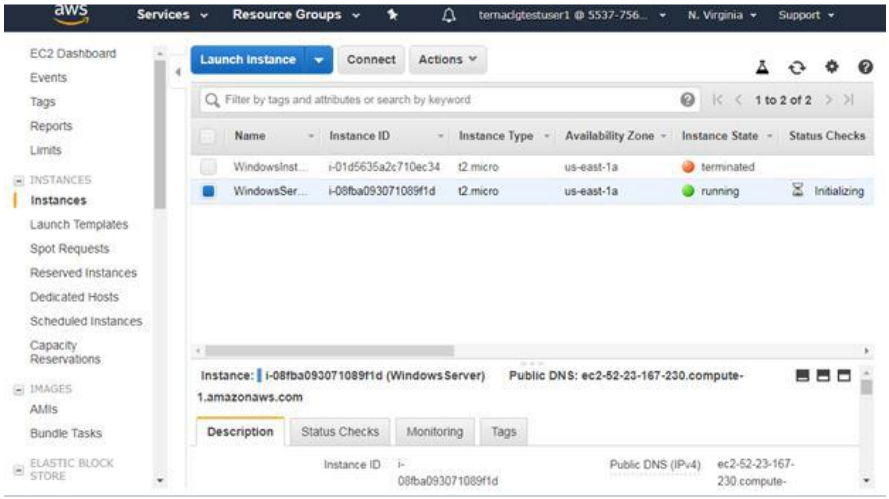


Step 11:Select launch

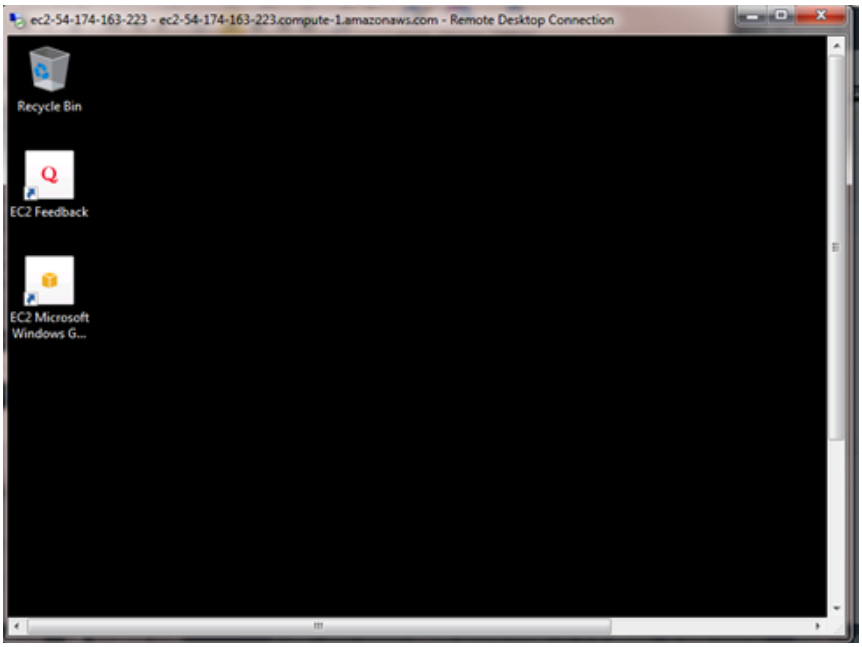


Step 12: Select an existing key pair and then click Launch

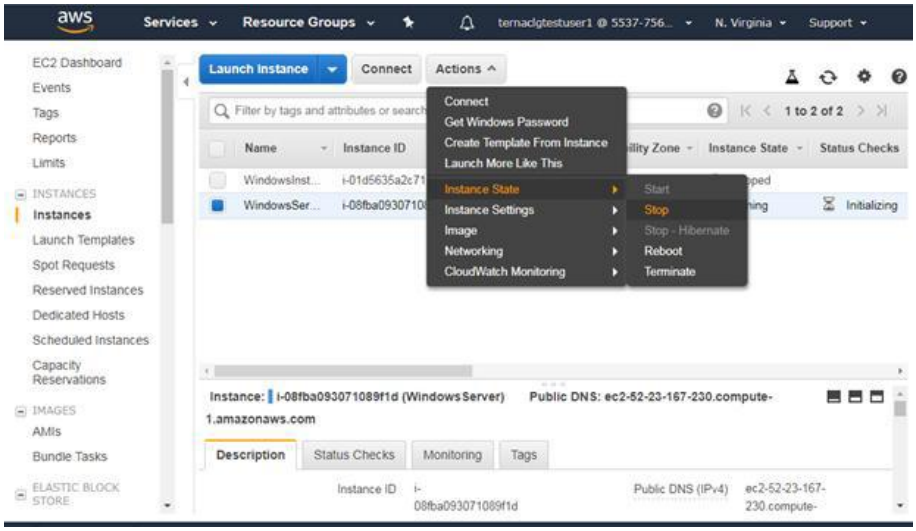




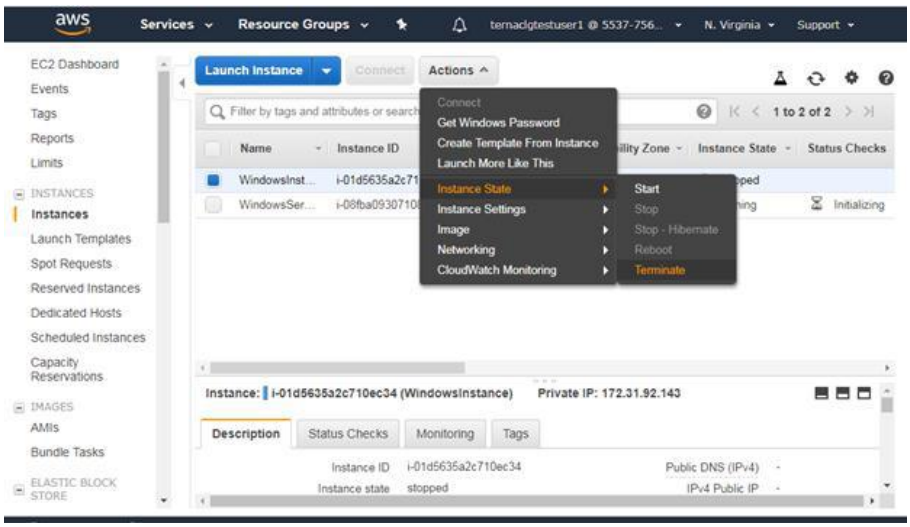
Step 13: Instance State then click Connect Instance



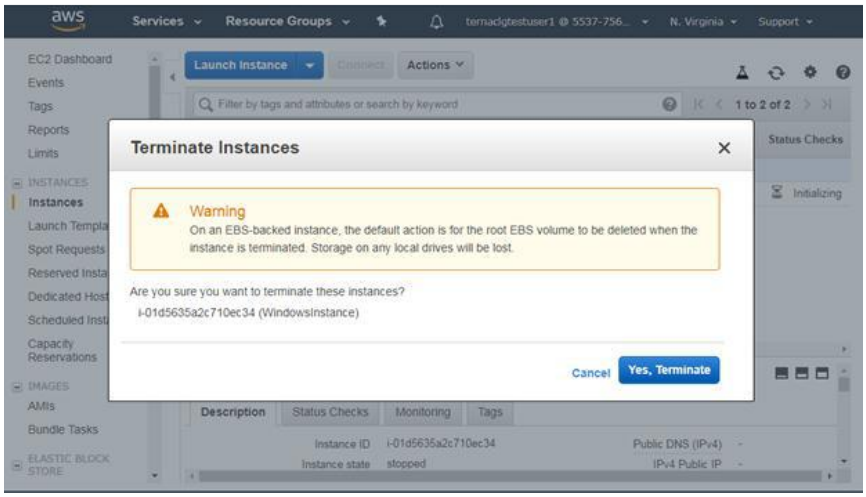
Step 14: After connecting instance



Step 15:How to stop instance



Step 16: How to terminate an instance



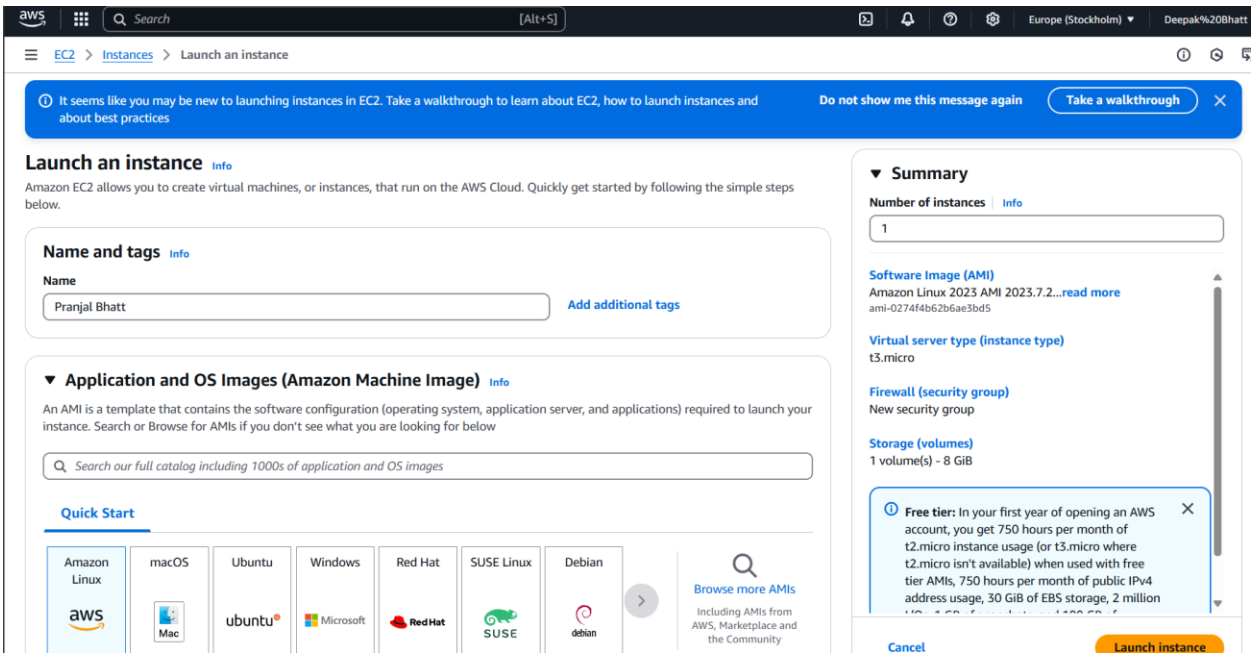
PART B

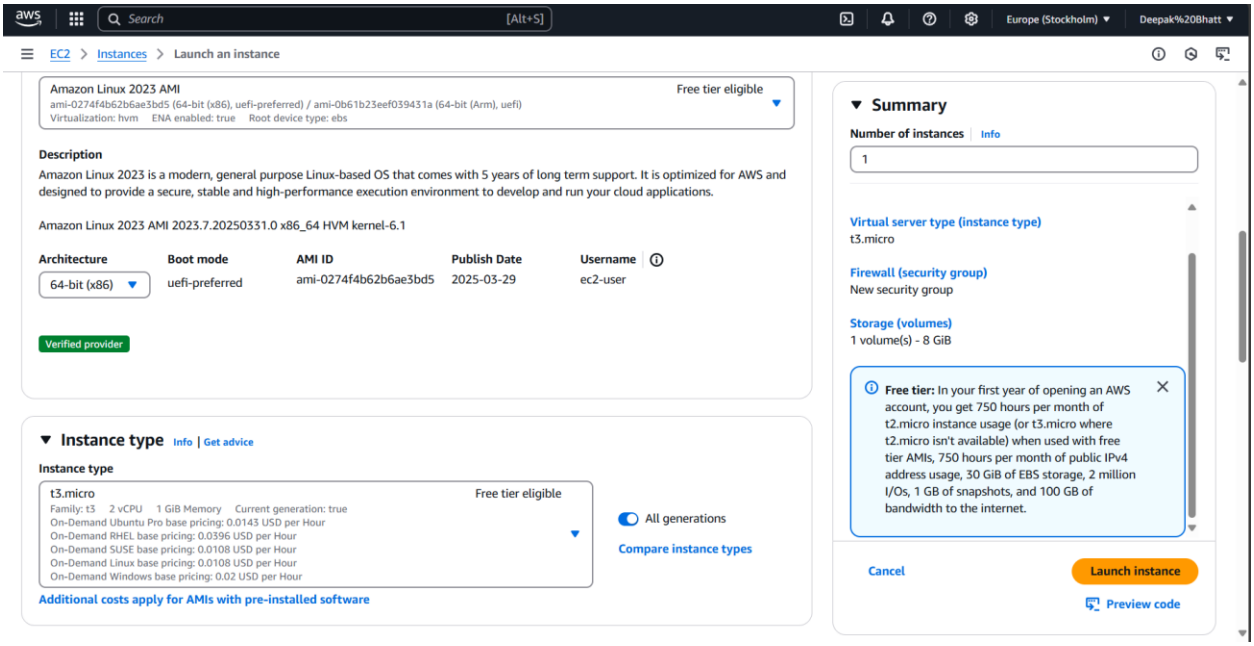
(PART B: TO BE COMPLETED BY STUDENTS)

