



**SAVEETHA SCHOOL OF ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CSA15 – CLOUD COMPUTING AND BIG DATA ANALYTICS  
LABORATORY**

**INDEX**

S.N O	DATE	EXPERIMENT NAME	MARK S	SIGN
1.		Create a simple cloud software application and provide it as a service using any Cloud Service Provider to demonstrate Software as a Service (SaaS).		
2.		Create a Virtual Machine with 1 vCPU, 2GB RAM and 15GB storage disk using a Type 2 Virtualization Software		
3.		Create a Virtual Hard Disk and allocate the storage using VM ware Workstation		
4.		Create a Snapshot and Cloning of a VM and Test it by loading the Previous Version/Cloned VM		
5.		Demonstrate Infrastructure as a Service (IaaS) by Creating a Virtual Machine using a Public Cloud Service Provider (Azure/GCP/AWS), configure with minimum CPU, RAM, and Storage and Launch the VM image.		
6.		Create a Simple Web Application using Java or Python and host it in any Public Cloud Service Provider (Azure/GCP/AWS) to demonstrate Platform as a Service (PaaS)		
7.		Create a Storage service using any Public Cloud Service Provider (Azure/GCP/AWS) and check the public accessibility of the stored file to demonstrate Storage as a Service		
8.		Create a SQL storage service and perform a basic query using any Public Cloud Service Provider (Azure/GCP/AWS) to demonstrate Database as a Service (DaaS)		
9.		Perform the basic configuration setup for Installing Hadoop 2.x like Creating the HDUSER and SSH localhost		
10.		Install Hadoop 2.x and configure the Name Node and Data Node.		
11.		Launch the Hadoop 2.x and perform MapReduce Program for a Word Count problem		

## **EXP NO 1: CREATE A SIMPLE CLOUD SOFTWARE APPLICATION AND PROVIDE IT AS A SERVICE USING ANY CLOUD SERVICE PROVIDER TO DEMONSTRATE SOFTWARE AS A SERVICE (SAAS).**

### **DATE:**

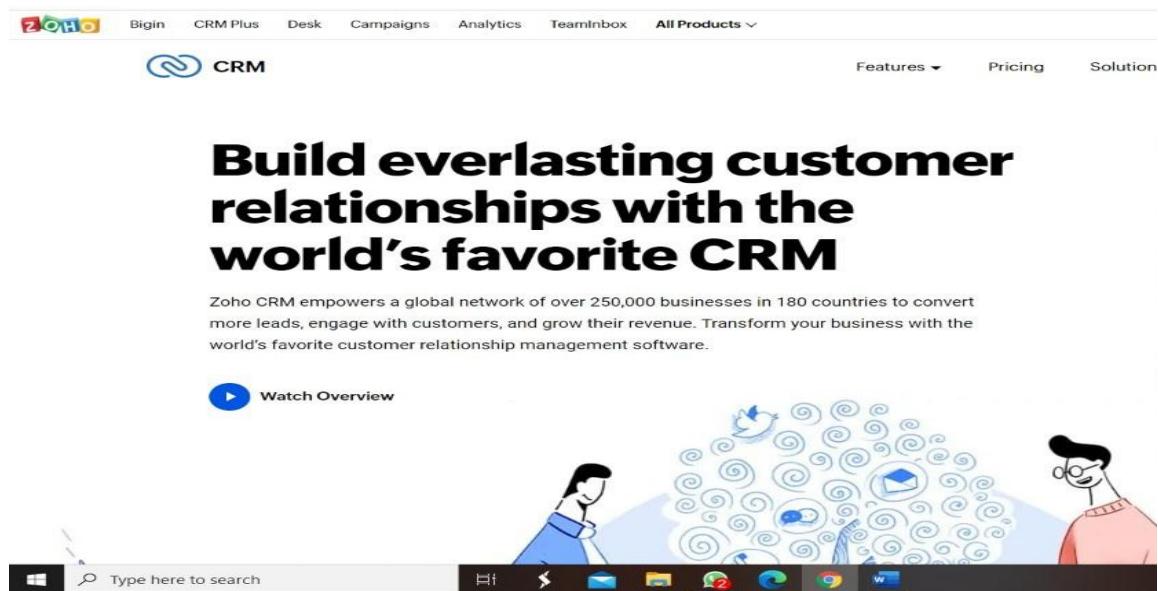
**AIM:** The aim of this project is to develop a simple cloud software application and provide it as a service using any cloud service provider to demonstrate Software as a Service (SaaS) model. The application will be accessible over the internet, allowing users to utilize its functionality without needing to install or maintain the software on their local machines.

### **PROCEDURE:**

1. Choose Application: Select a simple software (e.g., task manager).
2. Design & Develop: Create the application with necessary features.
3. Containerize: Package the application with Docker for easy deployment.
4. Select Cloud Provider: Choose a cloud service provider.
5. Deploy Application: Use the cloud provider's services to host the application.
6. Set Up Domain: Assign a domain name to the application.
7. Implement Security: Ensure SSL/TLS encryption and access control.
8. Monitoring: Set up monitoring tools for performance tracking.
9. Testing: Conduct thorough testing for functionality and security.
10. Result: Users can access and use the application online.

### **IMPLEMENTATION: STEP1:**

**GOTO ZOHO.COM**



## STEP 2: LOGINTO THE ZOHO.COM

You're one step away from getting started with Zoho CRM.

[Get Started for Free](#)

By creating a new account, you agree to our [Terms of Service](#).  
Learn more about Zoho CRM [here](#).

## STEP 3: SELECT ONE APPLICATION

### Create Application

You can create your application from scratch, or install a pre-made application from our gallery.

[Cancel](#)

The screenshot shows the 'Create Application' interface. At the top, there's a search bar with placeholder text 'Start typing to search for applications...'. Below it, a navigation bar includes 'All' (which is underlined), 'Information Technology', 'Business', 'Sales & Marketing', 'Educ...', and a 'More' dropdown. The main area displays eight pre-made application cards arranged in two rows of four:

- Create from scratch**: A card with a plus sign icon and a dashed border.
- Sales Management**: A blue card with a person icon.
- Order Management**: An orange card with a shopping cart icon.
- Employee Management**: A blue card with a person icon.
- IT Asset Tracker**: A green card with a computer monitor icon.
- Event Management**: A blue card with a calendar icon.
- Course Planner**: A black card with a brain icon.
- Expenses**: A green card with a hand icon.

Each card has a 'More Info' link and an 'Install this Application' button.

## STEP 4: ENTER APPLICATION NAME

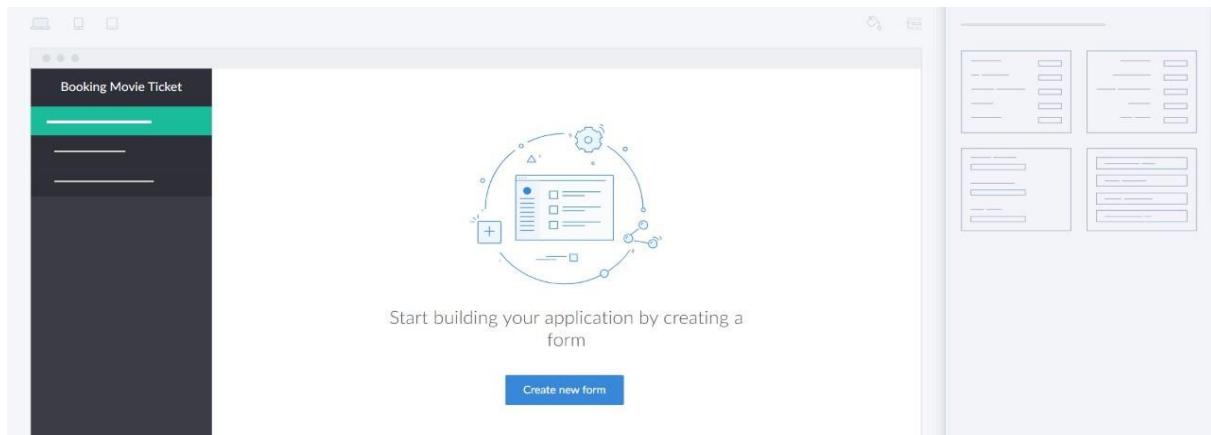
### Create Application

You can create your application from scratch, or install a pre-made application from our gallery.

[Cancel](#)

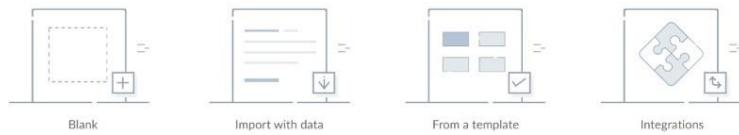
The screenshot shows the 'Create Application' interface with a modal window overlaid. The modal is titled 'Enter Application Name' and contains a text input field with placeholder text 'Examples: Campaign Monitor, Order Management' and a 'Create' button at the bottom. The background shows the same application gallery as in Step 3, with the 'Create from scratch' card highlighted by a dashed border. The other application cards are dimmed.

## STEP 5: CREATED NEW APPLICATION



## STEP 6: SELECT ONE FORM

How would you like to create your form?



## STEP 7: THE SOFTWARE HASE BEEN CREATED.

The screenshot shows the "Booking Movie Ticket" application with a "User Details" form. The form includes fields for Name (First Name and Last Name), Phone (with an Indian number +91 and the number 81234 56789), Email, Date-Time (with a calendar icon), Theatres (with First Name and Last Name), and a Drop Down menu. At the bottom are "Submit" and "Reset" buttons. To the right of the form is a panel titled "Form Customization - Web" which shows "Label placement" options for the form fields.

**Field Properties**

- Field name: Name
- Field link name: Name
- Validation: Mandatory
- Display Fields: Prefix, First Name, Last Name, Suffix
- Data Privacy

## RESULT:

- The task management application is accessible online via a provided domain.
- Users can securely sign up, log in, and manage tasks.
- The application is scalable, reliable, and secure.
- Monitoring ensures smooth operation and quick issue resolution.
- Demonstrates Software as a Service (SaaS) model effectively.