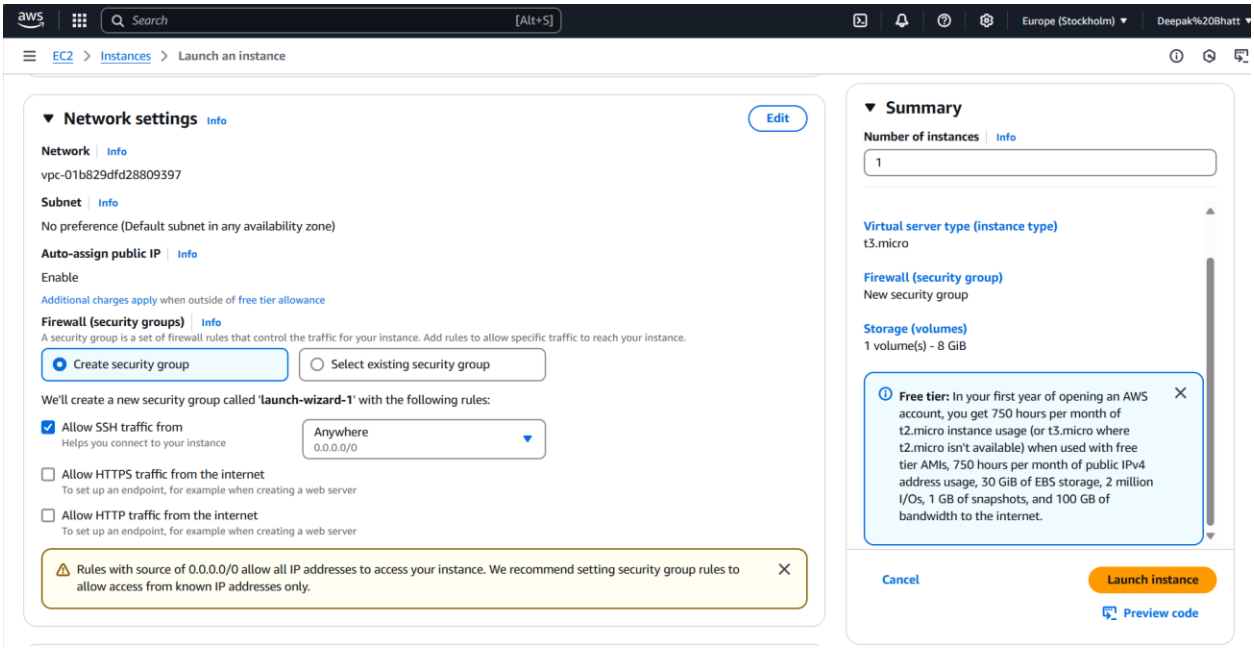
(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the ERP or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no ERP access available)

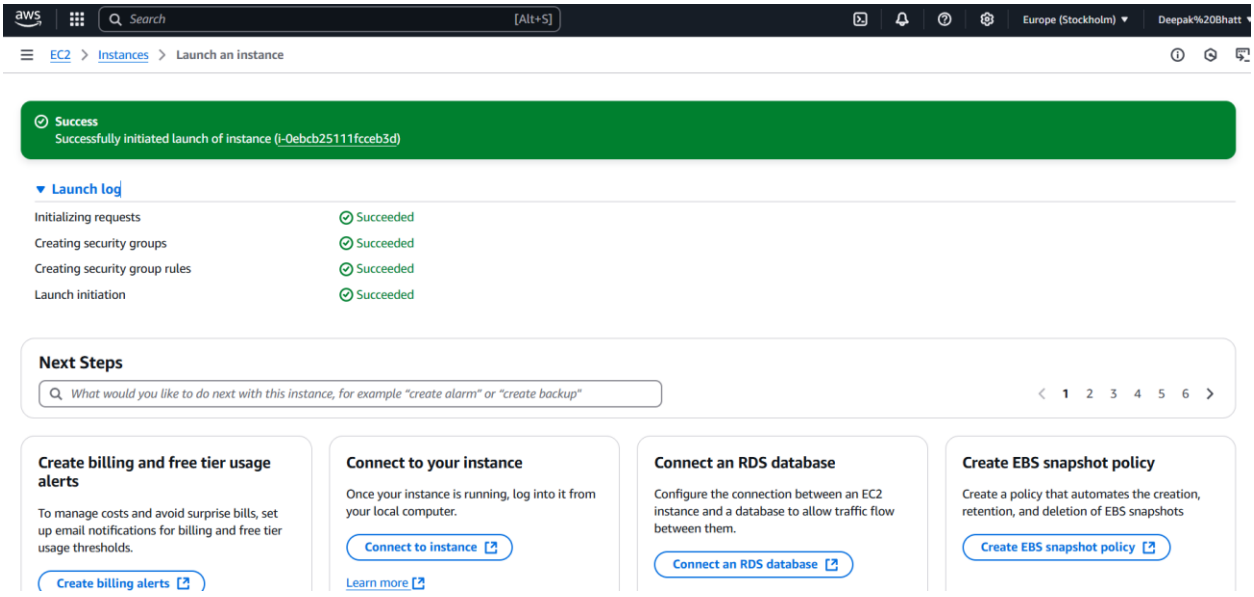
Roll No.B30	Name: Bhatt Pranjal Deepak
Class :TE Comps B	Batch :B2
Date of Experiment:	Date of Submission
Grade :	

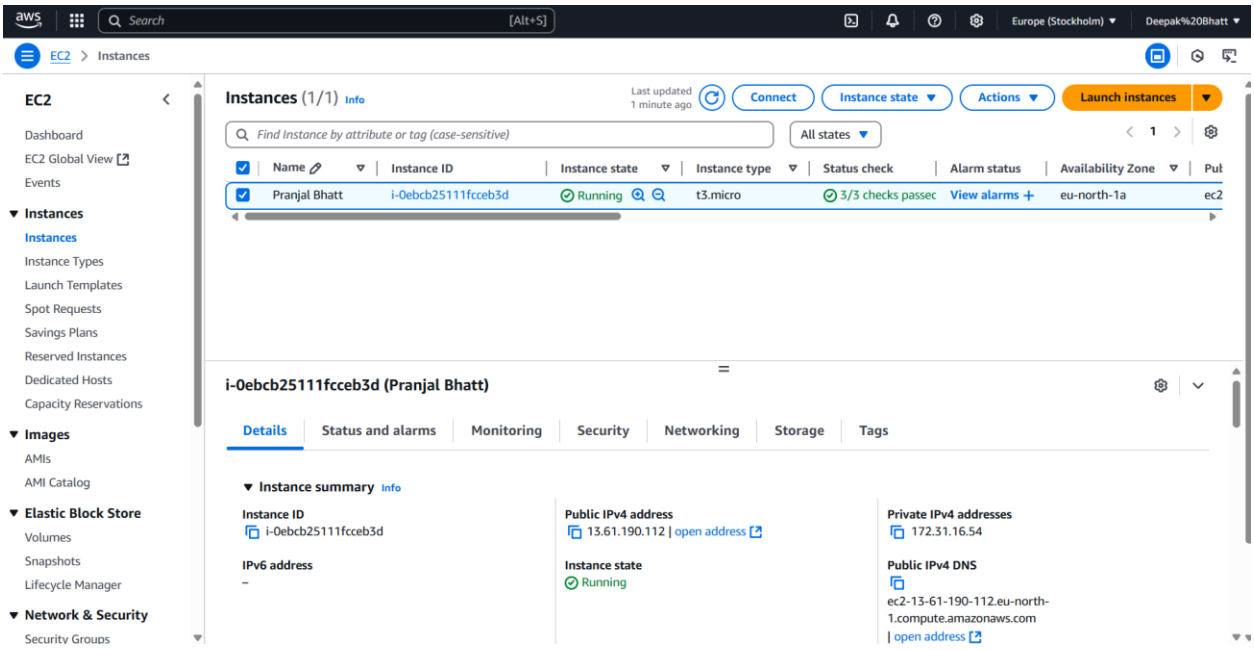
B.1Question of Curiosity:

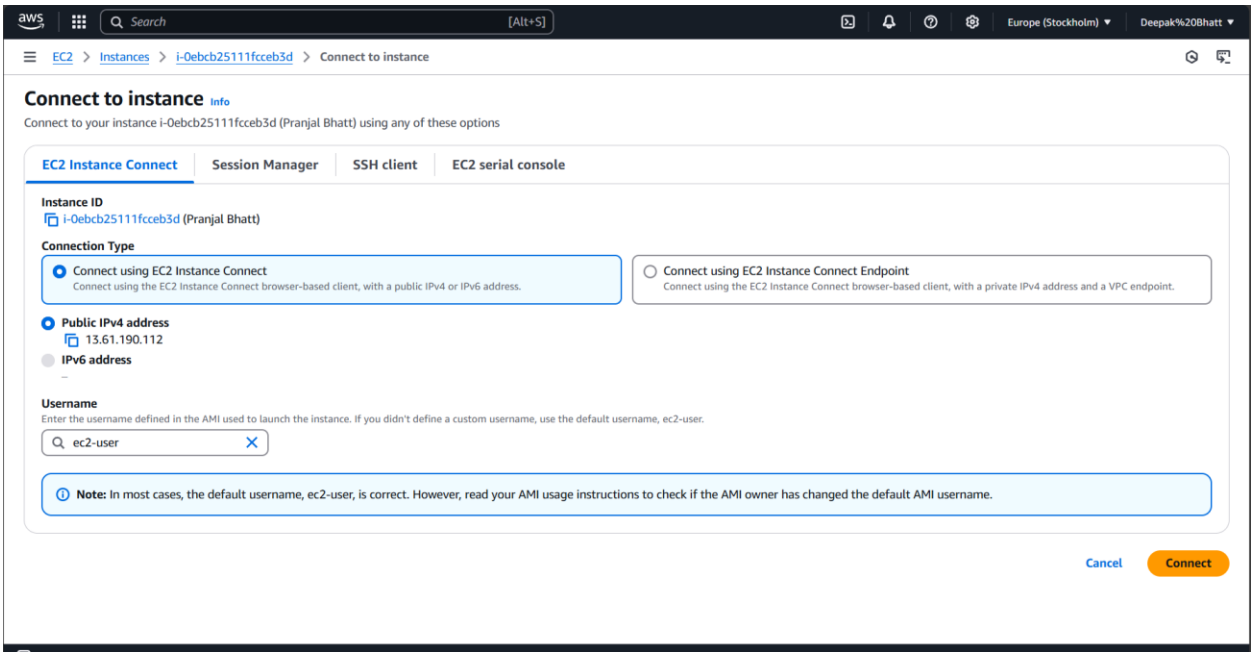


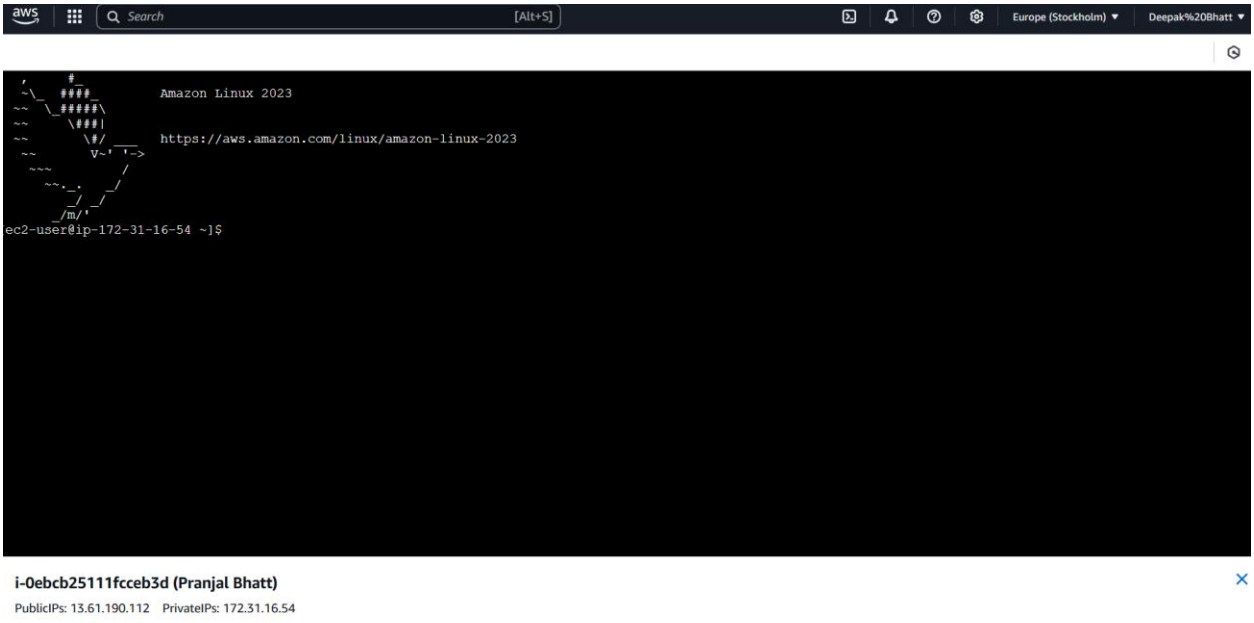

Search [Alt+S]
Europe (Stockholm)
Deepak%20Bhatt
EC2 > Instances > Launch an instance
Amazon Linux 2023 AMI
Free tier eligible
ami-0274f4b62b6ae3bd5 (64-bit (x86), uefi-preferred) / ami-0b61b23eef039431a (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Description
Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.
Amazon Linux 2023 AMI 2023.7.20250331.0 x86_64 HVM kernel-6.1
Architecture
64-bit (x86)
Boot mode
uefi-preferred
AMI ID
ami-0274f4b62b6ae3bd5
Publish Date
2025-03-29
Username
ec2-user
Verified provider
Instance type Info | Get advice
Instance type
t3.micro
Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0143 USD per Hour
On-Demand RHEL base pricing: 0.0396 USD per Hour
On-Demand SUSE base pricing: 0.0108 USD per Hour
On-Demand Linux base pricing: 0.0108 USD per Hour
On-Demand Windows base pricing: 0.02 USD per Hour
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Summary
Number of instances Info
1
Virtual server type (instance type)
t3.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Preview code


Search [Alt+S]
Europe (Stockholm)
Deepak%20Bhatt
EC2 > Instances > Launch an instance
Network settings Info
Edit
Network Info
vpc-01b829dfd28809397
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
We'll create a new security group called 'launch-wizard-1' with the following rules:
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server
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 Info
1
Virtual server type (instance type)
t3.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Preview code


Search [Alt+S]
Europe (Stockholm)
Deepak%20Bhatt
EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-0ebcb25111fcceb3d)
Launch log
Initializing requests Succeeded
Creating security groups Succeeded
Creating security group rules Succeeded
Launch initiation Succeeded
Next Steps
What would you like to do next with this instance, for example "create alarm" or "create backup"
1 2 3 4 5 6
Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts
Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more
Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy

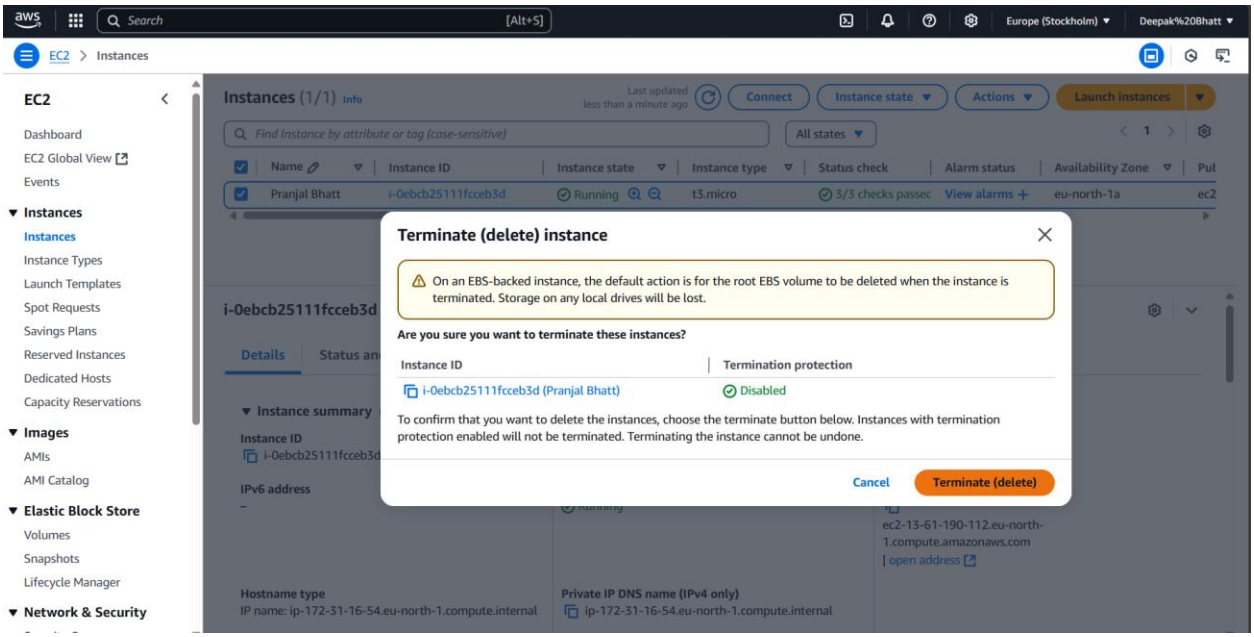

Search [Alt+S]
Europe (Stockholm)
Deepak%20Bhatt
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Instances (1/1) Info
Last updated 1 minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Pranjal Bhatt
i-0ebcb25111fcceb3d
Running
t3.micro
3/3 checks passed
View alarms
eu-north-1a
i-0ebcb25111fcceb3d (Pranjal Bhatt)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0ebcb25111fcceb3d
IPv6 address
–
Public IPv4 address
13.61.190.112 | open address
Instance state
Running
Private IPv4 addresses
172.31.16.54
Public IPv4 DNS
ec2-13-61-190-112.eu-north-1.compute.amazonaws.com
| open address

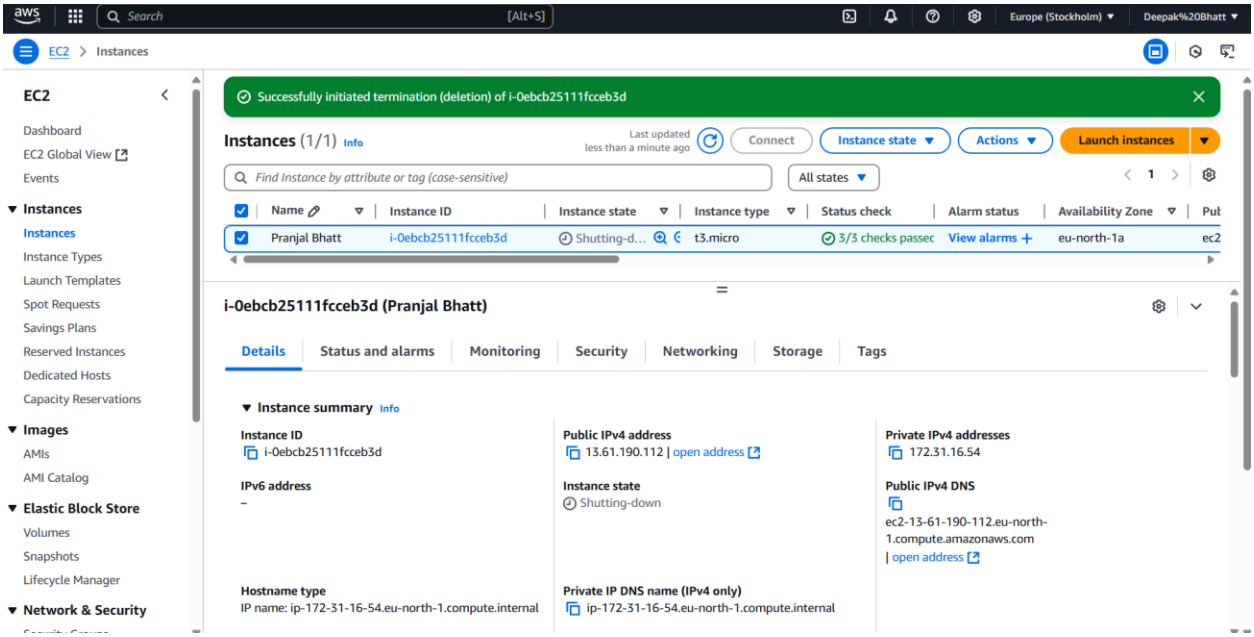


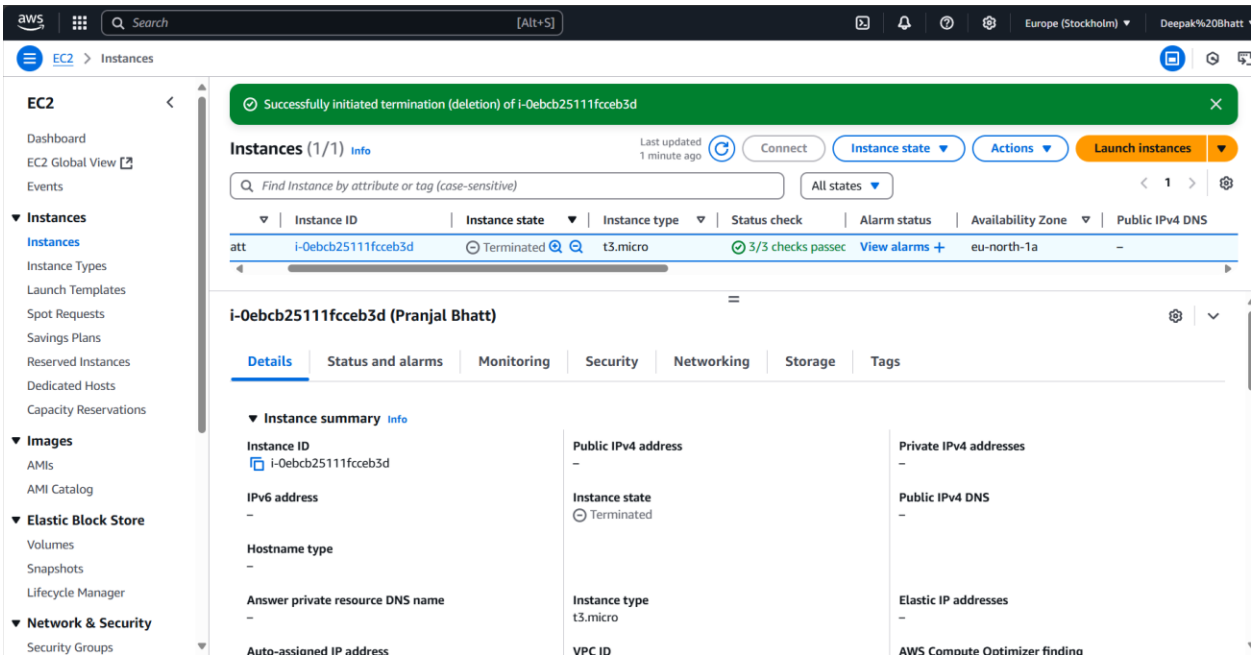


i-0ebcb25111fcceb3d (Pranjal Bhatt)

PublicIPs: 13.61.190.112 PrivateIPs: 172.31.16.54


Terminate (delete) instance
On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.
Are you sure you want to terminate these instances?
Instance ID
Termination protection
i-0ebcb25111fcceb3d (Pranjal Bhatt)
Disabled
To confirm that you want to delete the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.
Cancel
Terminate (delete)


Successfully initiated termination (deletion) of i-0ebcb25111fcceb3d
i-0ebcb25111fcceb3d (Pranjal Bhatt)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary



(To be answered by student based on the practical performed and learning/observations)

Q1: What is Hypervisor? List out various hypervisors which are used to create Virtual machines

Ans:

A **Hypervisor** is a layer of software or firmware that creates and manages virtual machines (VMs). It allows multiple operating systems (OS) to run on a single physical machine by abstracting the underlying hardware. The hypervisor manages the allocation of resources like CPU, memory, and storage to each VM, ensuring that they run independently of each other.

There are two main types of hypervisors:

1. **Type 1 Hypervisor (Bare-metal Hypervisor)**: Runs directly on the host hardware without the need for an underlying operating system. It's generally more efficient and secure because it has direct access to the hardware.

   Examples:

   - VMware vSphere/ESXi
   - Microsoft Hyper-V (when installed on bare metal)
   - Xen
   - KVM (Kernel-based Virtual Machine)

    - Oracle VM
2. **Type 2 Hypervisor (Hosted Hypervisor)**: Runs on top of an existing operating system, utilizing the host OS to manage hardware resources. It's usually easier to set up but less efficient than Type 1.

   Examples:

    - VMware Workstation
    - Oracle VirtualBox
    - Parallels Desktop
    - QEMU (can also function as a Type 1 hypervisor depending on the setup)

Each type has its use cases, with Type 1 being more suitable for enterprise or data center environments and Type 2 for personal or development use.

Q2: Compare and Contrast Hardware and Software Virtualization.

Ans:

Hardware Virtualization and **Software Virtualization** are two approaches for creating virtual machines (VMs), and they differ primarily in the way they interact with the underlying physical hardware. Here's a comparison and contrast of the two:

1. Definition:

- **Hardware Virtualization**: Involves using specialized hardware features, such as Intel VT-x or AMD-V, to create virtual machines. The hypervisor communicates directly with the physical hardware, leveraging the processor's virtualization extensions to create and manage VMs.
- **Software Virtualization**: Uses software techniques to emulate hardware or to simulate a virtualized environment. It doesn't require special hardware extensions and relies on the host OS and the hypervisor to simulate the hardware resources for the guest OS.

2. Performance:

- **Hardware Virtualization**: Offers better performance because the hypervisor runs directly on the hardware. With hardware-assisted features, VMs can achieve near-native speeds, reducing overhead.
- **Software Virtualization**: Typically introduces more overhead because the software is simulating the hardware environment. This can result in slower performance, as the guest OS must rely on the host OS for hardware access.

3. Complexity:

- **Hardware Virtualization**: Can be more complex to set up because it may require specific hardware support (like CPU virtualization extensions). However, once set up, it tends to be more stable and efficient.
- **Software Virtualization**: Generally easier to set up because it doesn't require specialized hardware. However, because it relies on software, it may be less efficient and more complex in terms of resource management.

4. Resource Management:

- **Hardware Virtualization**: Allows direct allocation of resources (CPU, memory, etc.) to VMs with less overhead. VMs have a more direct interaction with the hardware, which results in more efficient resource utilization.
- **Software Virtualization**: Relies on the host OS for resource allocation, which adds an extra layer of overhead and may limit the effective distribution of resources, especially in resource-intensive scenarios.

5. Security:

- **Hardware Virtualization**: Generally more secure because of its isolation between VMs. The hypervisor can take advantage of hardware-level isolation, which reduces the risk of interference between VMs.
- **Software Virtualization**: Can be less secure because the guest OS is running inside a virtualized environment on top of the host OS. If vulnerabilities exist in the software, they can potentially affect both the host and the VMs.

6. Use Cases:

- **Hardware Virtualization**: Ideal for enterprise environments, data centers, and situations where performance, security, and scalability are critical. It’s commonly used for running multiple VMs on a physical server.
- **Software Virtualization**: More commonly used for personal or development environments, where ease of use and lower resource consumption may be prioritized over maximum performance. It’s often seen in desktop virtualization or development environments.