## **EXPNO 2: CREATE A VIRTUAL MACHINE WITH 1 VCPU, 2GB RAM AND 15GB STORAGE DISK USING A TYPE 2 VIRTUALIZATION SOFTWARE**

**DATE:**

**AIM:** Create a virtual machine with 1 vCPU, 2GB RAM, and a 15GB storage disk using a type 2 virtualization software.

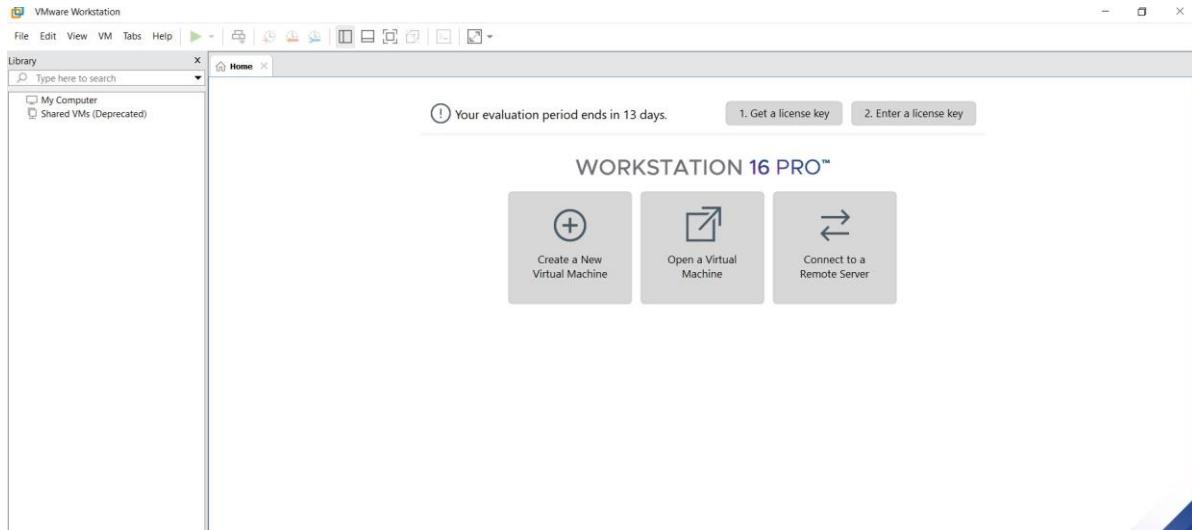
### **PROCEDURE:**

- 1. Install Type 2 Virtualization Software:** Choose and install a type 2 virtualization software like VirtualBox or VMware Workstation on your host machine.
- 2. Create Virtual Machine:** Open the virtualization software and create a new virtual machine.
- 3. Configure Settings:** Specify the settings for the virtual machine, including 1 vCPU, 2GB RAM, and a 15GB storage disk.
- 4. Install Operating System:** Install the desired operating system on the virtual machine.
- 5. Start Virtual Machine:** Start the virtual machine to ensure it boots up properly.

### **IMPLEMENTATION:**

#### **STEP 1:**

#### **DOWLOAD VMWARE WORKSTATION AND INSTALLED AS TYPE 2 HYPERVISOR**

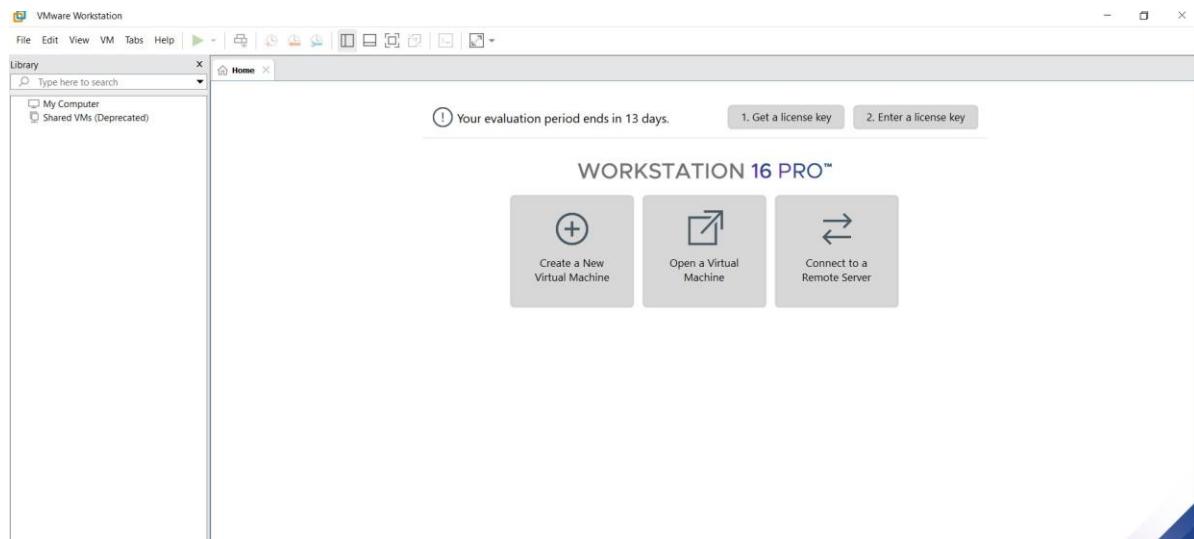


## STEP2: DOWNLOAD UBUNTU OR TINY OS AS ISO IMAGE FILE

### Index of /11.x/x86/release/

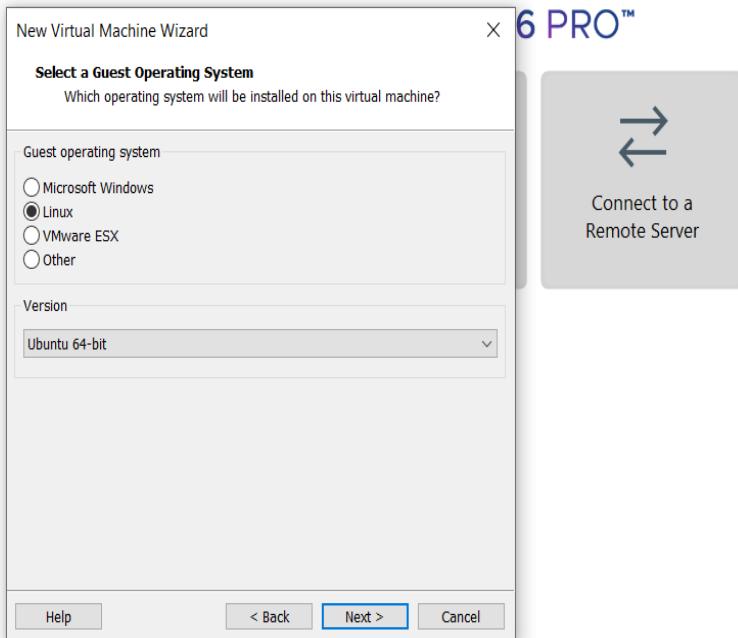
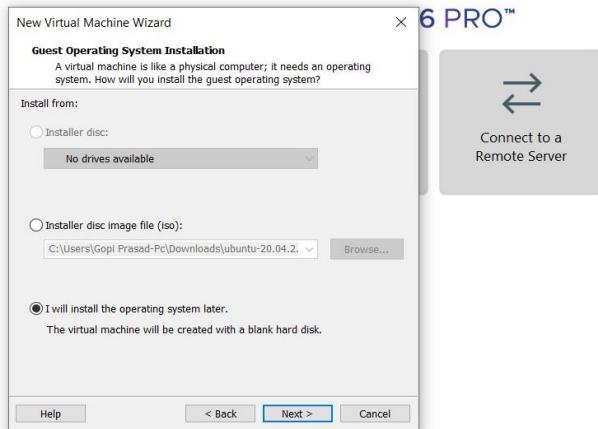
..	/	
distribution_files/		
src/		
Core-11.1.iso	09-Feb-2020 11:50	-
Core-11.1.iso.md5.txt	03-Dec-2019 11:14	
Core-11.1.iso.zsync	01-Apr-2020 07:49	14757888
Core-current.iso	01-Apr-2020 07:49	48
CorePlus-11.1.iso	01-Apr-2020 07:49	50639
CorePlus-11.1.iso.md5.txt	01-Apr-2020 07:50	14757888
CorePlus-11.1.iso.zsync	01-Apr-2020 07:50	216006656
CorePlus-current.iso	01-Apr-2020 07:50	52
TinyCore-11.1.iso	01-Apr-2020 07:50	369358
TinyCore-11.1.iso.md5.txt	01-Apr-2020 07:50	216006656
TinyCore-11.1.iso.zsync	01-Apr-2020 07:50	19922944
TinyCore-current.iso	01-Apr-2020 07:50	52
	01-Apr-2020 07:50	68301
	01-Apr-2020 07:50	19922944

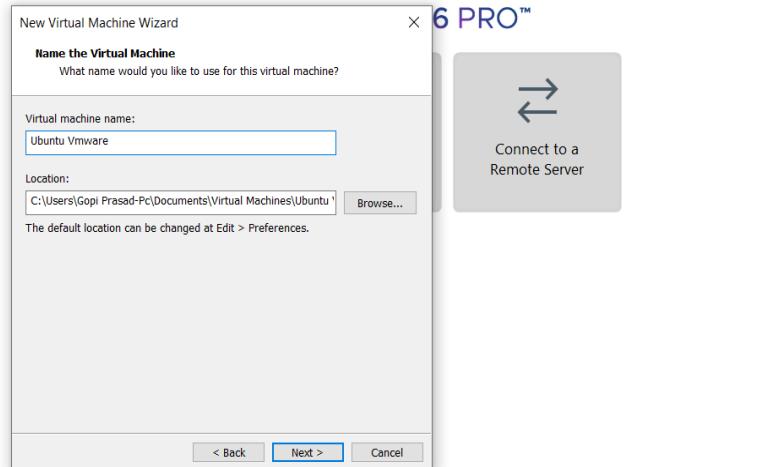
## STEP 3: IN VMWARE WORKSTATION->CREATE NEW VM



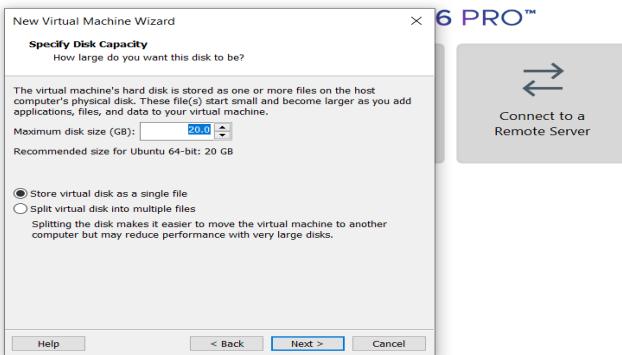
## STEP 4: DO THE BASIC CONFIGURATION SETTINGS.



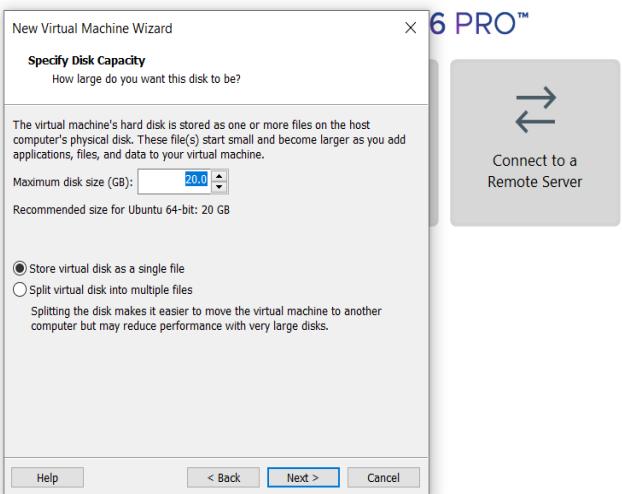


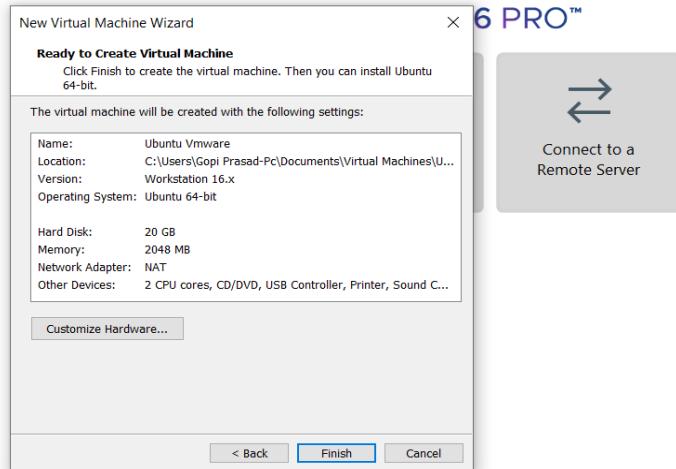


! Your evaluation period ends in 12 days. [1. Get a license key](#) [2. Enter a license key](#)

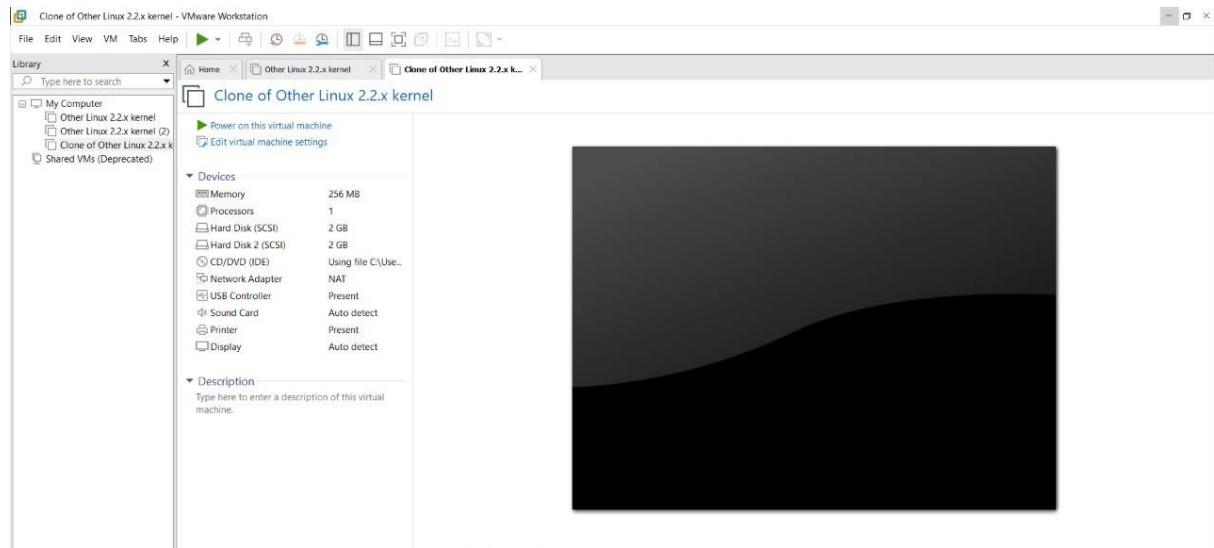


! Your evaluation period ends in 12 days. [1. Get a license key](#) [2. Enter a license key](#)

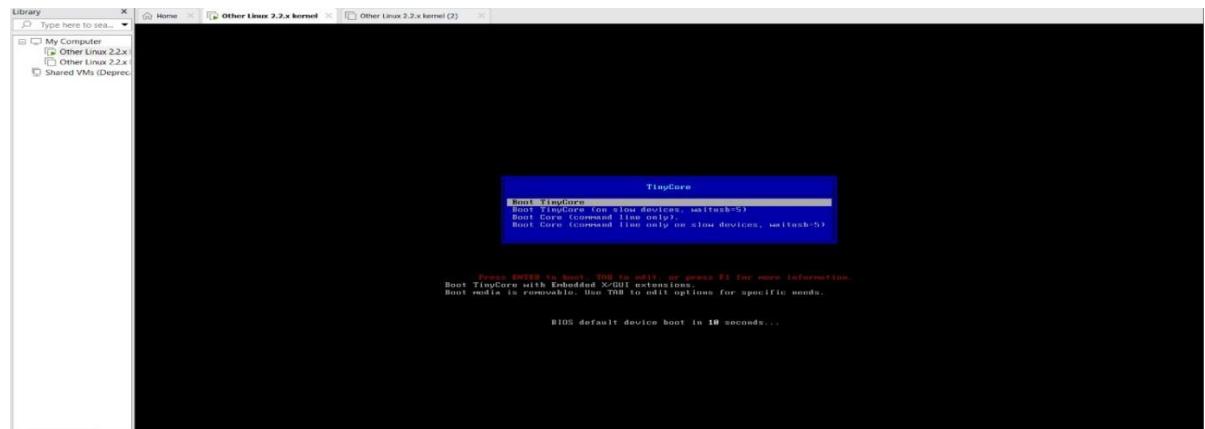




## STEP 4: CREATED TINYOS VIRTUAL MACHINE



## STEP 5: LAUNCH THE VM



## RESULT

**Successfully created a virtual machine with 1 vCPU, 2GB RAM, and a 15GB storage disk using the selected type 2 virtualization software.**

## **EXP 3: CREATE A VIRTUAL HARD DISK AND ALLOCATE THE STORAGE USING VM WARE WORKSTATION**

**DATE:**

**AIM:**

**Create a virtual hard disk and allocate storage using VMware Workstation.**

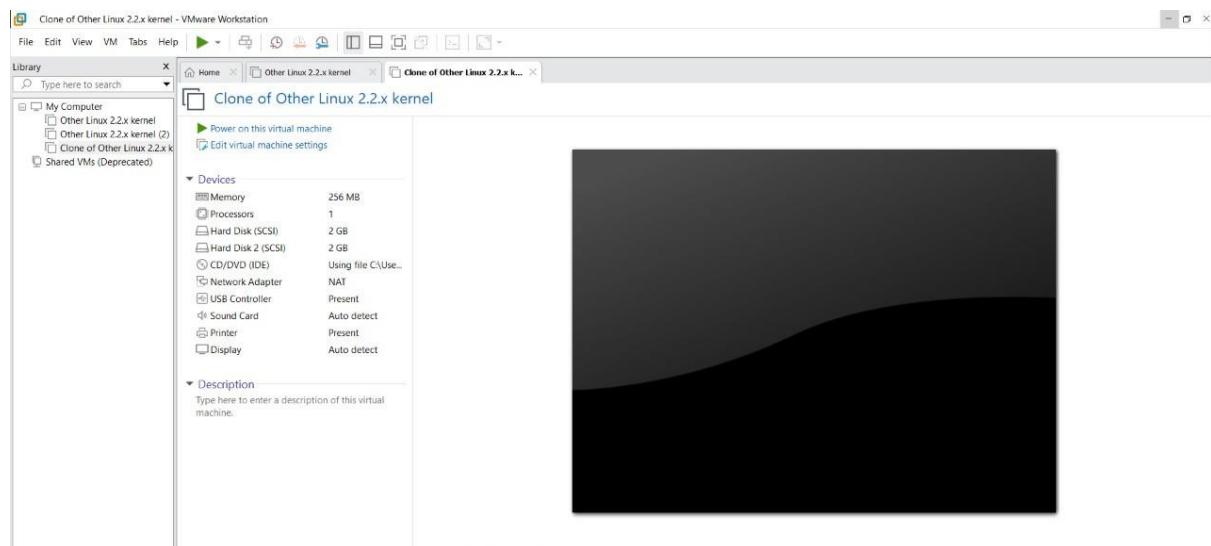
### **PROCEDURE:**

1. Open VMware Workstation: Launch VMware Workstation on your computer.
2. Create New Virtual Machine: Click on "File" > "New Virtual Machine" to start the creation process.
3. Select Typical Configuration: Choose "Typical" configuration and click "Next."
4. Select Guest Operating System: Choose the guest operating system and version, then click "Next."
5. Specify Disk Capacity: Choose "Create a new virtual disk" and specify the disk capacity.
6. Allocate Disk Space: Choose whether to allocate all disk space now or allow it to grow as needed.
7. Choose Disk File: Select the disk file location and name, then click "Finish" to create the virtual hard disk.