7. Examples:

- **Hardware Virtualization**: VMware ESXi, Microsoft Hyper-V (bare-metal), Xen, KVM.
- **Software Virtualization**: VMware Workstation, Oracle VirtualBox, Parallels Desktop.

Summary Table:

Feature	Hardware Virtualization	Software Virtualization
Definition	Direct access to hardware resources	Simulates hardware through software

Feature	Hardware Virtualization	Software Virtualization
Performance	High, near-native speed	Lower, more overhead
Complexity	More complex to set up	Easier to set up
Resource Management	Efficient resource allocation	Higher overhead, less efficient
Security	Better isolation and security	More prone to security risks
Use Cases	Data centers, enterprise, production	Personal use, development, testing
Examples	VMware ESXi, Hyper-V, KVM, Xen	VMware Workstation, VirtualBox, Parallels

In conclusion, **hardware virtualization** is better suited for high-performance, secure, and large-scale environments, while **software virtualization** is often used for less demanding tasks or in scenarios where simplicity and cost are more important.

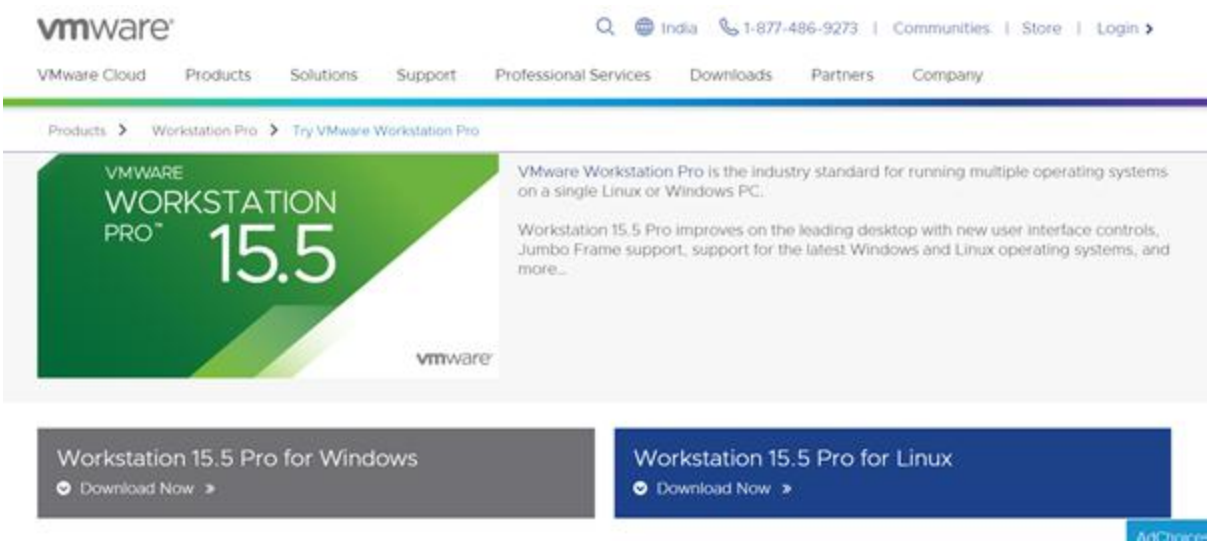
Q3: Write down steps to install VM ware workstation (with appropriate screen shot).

Ans:

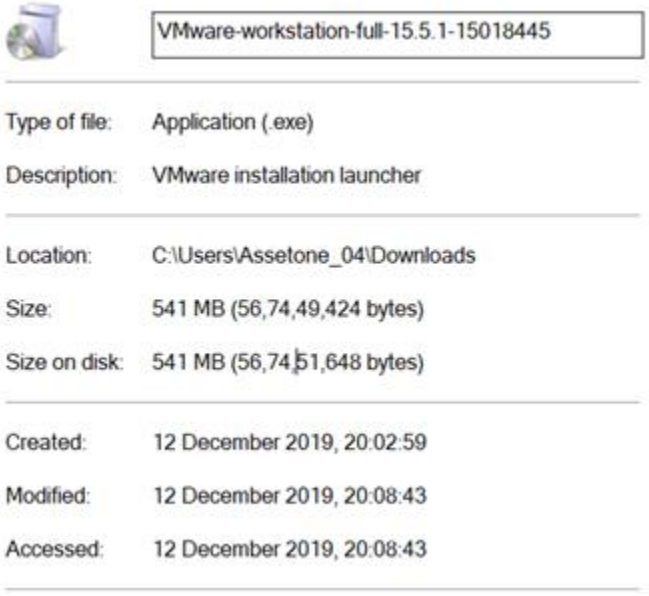
Let us start Setup and Installation :

1. Installing VMware Workstation from given below link. There are two options for downloading one is Windows and other for Linux. My Base Operating System is Windows8, So I choose for VMware for Windows. If Your Base OS is Linux go and choose VMware for Linux Link.

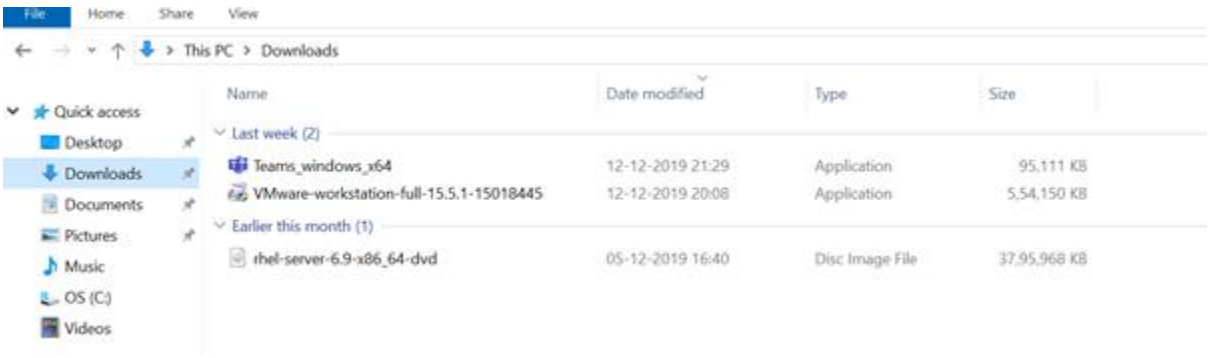
https://www.vmware.com/in/products/workstation-pro/workstation-pro-evaluation.html



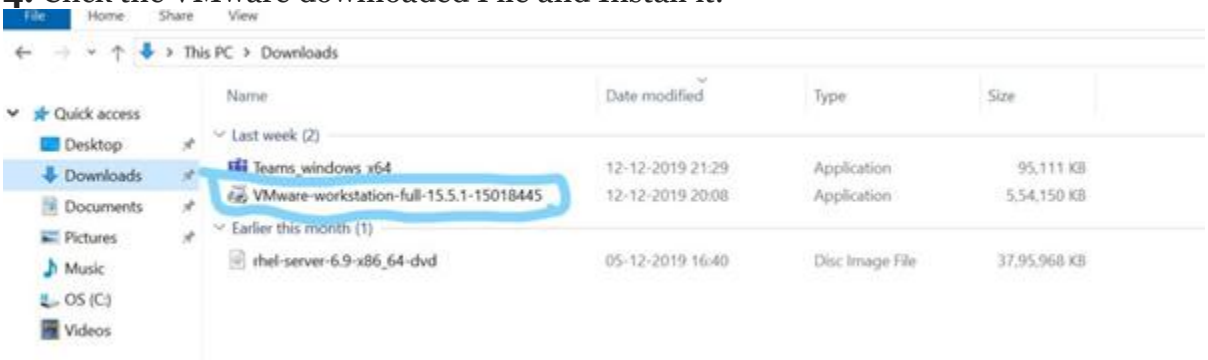
2.Check your VMware Properties.



3. Go to Download Folder.



4. Click the VMware downloaded File and Install it.

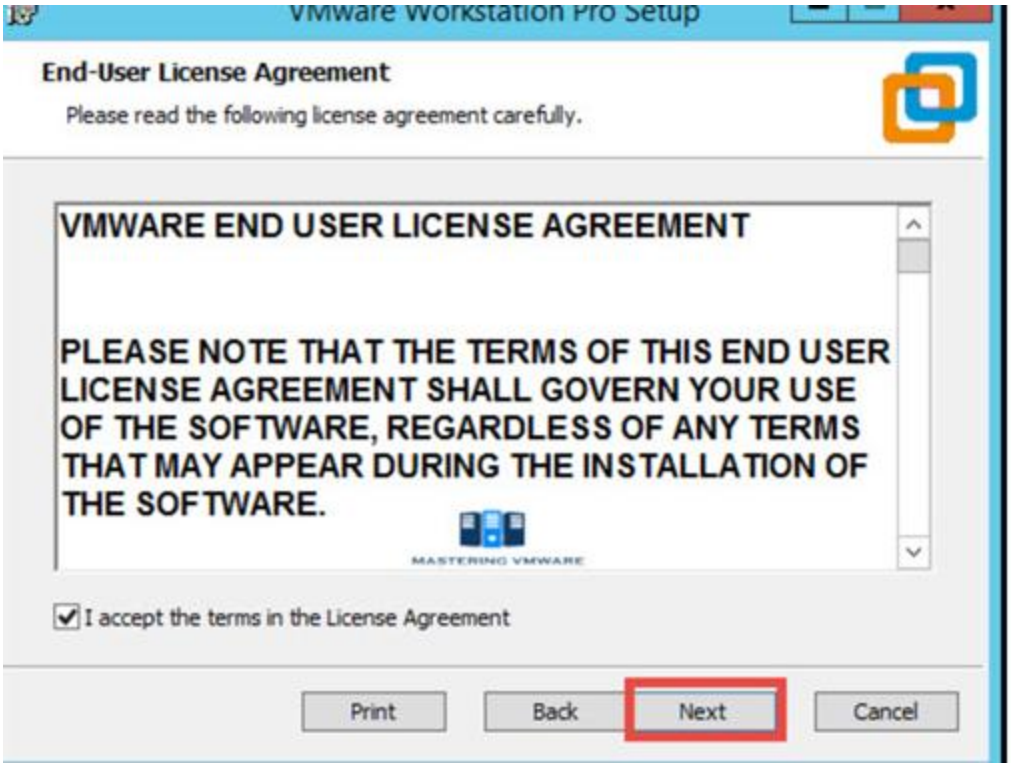


5.Click on VMware Software and click and choose "**Pin to Taskbar".**

6.Click on VMware Software and Click on Next to the Installation wizard.

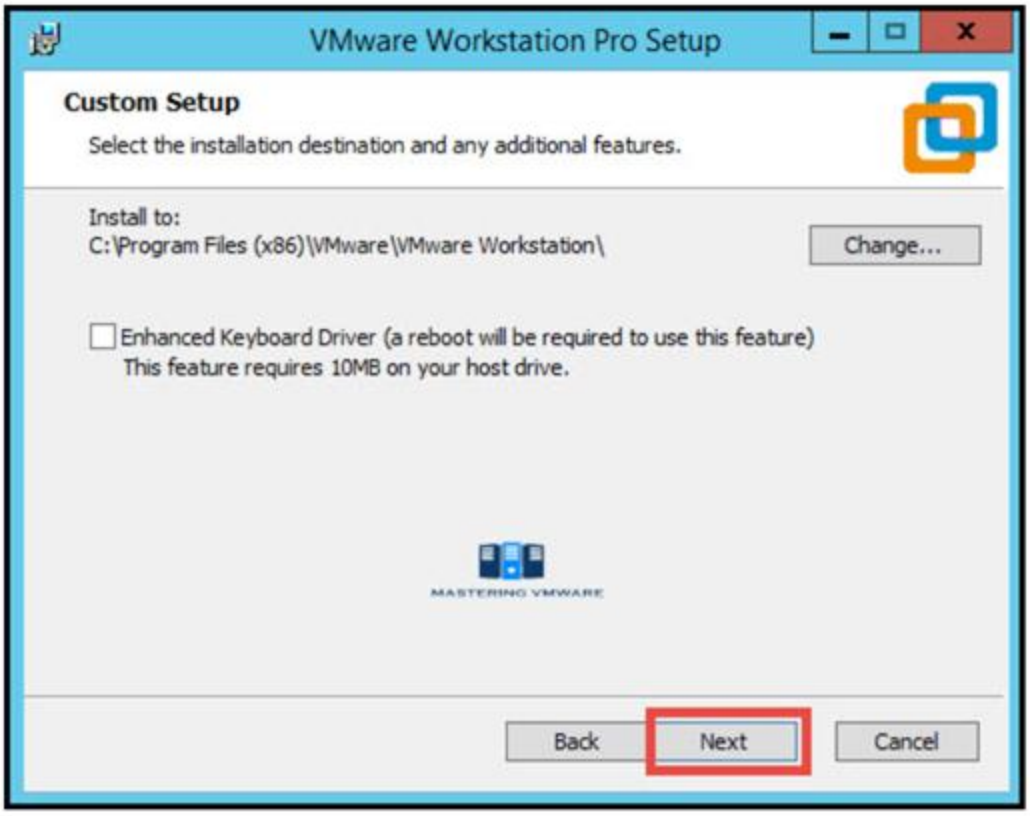
7. Read and Accept the VMware End User license agreement.

Click Next to Continue.