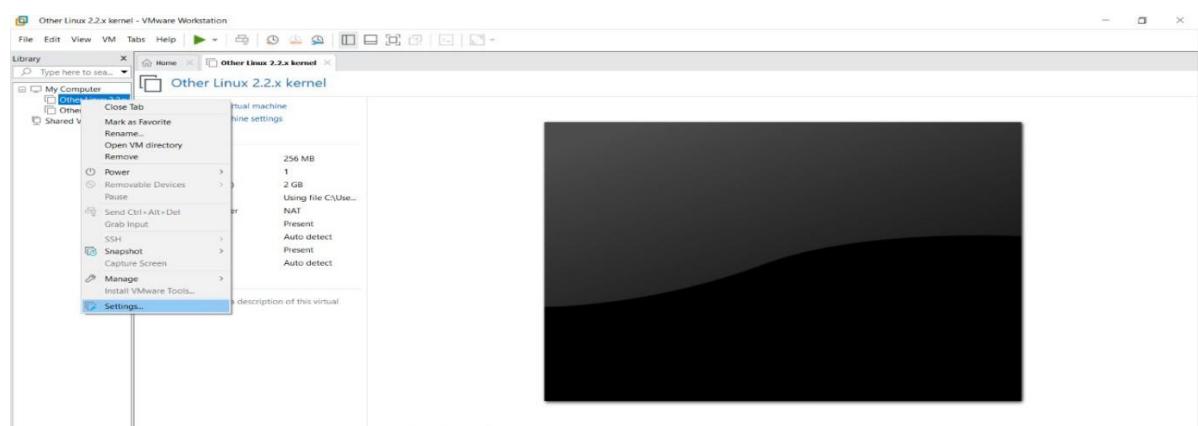
### **IMPLEMENTATION:**

#### **STEP 1:**

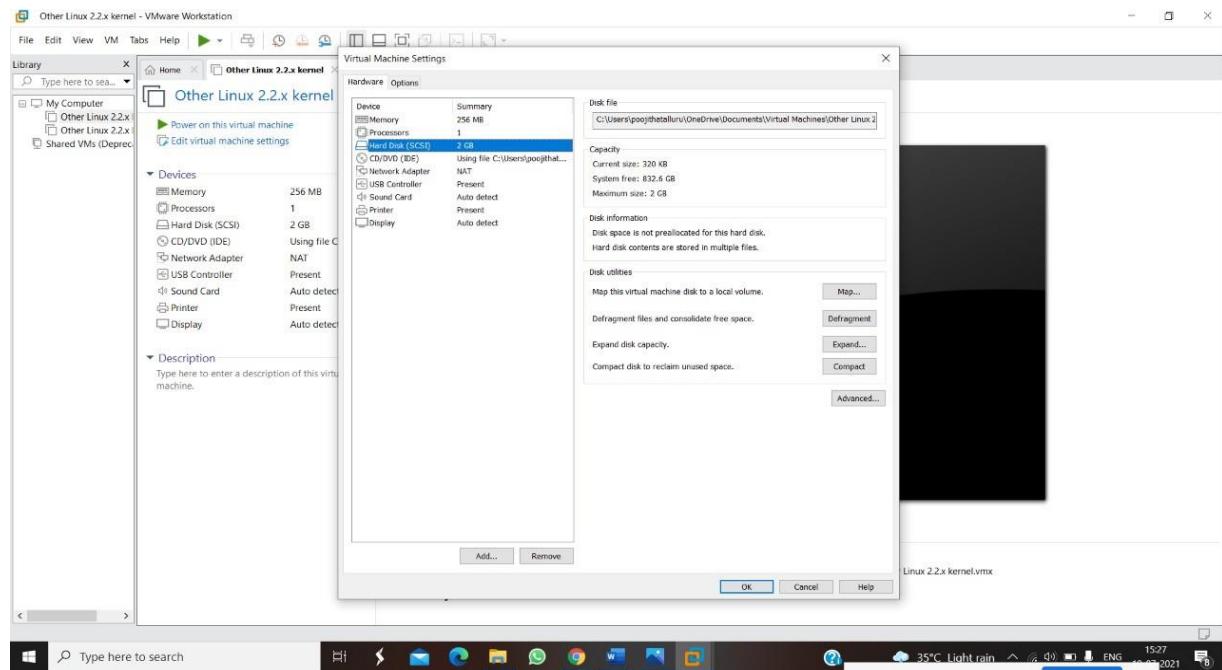
#### **GOTO VM WARE WORKSTATION**



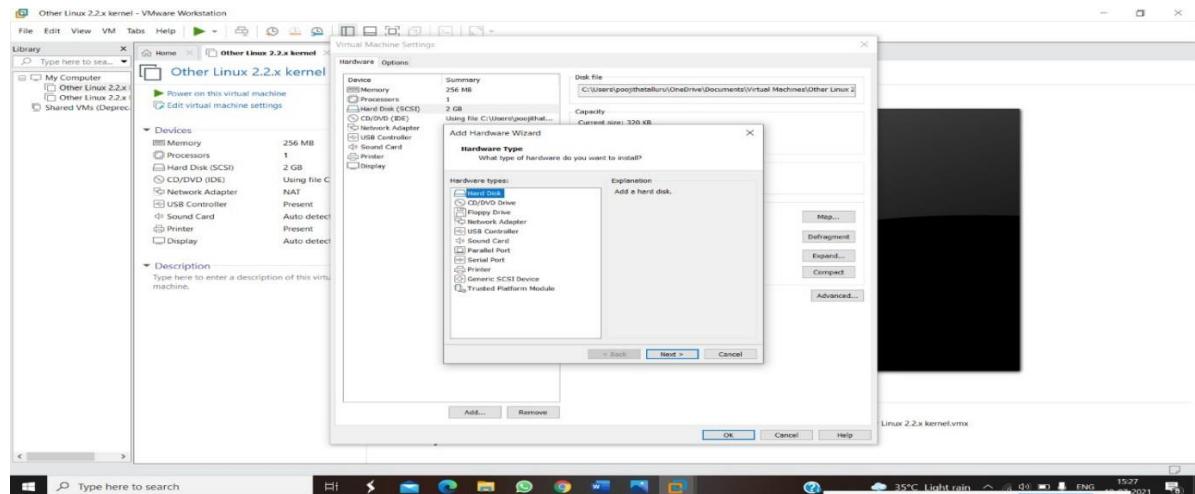
#### **STEP2: RIGHT CLICK THE VM AND GOTO THE SETTINGS**