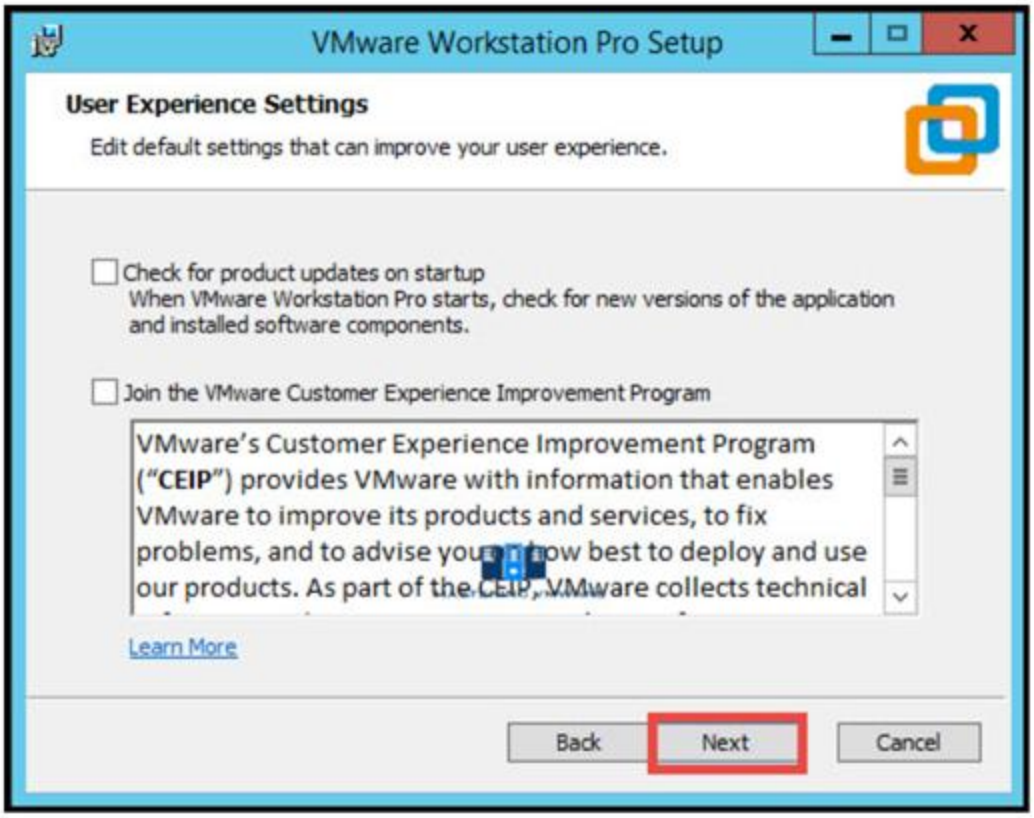
8. Specify the Installation directory. You can also enable Enhance keyboard driver here.

Click Next to continue.



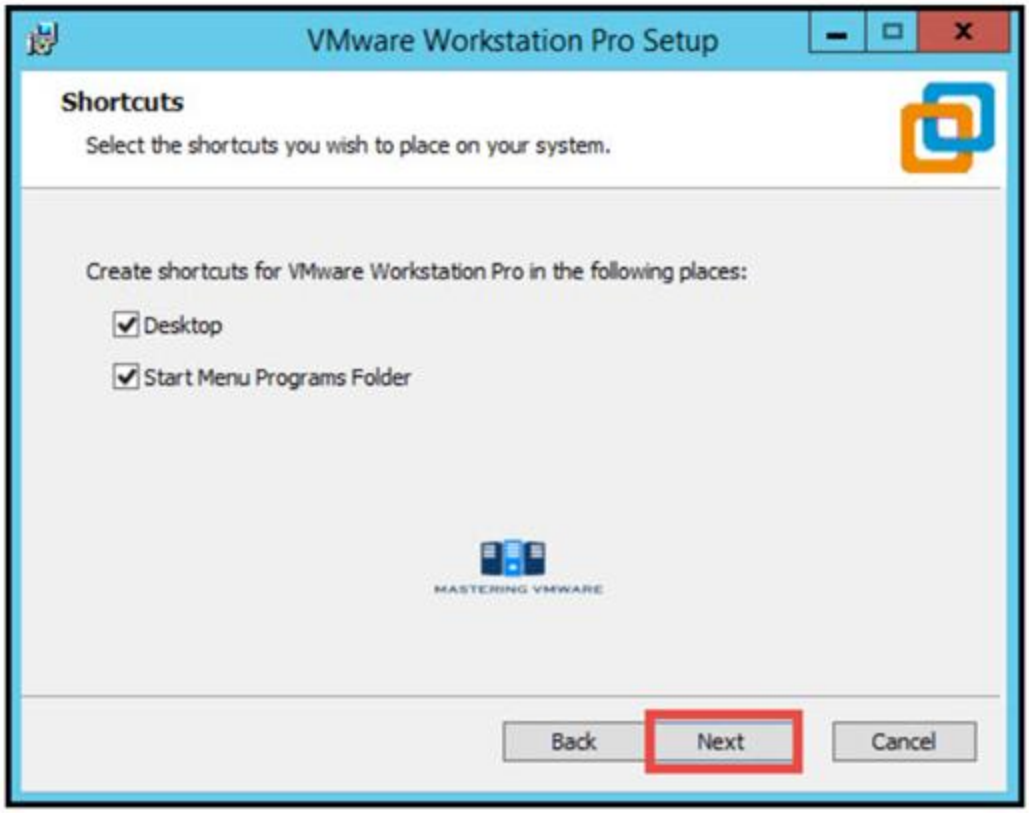
9. You can enable product startup and join the VMware Customer experience Improvement program here.

Click Next to Continue.

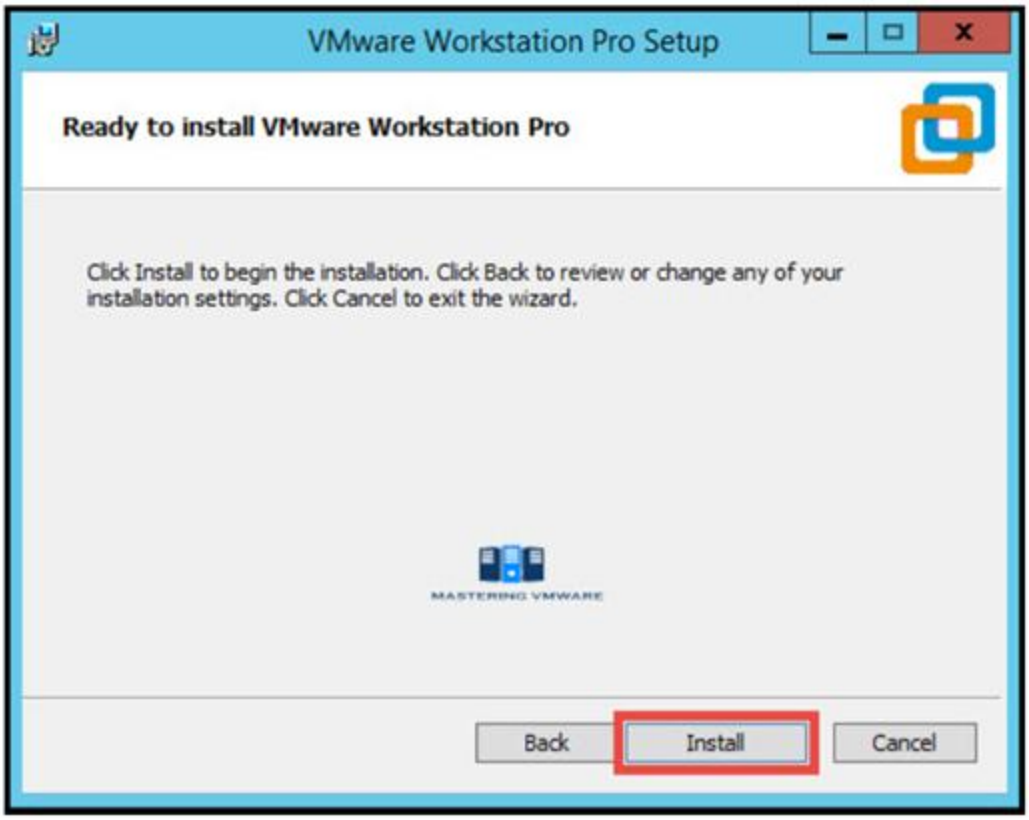


10. Select the shortcuts you want to create for easy access to VMware Workstation.

Click Next to Continue.



11. Click Install button to start the installation.

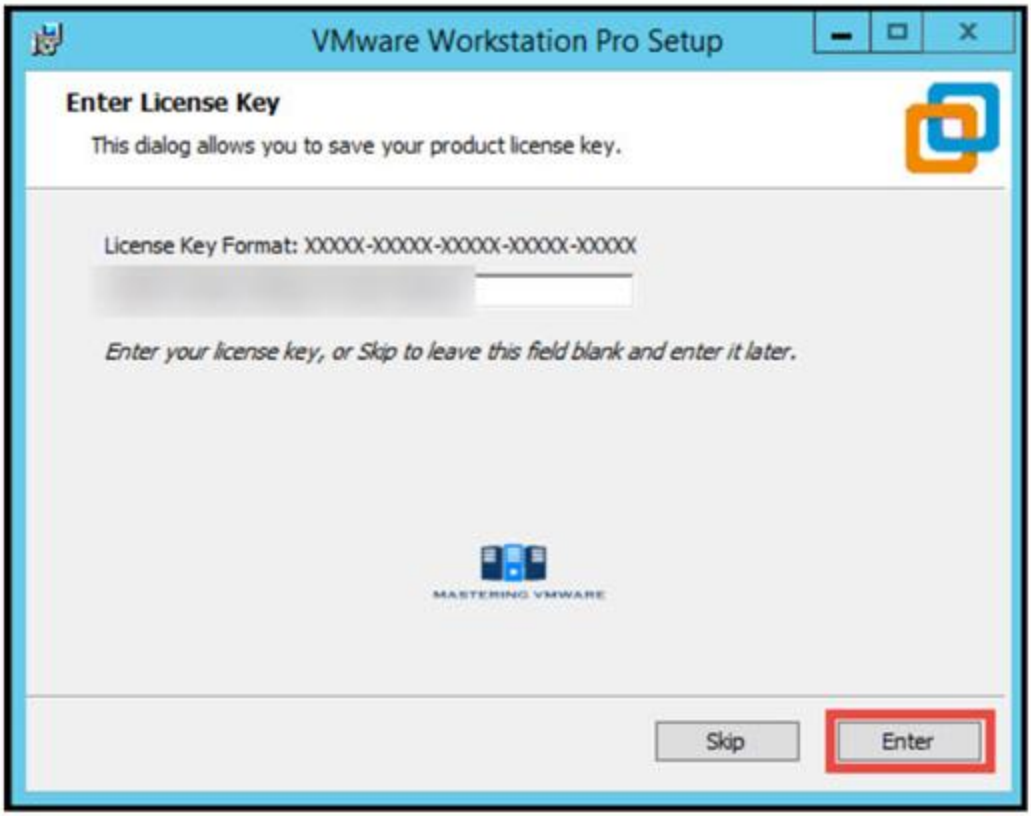


12. Installation will take just few seconds to complete.

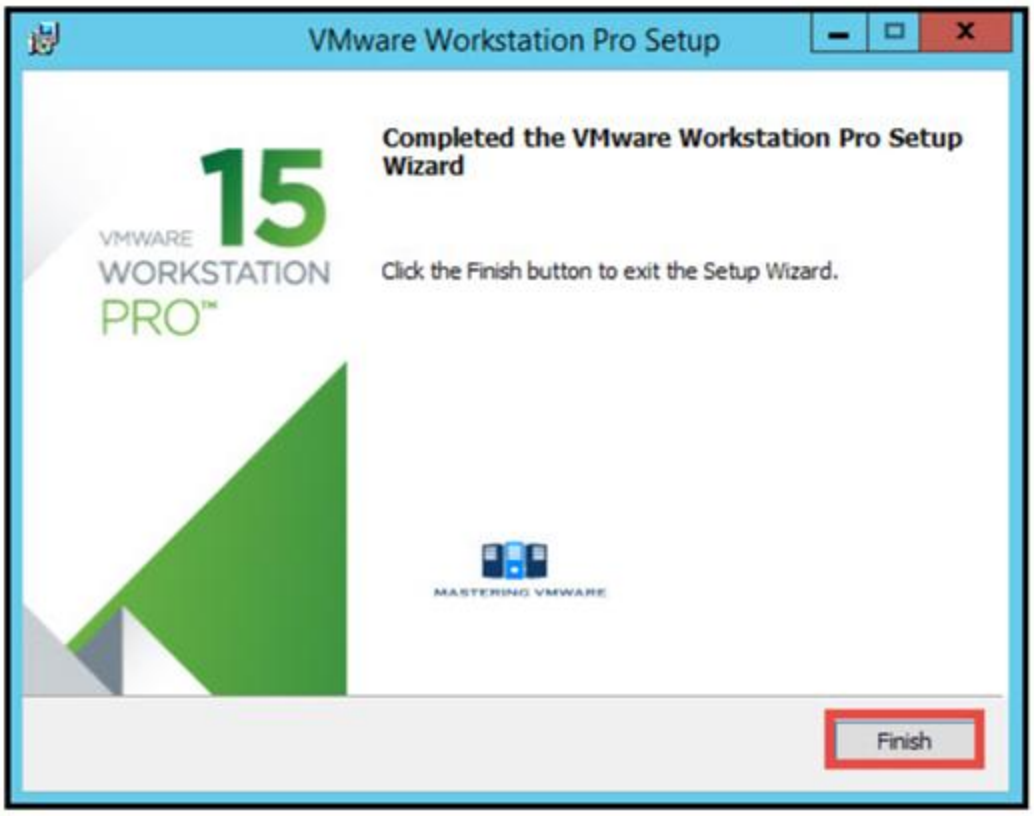
If you have license-key then click on License to enter the license or you can also click Finish to exit the Installer.