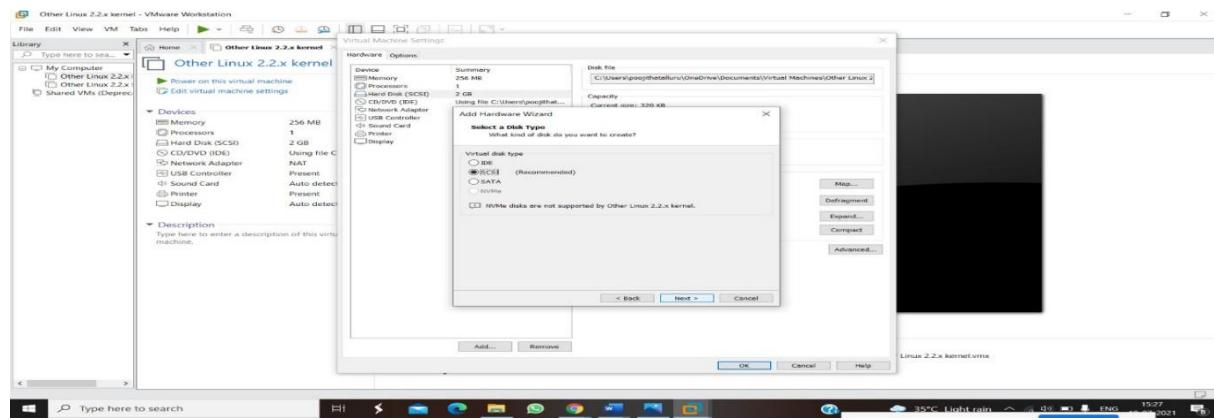


## STEP 3: ADD HARDWARE WIZARD AND SELECT SCSI AND CLICK NEXT

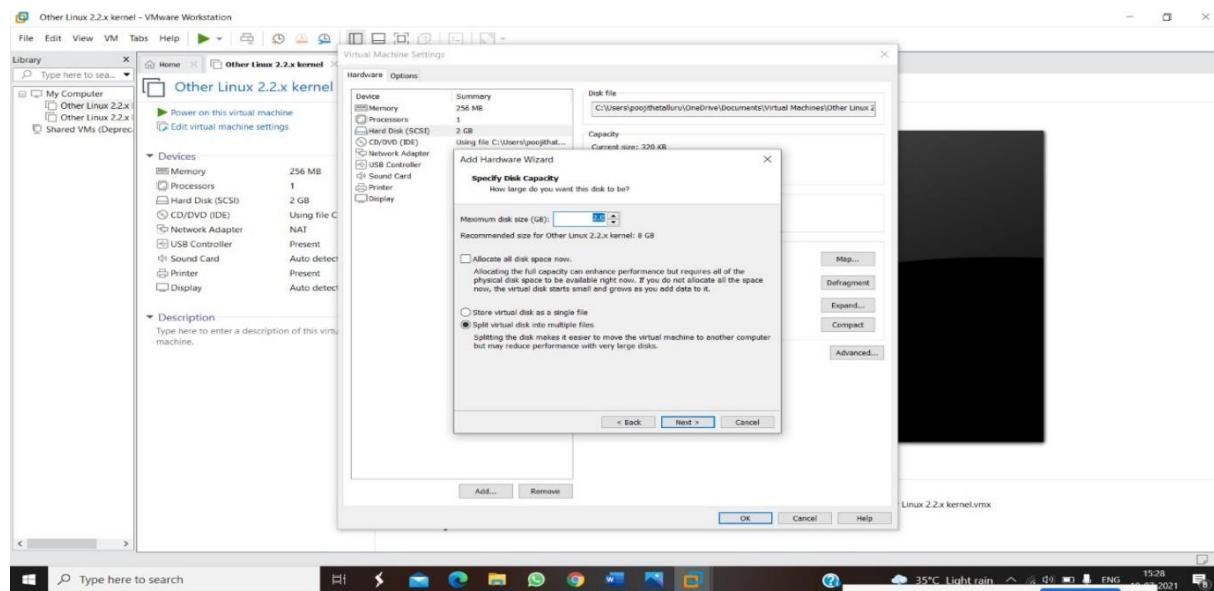


## STEP 4: CREATE NEW VIRTUAL DISK

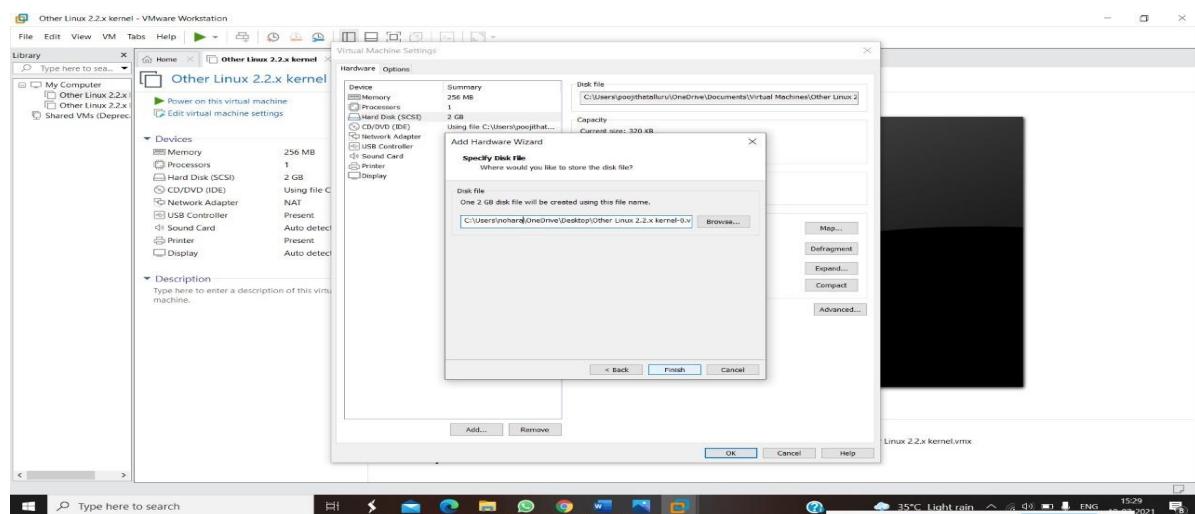


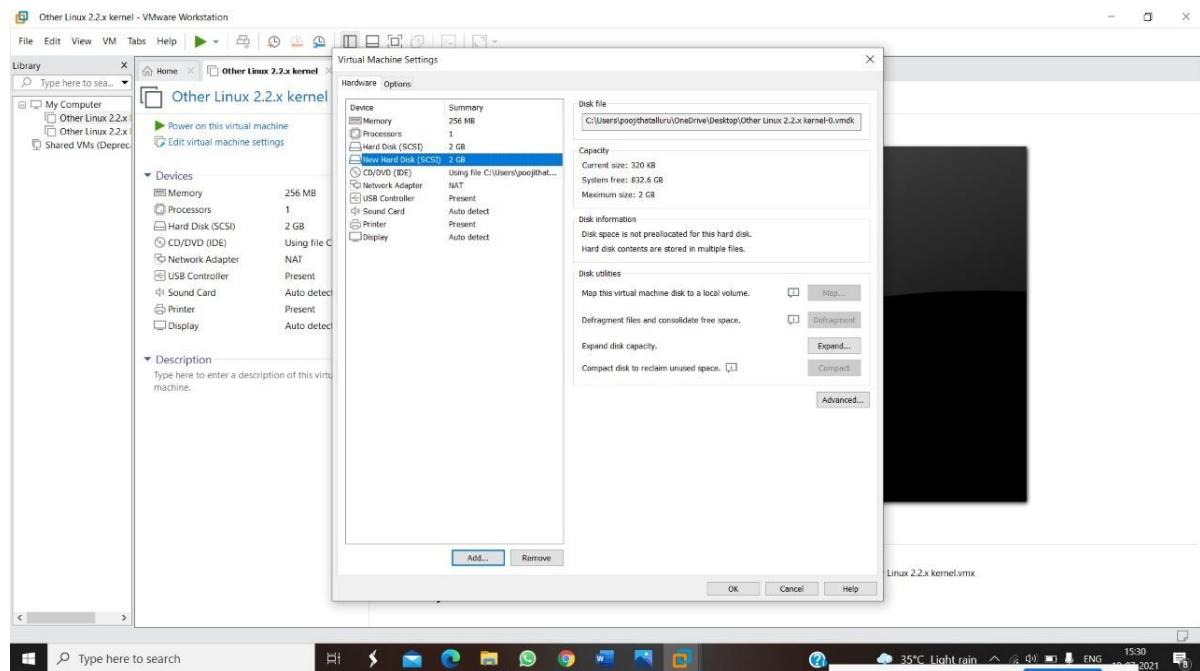


## STEP 5: SELECT THE DISK SIZE AS 2.0. AND SELECT SPLIT VIRTUAL DISK INTO MULTIFILES.



## STEP 6: GIVE NAME AND CLICK THE FINISH





## RESULT:

A virtual hard disk is created and storage is allocated for the virtual machine in VMware Workstation, ready for use with the specified guest operating system

## **EXPNO 4: CREATE A SNAPSHOT AND CLONING OF A VM AND TEST IT BY LOADING THE PREVIOUS VERSION/CLONED VM**

**DATE:**

**AIM:**

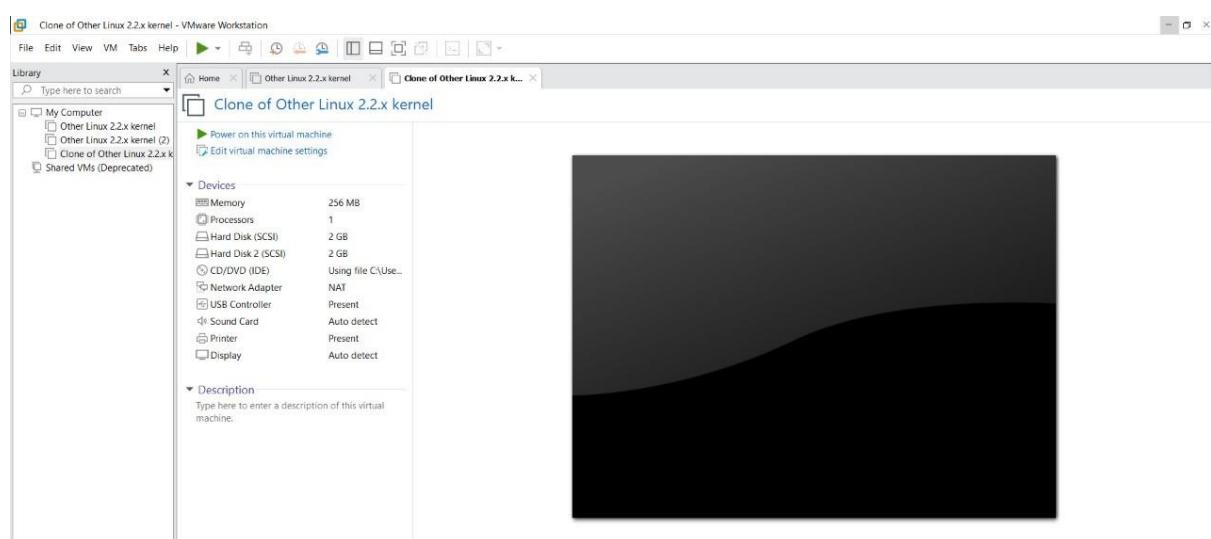
The aim is to create a snapshot and clone of a virtual machine (VM) and then test it by loading the previous version or the cloned VM

**PROCEDURE:**

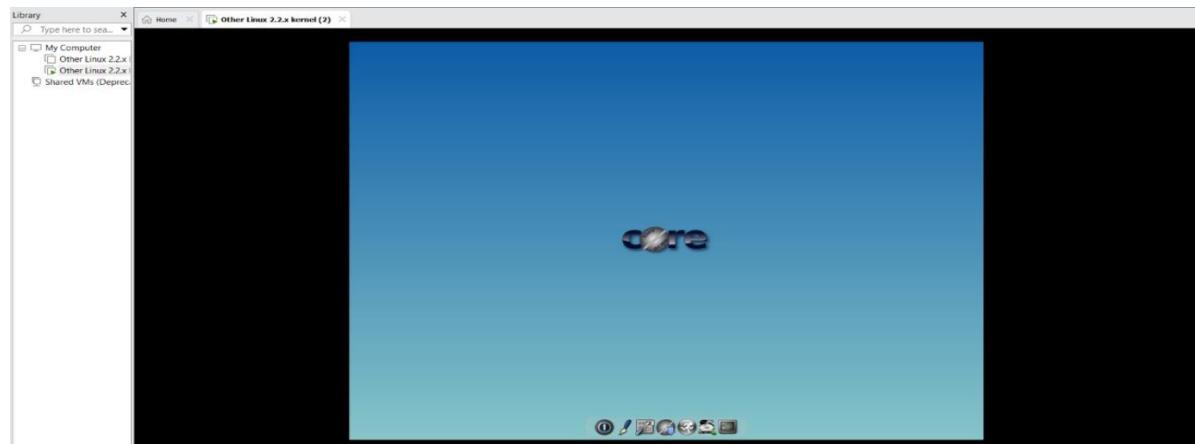
1. Create Snapshot: Take a snapshot of the VM's current state.
2. Clone VM: Create a clone of the VM using the snapshot.
3. Test Cloned VM: Load the cloned VM to ensure it functions as expected.
4. Test Previous Version: Optionally, load the VM from the snapshot to verify its state before changes.

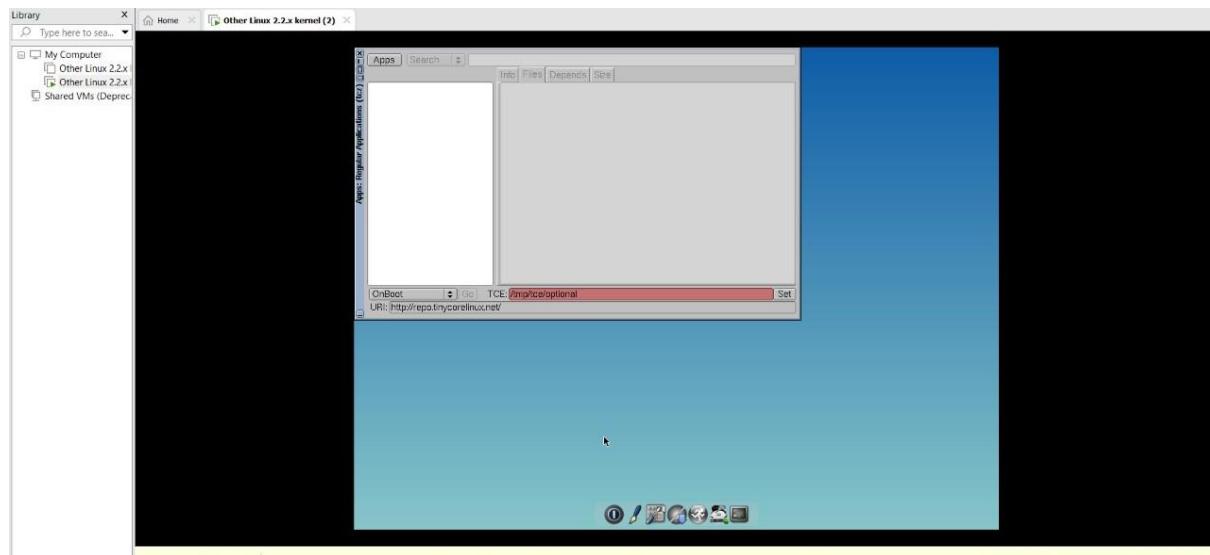
**IMPLEMENTATION:**

### **STEP 1: GOTO VMWARE WORKSTATION**

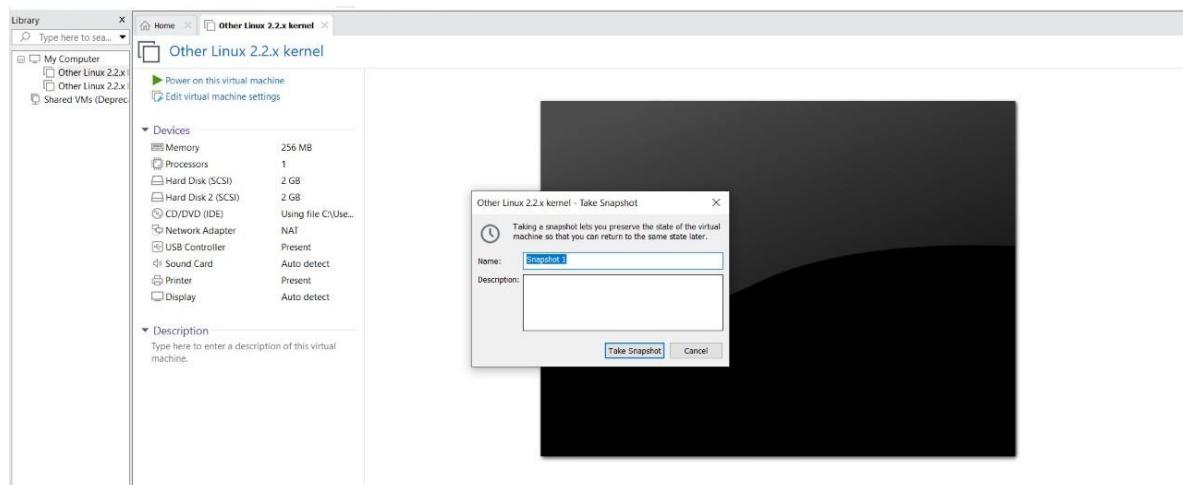
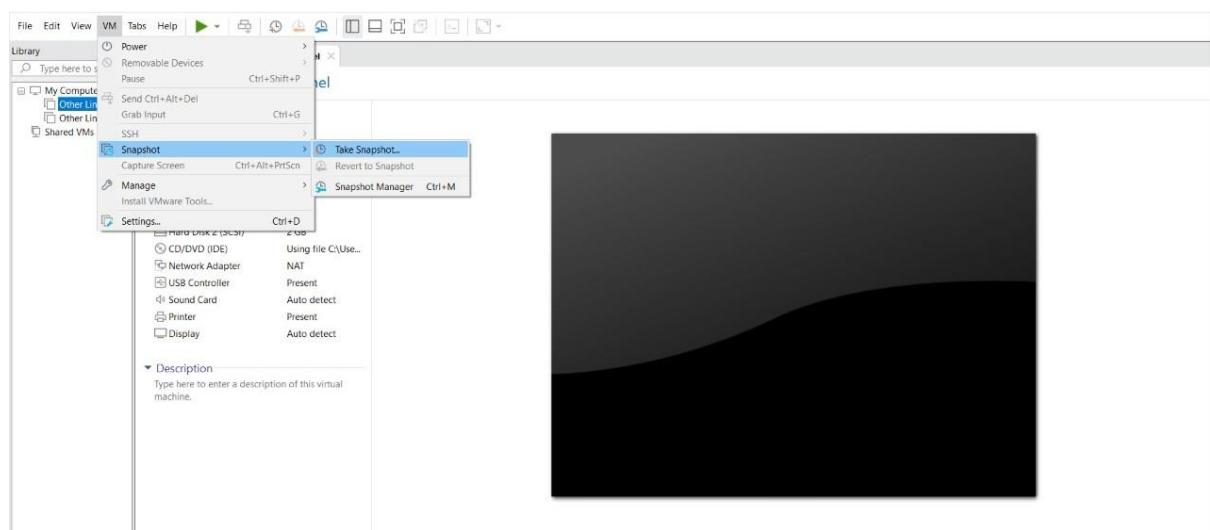


### **STEP 2: CREATE FILES ON DESKTOP**

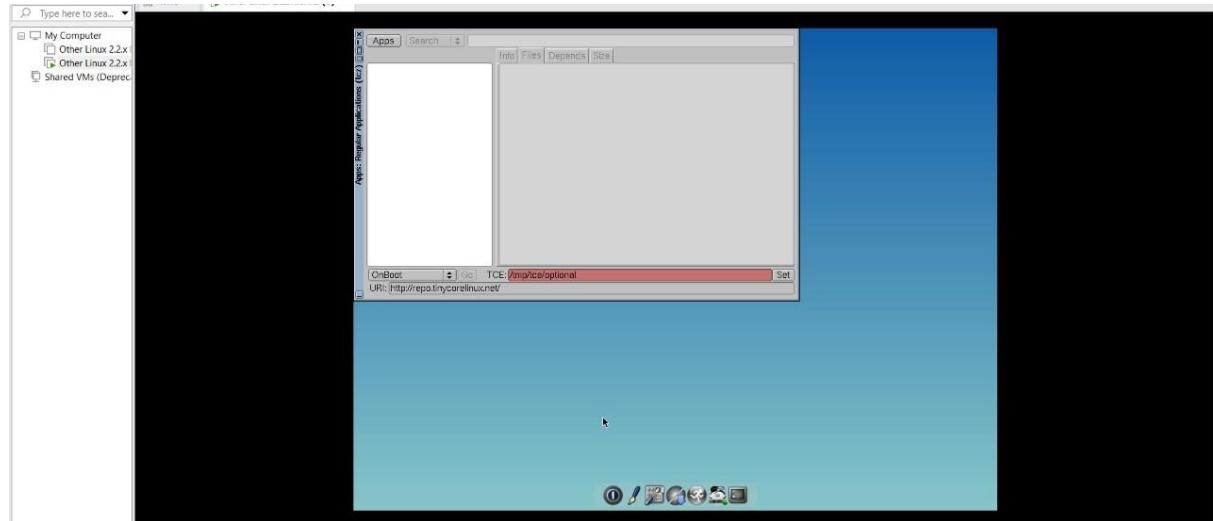




## STEP 2: CLICK ON VM AND SELECTS SNAPSHOT-> TAKE SNAPSHOT.

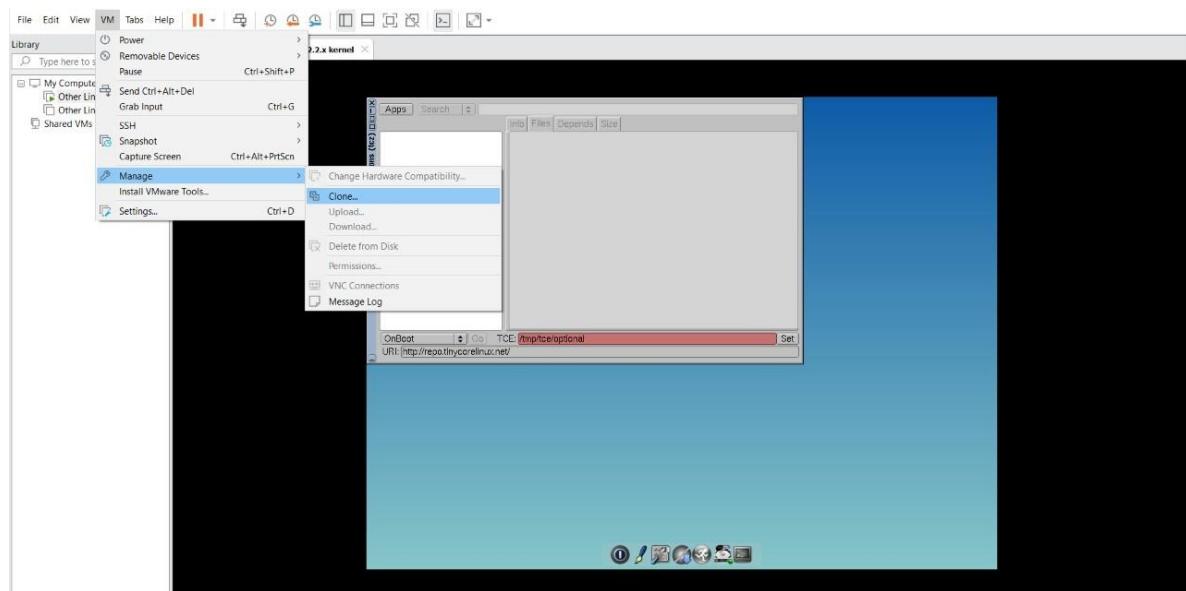


## STEP 4: SNAPSHOT IS BEING DONE

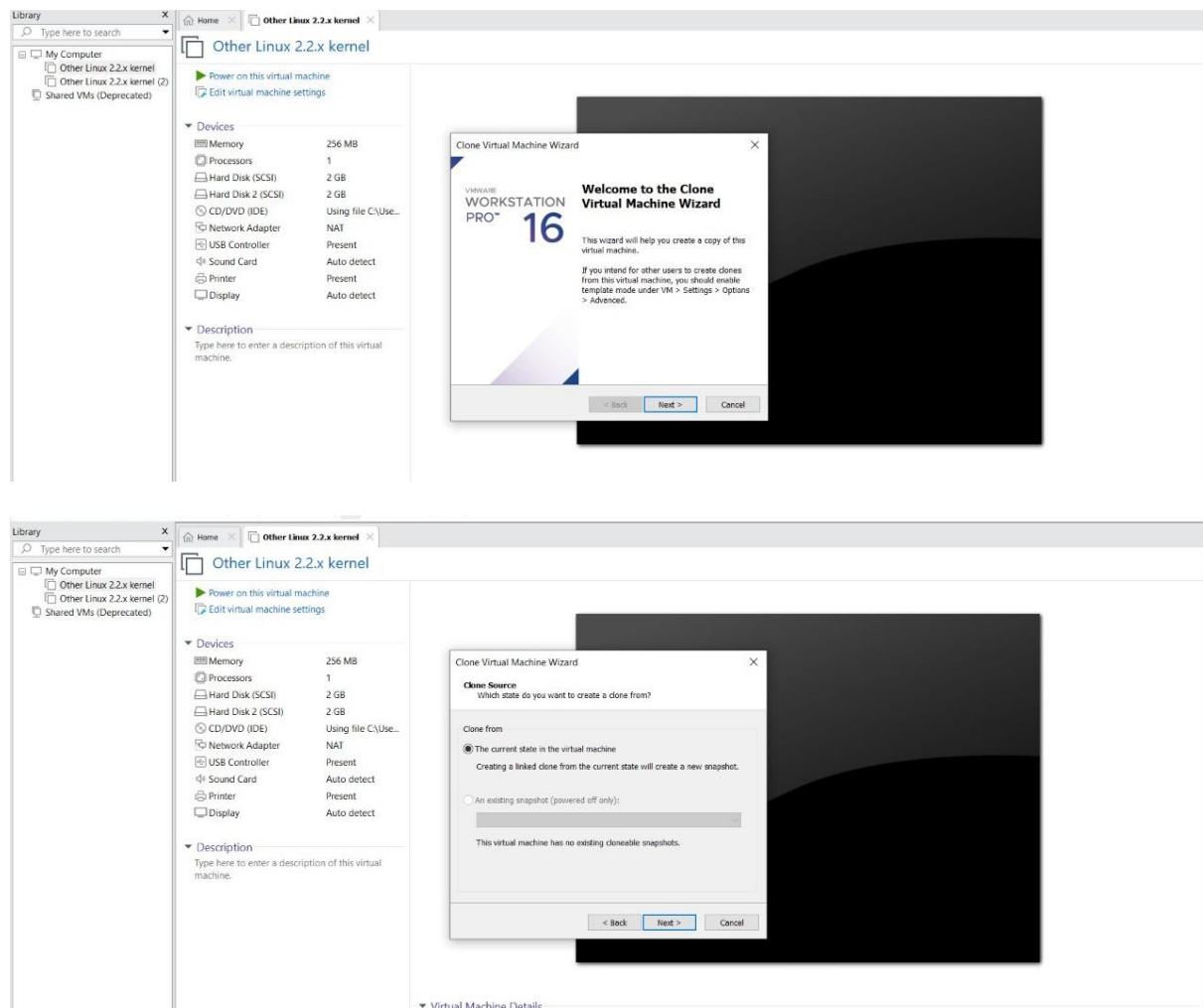


## CLONING OF A VM

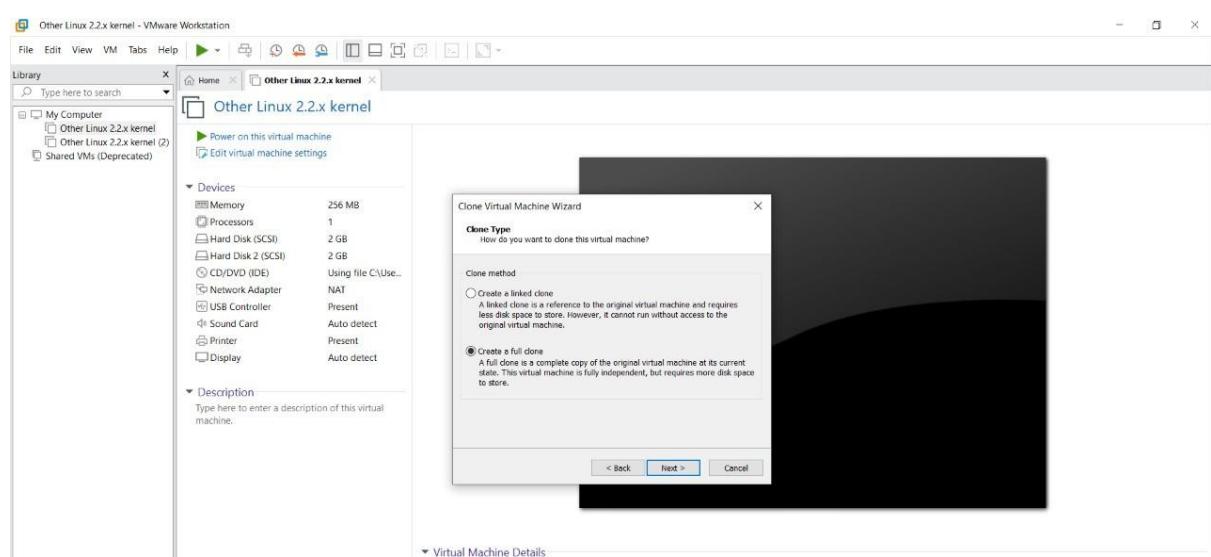
### STEP 1: GO TO VM AND GOTO MANAGE AND CLICK CLONE