13. Provide the License Key for VMware Workstation Pro.

Press Enter to continue.



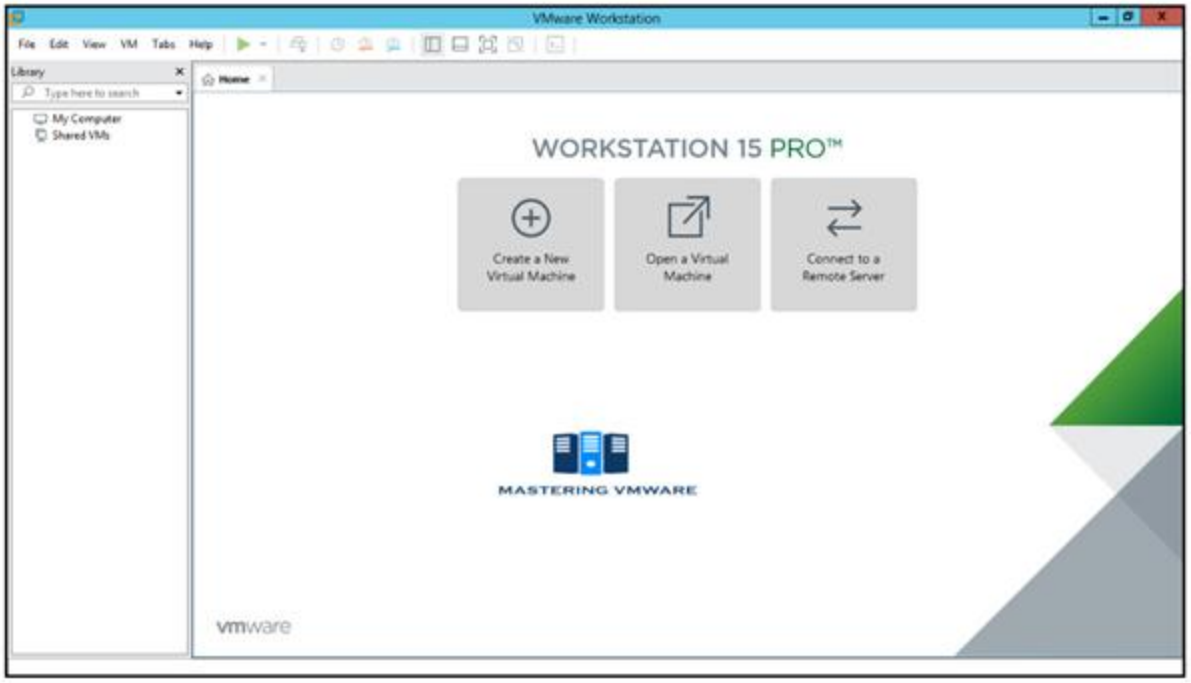
14. Click Finish to exit the wizard.



15. That's it we have successfully installed VMware Workstation Pro.

Now you can start the VMware Workstation Pro by clicking on the shortcut on Desktop.

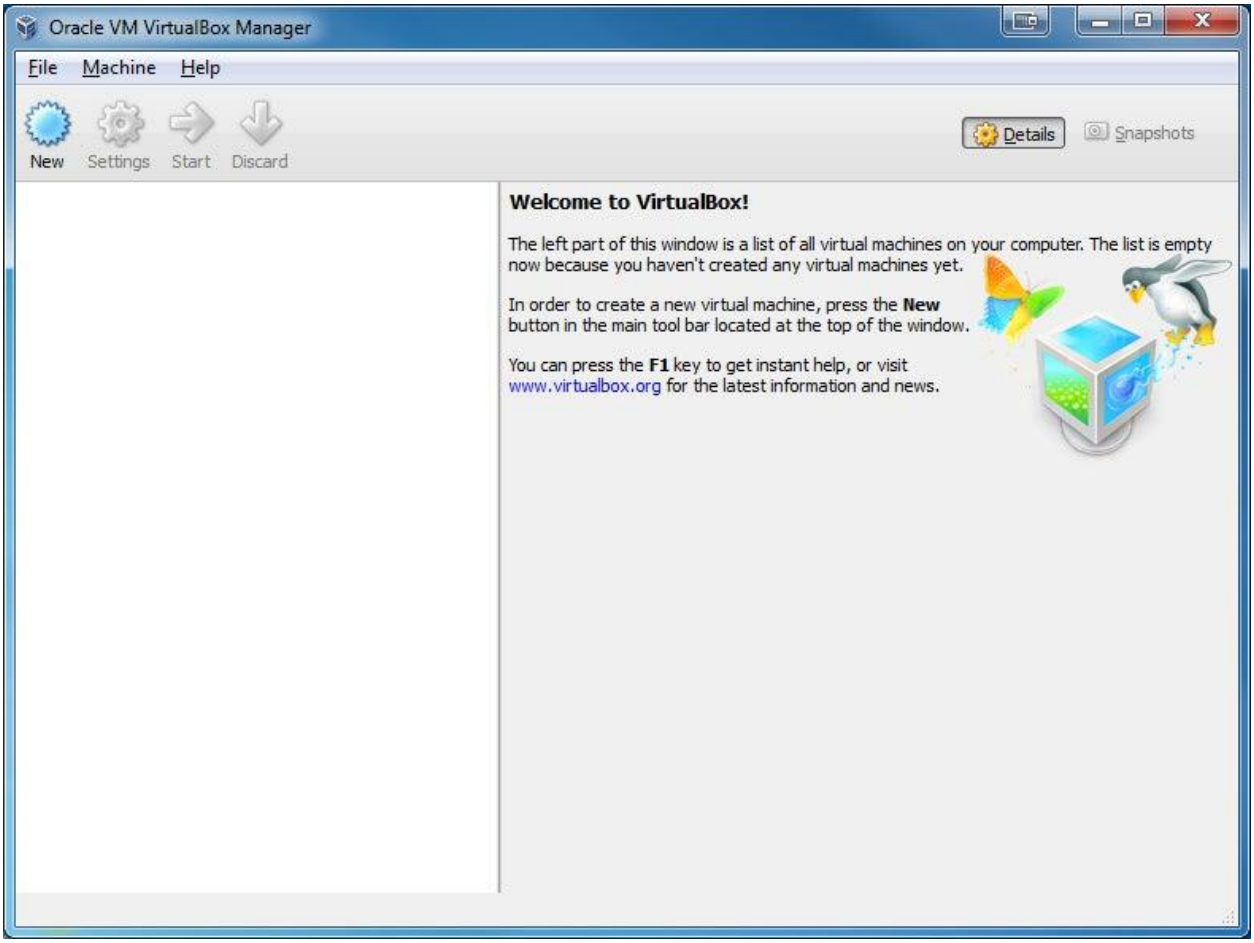
Below is the Home screen of the VMware Workstation pro which you will see every time when you start Workstation.



VMware successfully setup and installed.

Q4: Write down steps to create virtual machines (with appropriate screen shot).

Step 1: Download and Install VirtualBox

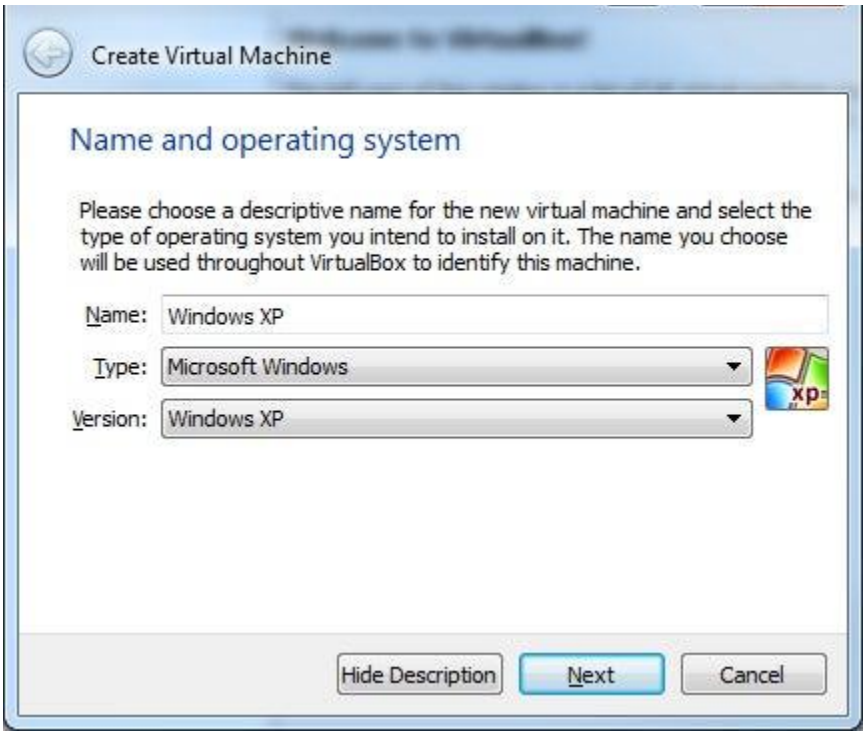


Download VirtualBox from https://www.virtualbox.org/wiki/Downloads. Select the version for your Operating System.

Install VirtualBox.

Keep all of the default settings.
You will be prompted to install several Oracle components. Install all of them.

Step 2: Create a Virtual Machine

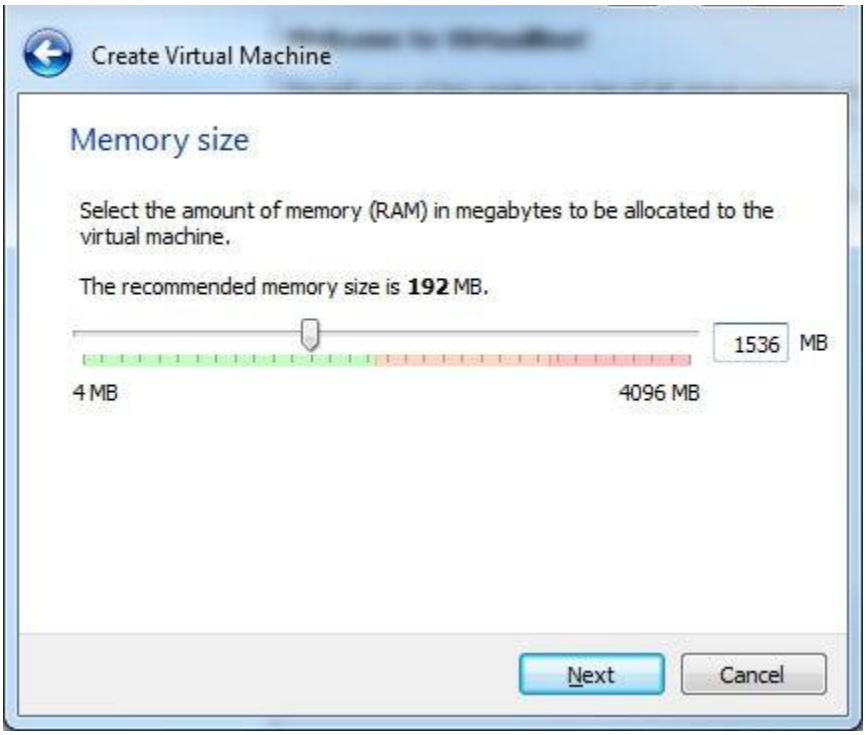


Start VirtualBox and Click on 'New' in the menu.

Enter the Name of your VM. This is how you will identify it in VirtualBox so name it something meaningful to you.

Select Type and Version. This depends on what OS you are installing.

Step 3: Allocate Memory

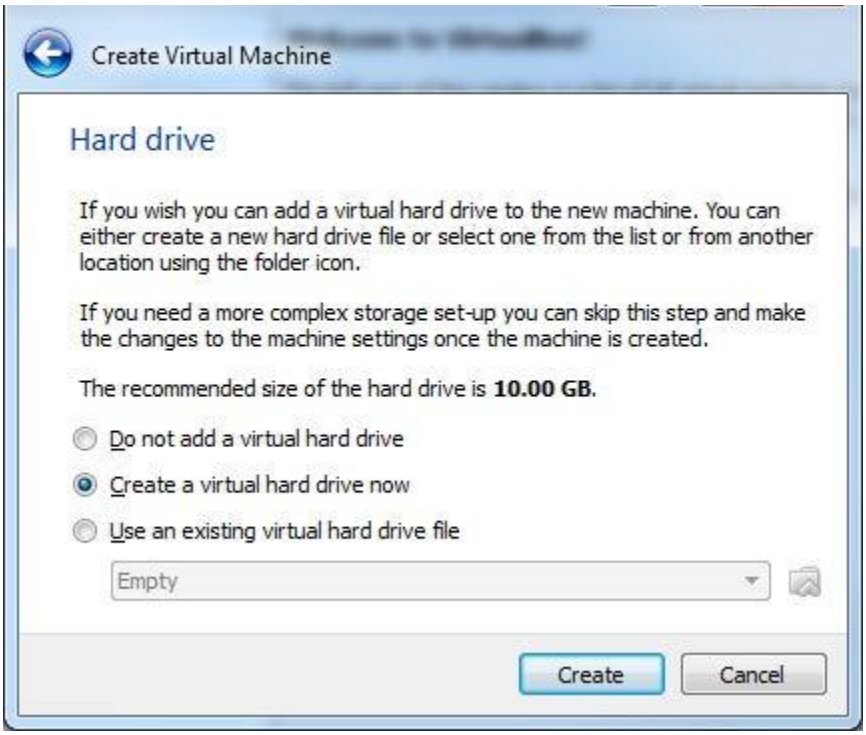


This depends on how much memory you have on your host computer. Never allocate more than half of your available RAM.

If you are creating a Windows VM I recommend at least (1-2 GB)
If you are creating a Linux VM I recommend at least (512 MB)

NOTE: Stay within the green target amount.

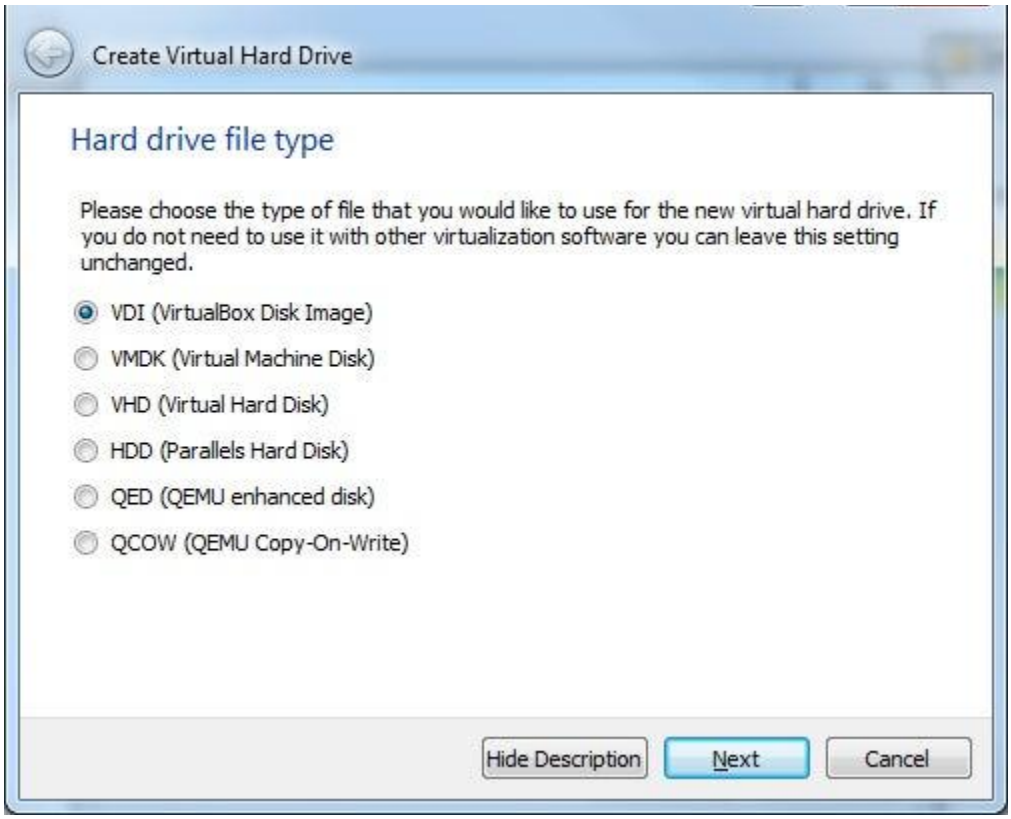
Step 4: Setup the Hard Drive



If you already have an existing VM that you want to add select "Use an existing Virtual hard drive file."

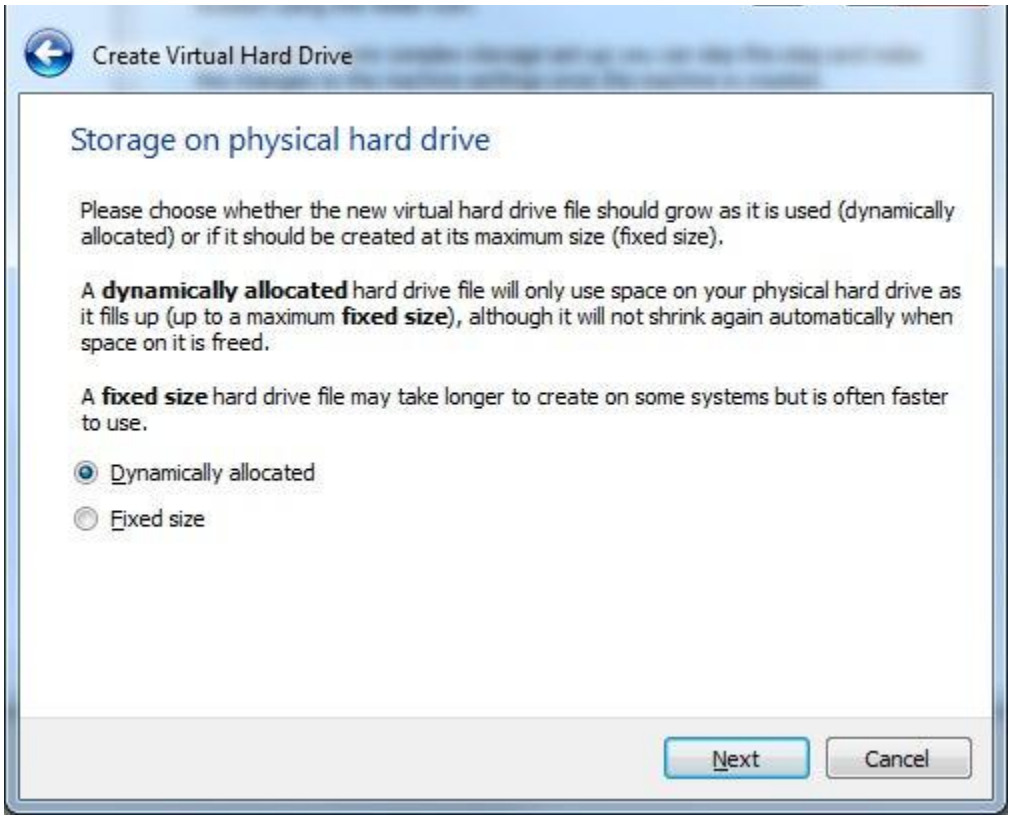
Otherwise select "Create a virtual hard drive now."

Step 5: Select Hard Drive File Type



Select 'VDI.' This is usually the best option.
The VM will be stored in a single file on your computer with the .vdi extension.

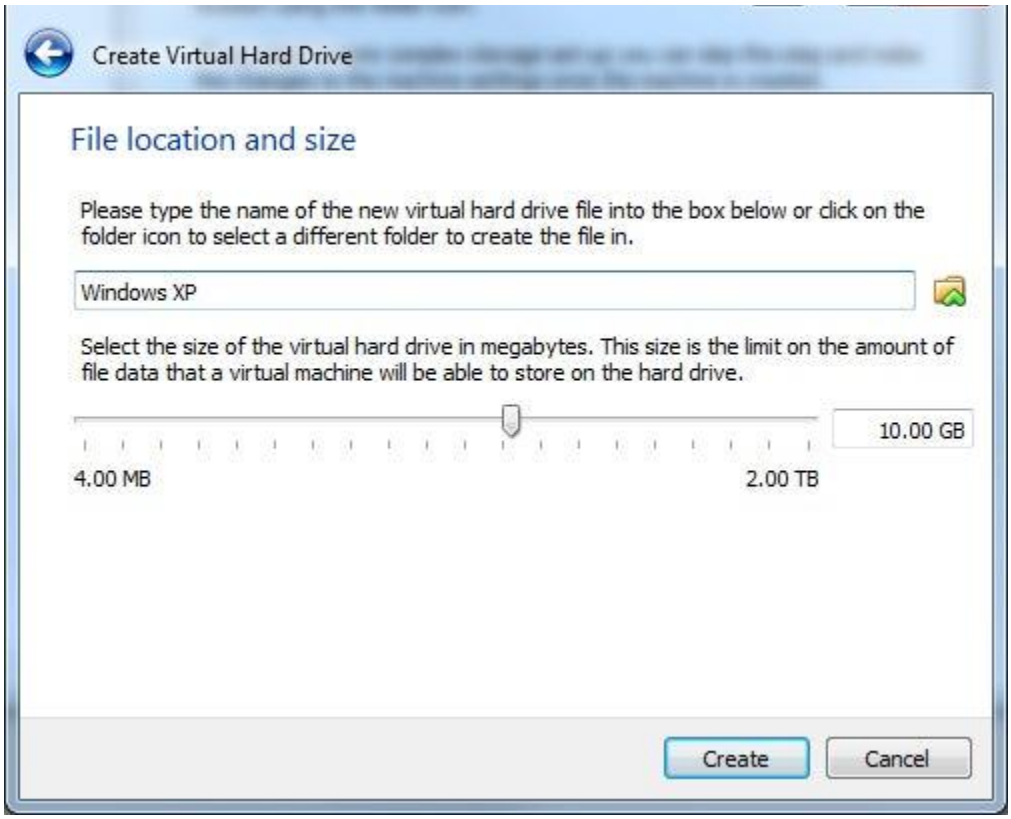
Step 6: Select Storage on Physical Hard Drive



I recommend you choose "Dynamically allocated." This will save space on your computer.

NOTE: "Fixed size" may have a slightly better performance, but will take up more space on your computer.

Step 7: Setup File Location and Size

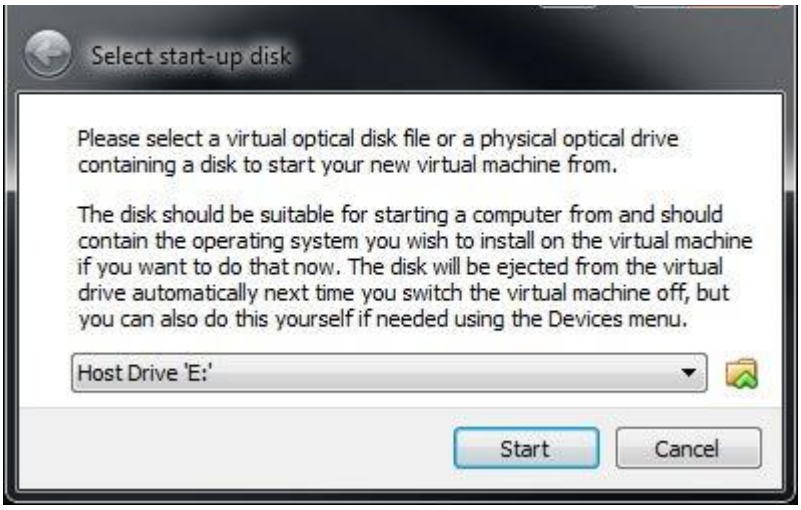


By default, Virtualbox selects the minimum size you should choose.
Depending on what you want to do with the VM you may want to select a bigger size.

NOTE: This will take up space on your computer. Do not select more space than you are willing to give the VM

You have created a VM. Now you need to install an OS on it.

Step 8: Install the Operating System



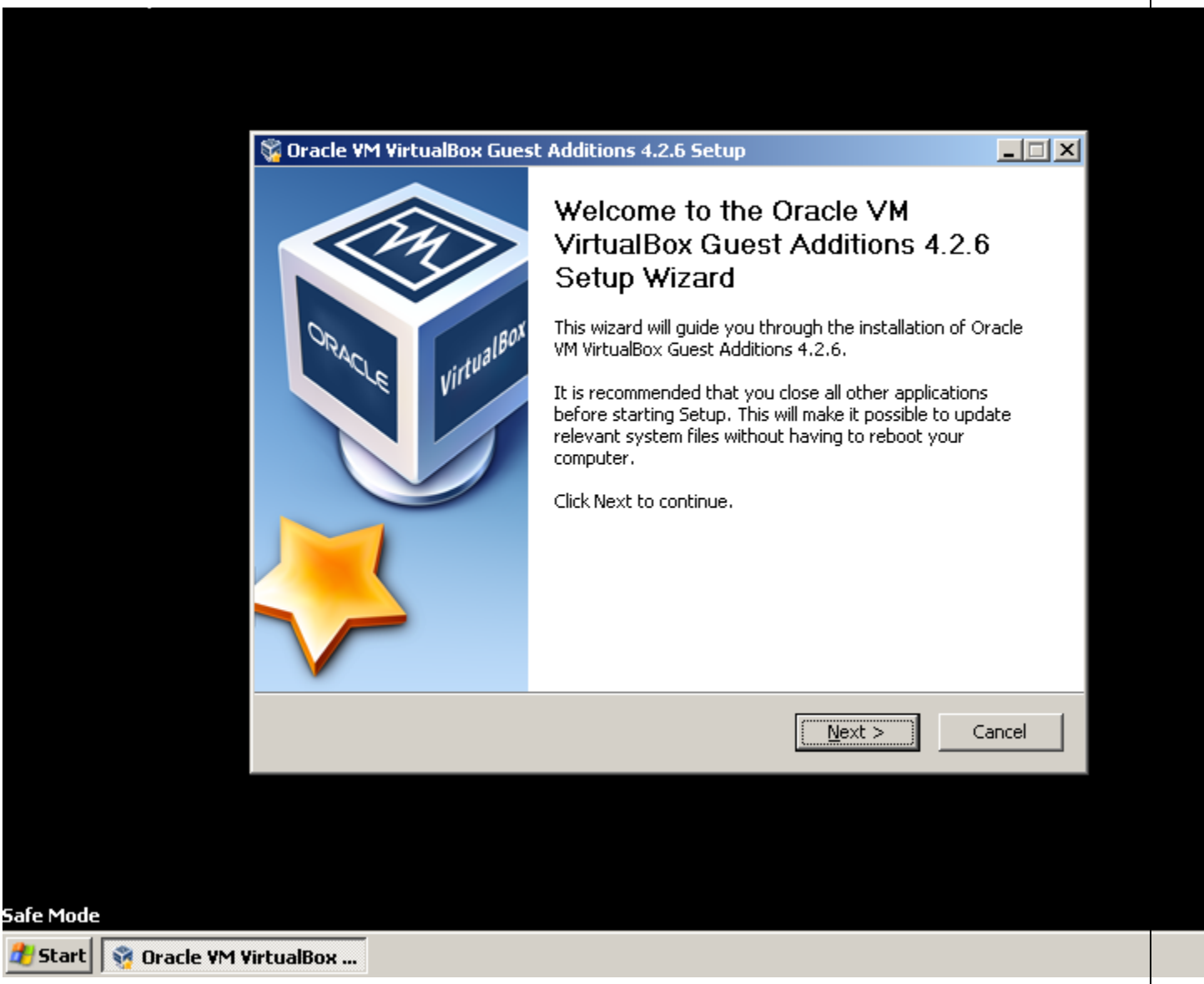
Double click on your newly created VM (It will be on the left hand side and will have the name you gave it in Step 2).