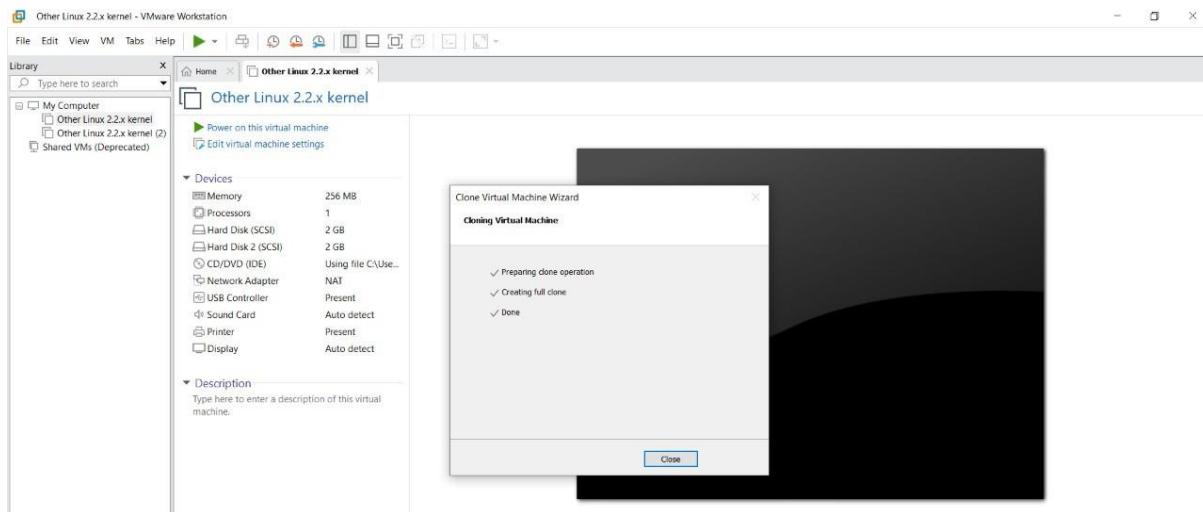


## STEP 2: CLICK CLONE

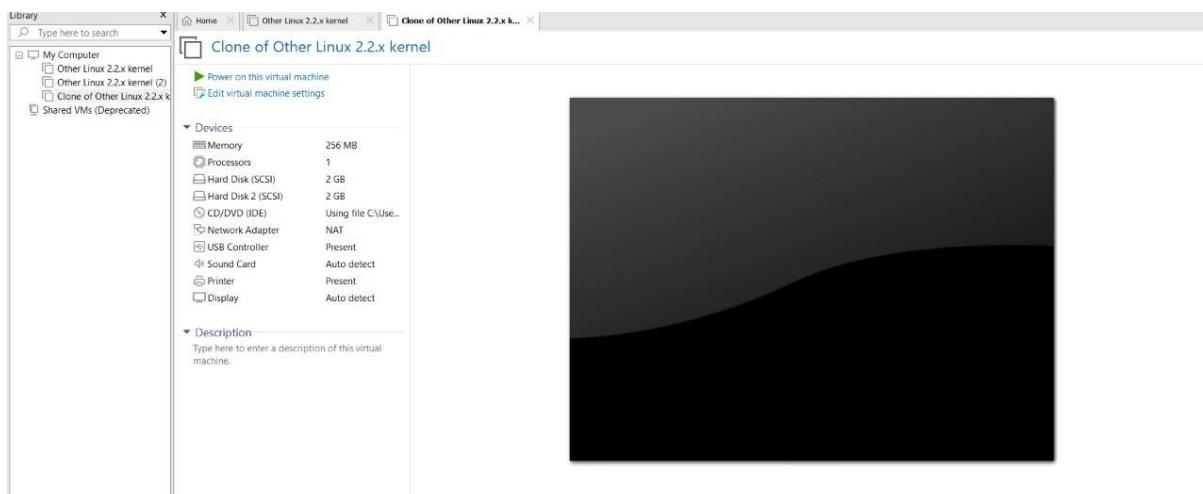


## STEP 3: SELECT THE FULL CLONE





#### STEP 4: AFTER CLONE AGAIN OR VM IS OPENED.



#### RESULT:

Successfully created a snapshot and clone of the VM.

Tested the cloned VM to confirm functionality.

Optionally tested the previous version loaded from the snapshot.

## **5. DEMONSTRATE INFRASTRUCTURE AS A SERVICE(IAAS) BY CREATING A VIRTUAL MACHINE USING A PUBLIC CLOUD SERVICE PROVIDER(AZURE/GCP/AWS) CONFIGURE WITH MINIMUM CPU,RAM AND STORAGE AND LAUNCH THE VM IMAGE.**

**AIM:** Demonstrate Infrastructure as a Service (IaaS) by creating a virtual machine (VM) with minimal CPU, RAM, and storage using a public cloud service provider (Azure/GCP/AWS) and launching the VM image.

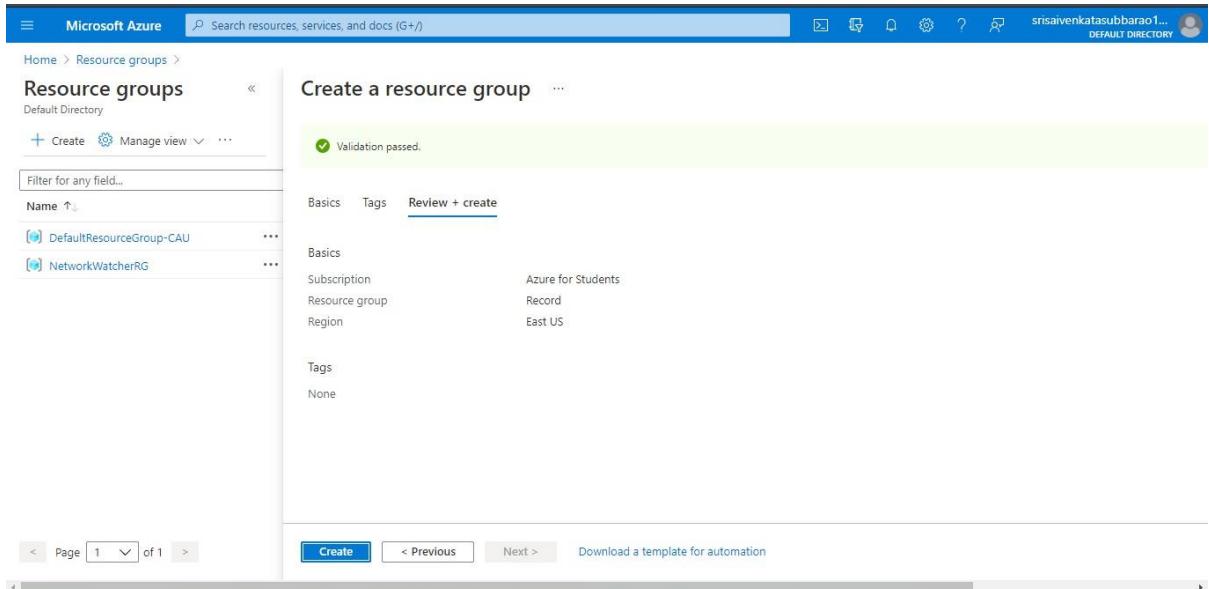
### **Procedure:**

- 1. Select Provider:** Choose a public cloud service provider (e.g., Azure, GCP, AWS).
- 2. Define Specifications:** Determine the minimum CPU, RAM, and storage requirements for the VM.
- 3. Create VM:** Access the cloud provider's dashboard and create a new VM instance.
- 4. Configure Specifications:** Set CPU, RAM, and storage to minimum values during VM creation.
- 5. Launch VM Image:** Choose the desired operating system image for the VM.
- 6. Start VM:** Initiate the VM instance to launch the virtual machine.

### **Implementation:**

#### **STEP1:CREATE AN ACCOUNT IN MICROSOFT AZURE.**

#### **STEP2: GOTO RESOURCE GROUP AND CREATE A RESOURCE GROUP.**



The screenshot shows the 'Create a resource group' wizard in the Microsoft Azure portal. The 'Validation passed.' message is displayed at the top. The 'Review + create' tab is selected. In the 'Basics' section, the subscription is set to 'Azure for Students', the resource group name is 'Record', and the region is 'East US'. Under the 'Tags' section, there are no tags assigned. At the bottom, there are buttons for 'Create', '< Previous', 'Next >', and 'Download a template for automation'.

## STEP3: GIVE NECESSARY THINGS FOR RESOURCE GROUP.

Microsoft Azure

Search resources, services, and docs (G+)

srisaivenkatasubbarao1...  
DEFAULT DIRECTORY

Home > Resource groups >

Resource groups

Create a resource group

Basics Tags Review + create

Resource group - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. Learn more

Project details

Subscription \* Azure for Students

Resource group \* Record

Resource details

Region \* (US) East US

Review + create < Previous Next : Tags >

Microsoft Azure

Search resources, services, and docs (G+)

srisaivenkatasubbarao1...  
DEFAULT DIRECTORY

Home > Resource groups >

Resource groups

Create a resource group

Basics Tags Review + create

Apply tags to your Azure resources to logically organize them by categories. A tag consists of a key (name) and a value. Tag names are case-insensitive and tag values are case-sensitive. Learn more

Name	Value	Resource
Resource group	Record	

Review + create < Previous Next : Review + create >

https://portal.azure.com/#

## STEP4: CREATE A VIRTUAL NETWORK FOR TO CREATE A VIRTUAL MACHINE .

The screenshot shows the Microsoft Azure portal's Resource groups page. At the top, there is a search bar and a user profile. Below the header, there are buttons for 'Create', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', 'Assign tags', and 'Feedback'. A filter bar allows filtering by 'Subscription' (set to 'all') and 'Location' (set to 'all'). The main table lists three resource groups:

Name	Subscription	Location	Actions
DefaultResourceGroup-CAU	Azure for Students	Australia Central	...
NetworkWatcherRG	Azure for Students	East US	...
Record	Azure for Students	East US	...