Browse to your installation media or .iso file.

Finish installation.

The actual installation process will depend on which OS you are installing.

Step 9: Install Guest Additions



Guest additions add more functionality to your VM, including the option to make the VM fullscreen.

NOTE: You must install guest additions in safe mode (Windows) to enable 2D and 3D acceleration.

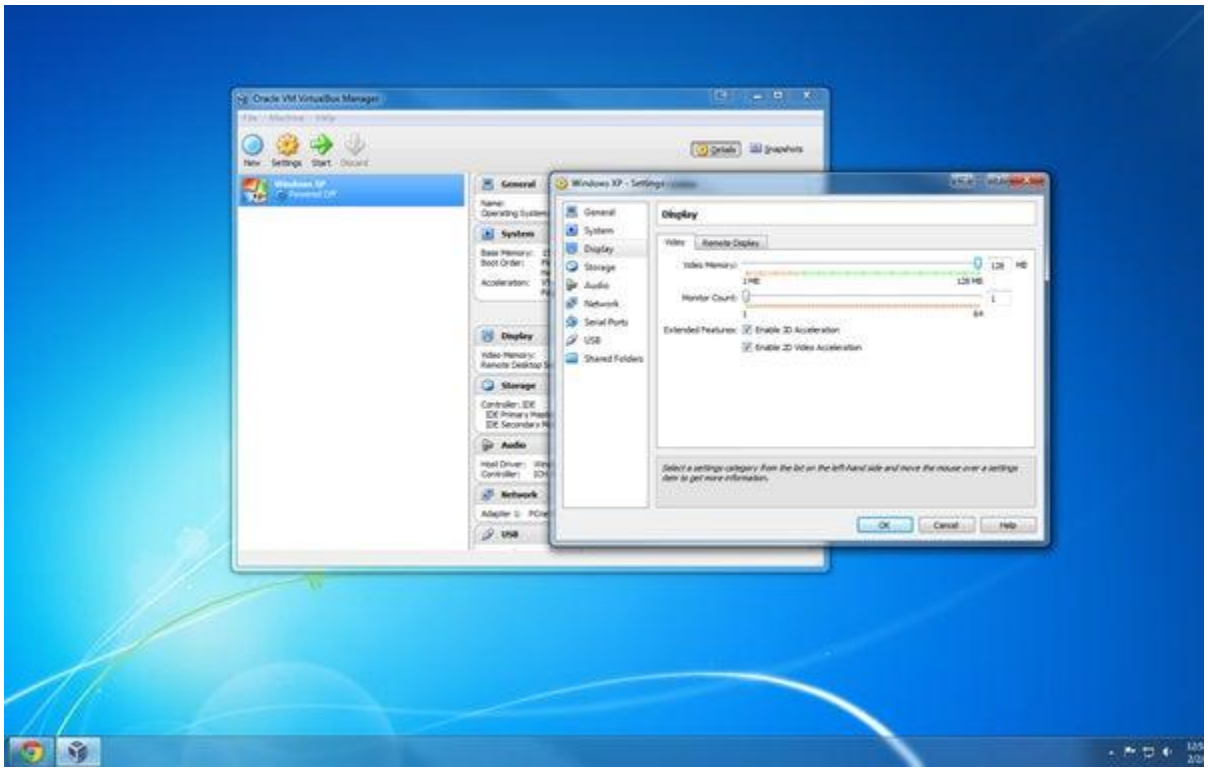
Boot in Safe Mode.
Instructions for Windows XP

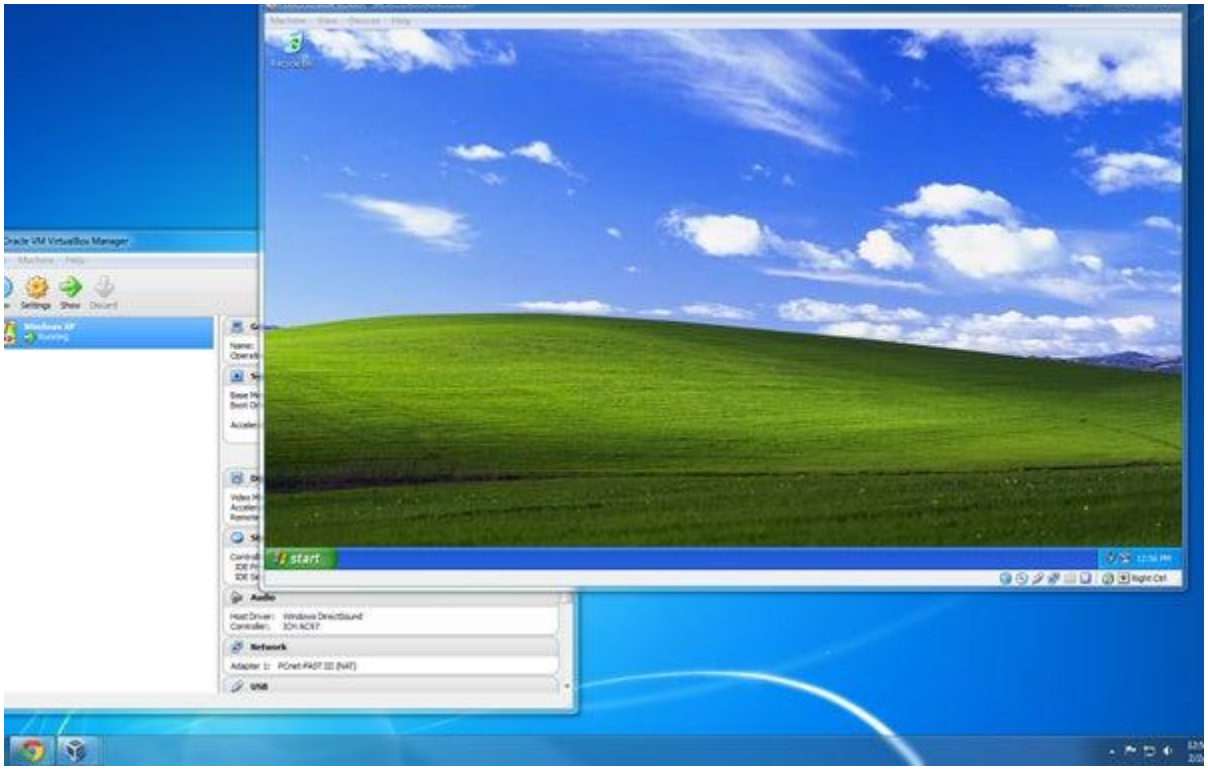
When you have booted in Safe Mode, click Devices --> Install Guest Additions

Follow the Prompts and Install.

Shutdown the VM.

Step 10: Update Settings for 2D and 3D Acceleration





Go to the Display settings for your VM.

Check both 2D and 3D acceleration.

Increase Video Memory (I recommend 128 MB)

Start your VM.

Q.5 what do you mean by system image?

A **system image** refers to an exact replica or snapshot of the entire system, including the operating system (OS), installed applications, settings, and all files stored on the disk. It captures everything on the disk as it exists at the time the image is created. This includes the OS, system files, user data, and configuration settings, allowing you to restore the system to its exact state if needed.

Key Points About a System Image:

1. **Complete Backup**: A system image is a comprehensive backup that includes all files and configurations. It is different from a file-level backup, which typically only saves specific files or folders.
2. **Restoration**: The main use of a system image is for **disaster recovery**. If your system crashes, you can restore it back to its previous state using the system image, which can save a lot of time compared to reinstalling the OS, applications, and reconfiguring settings manually.
3. **Cloning**: A system image can also be used to **clone** a system onto another machine, especially useful when migrating to a new hard drive or setting up multiple systems with identical configurations.
4. **Tools to Create**: You can create a system image using various tools, such as built-in utilities in the operating system (like **Windows Backup and Restore**), or third-party software like **Acronis True Image**, **Macrium Reflect**, or **Clonezilla**.
5. **File System Integrity**: A system image preserves the integrity of the file system and the boot record, ensuring that everything, including system partitions, is exactly copied.

Example of a Use Case:

Imagine you've set up your computer with a particular configuration, applications, and files. If your system encounters a critical failure, rather than trying to recover files and reinstall everything, you can simply restore the system image to bring everything back exactly as it was.

In short:

A **system image** is a complete, exact replica of the system, providing a way to restore the system to its original state quickly and easily after a failure, making it a crucial tool for backup and recovery.

Q.6 List out steps to create system image backup.

Ans:

Creating a **system image backup** is a crucial step for data protection and disaster recovery. The process involves creating a full copy of your operating system, installed programs, settings, and files. Below are the steps to create a **system image backup** on **Windows 10/11** (using built-in tools). I'll also provide steps for using **third-party software** at the end.

Creating a System Image Backup on Windows 10/11 (Using Built-in Tools)

Step 1: Open Control Panel

1. Press **Windows + S** to open the search bar.
2. Type **Control Panel** and select it from the search results.

Step 2: Navigate to Backup and Restore

1. In the Control Panel, select **System and Security**.
2. Click on **Backup and Restore (Windows 7)**. (Even on Windows 10/11, this is the same tool.)

Step 3: Start Creating a System Image

1. On the left panel, click on **Create a system image**.
2. A new window will appear asking where you want to save the system image. You have several options:
    - **On a hard disk** (Select an external hard drive or another internal disk).
    - **On one or more DVDs** (This is not recommended as it requires a lot of DVDs and can be cumbersome).
    - **On a network location** (If you have a network drive set up, you can use this option).
3. Choose your preferred location (external hard drive is usually the most reliable and fastest).
4. Click **Next**.

Step 4: Confirm the Drives to Include

1. You will be shown the drives that will be included in the system image. The **C: drive** (where Windows is installed) will be selected by default. You can choose to include other drives if you want them backed up too.
2. Ensure **System Reserved** and **C:** are selected.
3. Click **Next**.

Step 5: Start the Backup

1. You'll be prompted to confirm your settings. Review your choices, and if everything looks good, click **Start Backup**.
2. The backup process will begin. This can take anywhere from several minutes to a few hours, depending on the size of your data and the speed of your backup drive.

Step 6: Wait for the Backup to Complete

1. Once the backup is complete, you'll see a confirmation message.
2. You'll be asked if you want to create a **System Repair Disc** (recommended). This disc can help you boot your system and restore the backup if needed.
    - If you don't have a disc, you can create a **recovery USB** instead (more modern option).
    - If you don't have any writable media, you can skip this step, but it's highly recommended for recovery purposes.
3. After completing the backup and optionally creating a repair disc, click **Close**.

Restoring a System Image Backup (In Case of System Failure)

If you need to restore the system image backup, follow these steps:

1. Connect the external drive where the system image is stored.
2. Restart your computer, and press the required key (usually **F8**, **F12**, or **Esc**) to enter the **boot menu**.
3. Select **Repair your computer**.
4. Go to **Troubleshoot** → **Advanced Options** → **System Image Recovery**.
5. Choose the system image you want to restore from the backup.
6. Follow the on-screen instructions to complete the restoration.

Using Third-Party Tools to Create a System Image Backup

If you prefer using third-party software, here are some popular options:

1. **Macrium Reflect** (Free and Paid versions):

- Install Macrium Reflect from Macrium's official website.
- Open the application, select your main drive (usually C:), and choose **Create an image of the partition(s)**.
- Select the destination (external drive, network share, etc.).
- Start the backup and follow the prompts.

2. **Acronis True Image** (Paid):
   - Download and install Acronis True Image from Acronis website.
   - Open the software, select **Backup** > **Disk & Partition Backup**.
   - Choose the source (the system drive) and the destination.
   - Start the backup and follow on-screen instructions.
3. **Clonezilla** (Free, Open-source):
   - Clonezilla is a more advanced, free, open-source tool that can create system images. It requires booting from a USB or CD and has more configuration options.
   - Follow Clonezilla's official documentation for creating an image.

Summary:

1. Open **Control Panel** > **Backup and Restore (Windows 7)**.
2. Click **Create a system image** and choose the destination (external hard drive is recommended).
3. Confirm the drives you want to back up and click **Start Backup**.
4. Wait for the process to finish, and optionally create a **System Repair Disc**.
5. To restore, go to **Troubleshoot** > **System Image Recovery** from the recovery options.

Let me know if you need more details or help with a specific step!

B.2 Conclusion:

In conclusion, this experiment involved launching a **virtual machine (instance)** on **AWS** to implement **Infrastructure as a Service (IaaS)**. By creating and configuring an EC2 instance, we gained practical experience with cloud computing, including provisioning, networking, and securing virtual machines in the **public cloud**. The exercise demonstrated the flexibility and scalability of AWS, showcasing how cloud resources can be easily managed and deployed. This hands-on experience provided a deeper understanding of cloud infrastructure and its application in real-world scenarios.