At the bottom, there are navigation links for '< Previous', 'Page 1 of 1', and 'Next >'.

## STEP5: NOW CREATE A VIRTUAL MACHINE WITH UR IP ADDRESS AN USERNAME AND PASSWORD FOR YOUR VIRTUAL MACINE.

## STEP6: AND YOUR VIRTUAL MACHINE IS DEPLOYED.

The screenshot shows the Microsoft Azure portal's 'CreateVm-MicrosoftWindowsServer.Overview' page. At the top, there is a search bar and a user profile. Below the header, there are buttons for 'Delete', 'Cancel', 'Redeploy', and 'Refresh'. The main area displays deployment details and next steps:

**Deployment details:**  
Deployment name: CreateVm-MicrosoftWindowsServer. Windows Server 2016  
Subscription: Azure for Students  
Resource group: Record  
Start time: 7/21/2021, 10:52:14 AM  
Correlation ID: a0f40b35-8270-49dc-bcf7-42eec66e5c61

**Next steps:**  
Setup auto-shutdown Recommended  
Monitor VM health, performance and network dependencies Recommended  
Run a script inside the virtual machine Recommended

At the bottom, there are two buttons: 'Go to resource' and 'Create another VM'.

## STEP7: NOW CONNECT THE VIRTUAL MACHINE AND DOWNLOAD THE RDP FILE TO OPEN YOUR WINDOWS VIRTUAL MACHINE.

**Essentials**

Resource group (change) : Record	Operating system : Windows (Windows Server 2019 Datacenter)
Status : Running	Size : Standard DS1 v2 (1 vcpus, 3.5 GiB memory)
Location : East US	Public IP address : 23.96.9.147
Subscription (change) : Azure for Students	Virtual network/subnet : Record-vnet/default
Subscription ID : db4eee0b-1e34-4be0-9c9c-65cc8d398405	DNS name : Not configured
Tags (change) : Click here to add tags	

**Properties**    Monitoring    Capabilities (8)    Recommendations    Tutorials

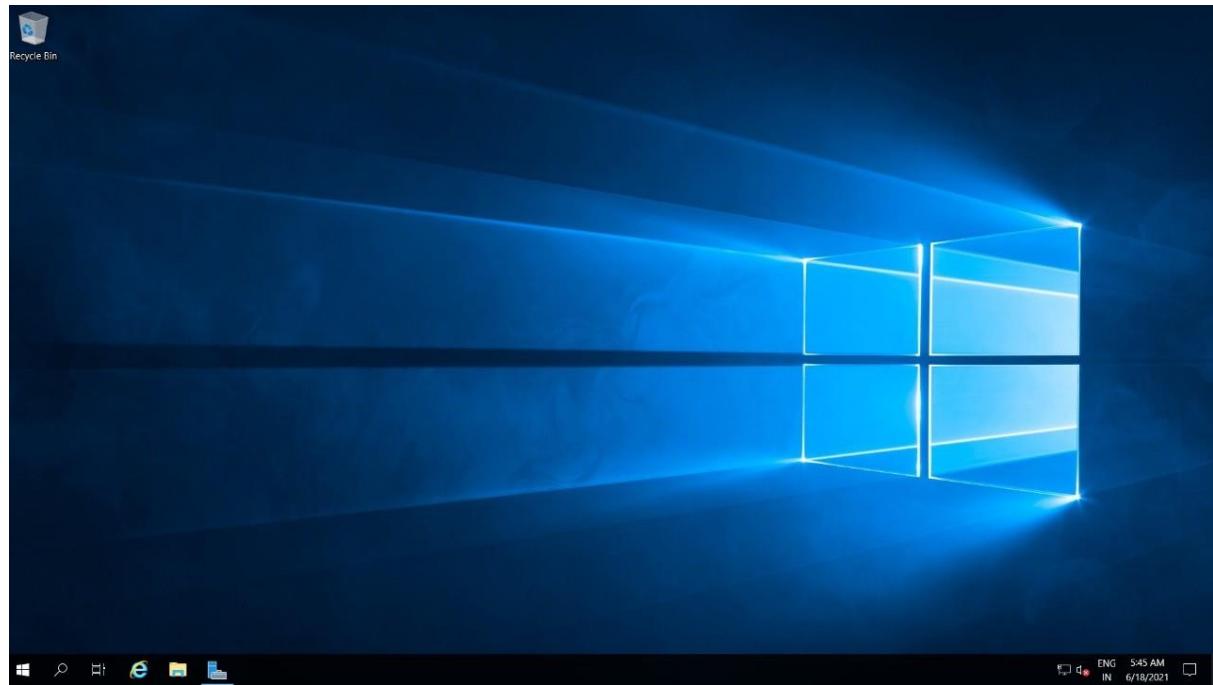
**Virtual machine**

Computer name	Record-virtual
Operating system	Windows (Windows Server 2019 Datacenter)
Publisher	MicrosoftWindowsServer
Offer	WindowsServer
Plan	2019-Datacenter
VM generation	V1
Agent status	Ready

**Networking**

Public IP address	23.96.9.147
Public IP address (IPv6)	-
Private IP address	10.0.0.4
Private IP address (IPv6)	-
Virtual network/subnet	Record-vnet/default
DNS name	Configure

## STEP8: CREATED A NEW WINDOWS VIRTUAL MACHINE.



## RESULT:

- A virtual machine with minimal CPU, RAM, and storage is successfully created and launched on the chosen public cloud service provider.
- This demonstrates the Infrastructure as a Service (IaaS) model, where users can provision and manage computing resources as needed without the hassle of maintaining physical hardware.

## **6. CREATE A SIMPLE WEB SITE USING ANY PUBLIC CLOUD SERVICE PROVIDER (AZURE/GCP/AWS) AND CHECK THE PUBLIC ACCESSIBILITY OF THE STORED FILE TO DEMONSTRATE STORAGE AS A SERVICE**

### **AIM:**

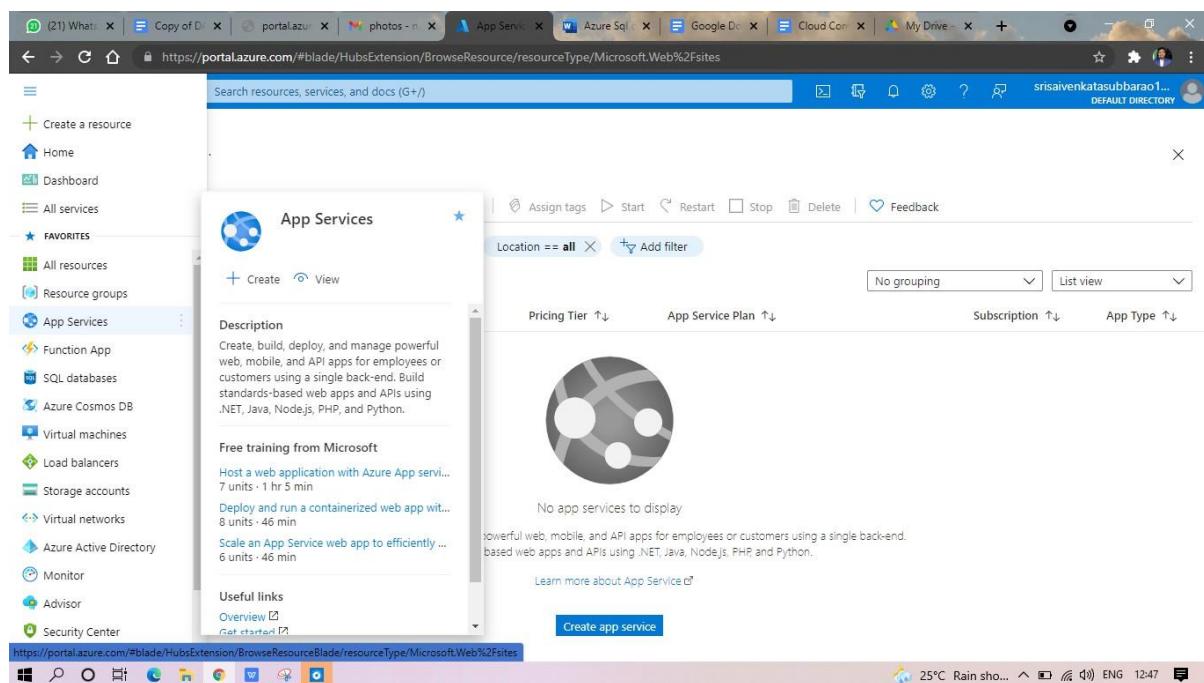
**Demonstrate Storage as a Service by creating a simple website using any public cloud service provider (Azure/GCP/AWS) and checking the public accessibility of stored files.**

### **Procedure:**

- 1. Select Provider:** Choose a public cloud service provider (Azure, GCP, AWS).
- 2. Create Website:** Use the provider's services to create a basic website.
- 3. Upload Files:** Upload files (e.g., HTML, CSS, images) to the storage service provided by the cloud provider.
- 4. Configure Access:** Ensure the storage service allows public access to the uploaded files.
- 5. Check Accessibility:** Access the website using a web browser to verify that the stored files are publicly accessible

### **IMPLEMENTATION:**

#### **STEP1: FIRSTLY GO TO APPSERVICE TO CREATE AN WEBAPP.**



## STEP2: ENTER THE RESOURCE GROUP AND WEBAPP NAME AND REGION AND SELECT THE LINUX OS.

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. [Learn more](#)

**Project Details**

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*  Resource Group \*

Instance Details

Name \*  .azurewebsites.net

Publish \*  Code  Docker Container

Runtime stack \*

Operating System \*  Linux  Windows

**Review + create** **< Previous** **Next : Deployment (Preview) >**

## STEP3: AFTER ENTER THE ALL THE NECESSARY THINGS CLICK THE REVIEW AND CREATE AND CLICK THE CREATE THE WEB APP.

Basics Deployment (Preview) Monitoring Tags **Review + create**

**Summary**

**Web App** by Microsoft **Basic (B1) sku**  
Estimated price - loading ...

**Details**

Subscription	db4eee0b-1e34-4be0-9c9c-65cc8d398405
Resource Group	Record
Name	Record-app
Publish	Code
Runtime stack	Node 14 LTS

**App Service Plan (New)**

Name	ASP-Record-92e3
Operating System	Linux
Region	Australia Central

**Create** **< Previous** **Next >** Download a template for automation

## STEP4: AND OUR DEPLOYMENT IS COMPLETED.

**Microsoft.Web-WebApp-Portal-1b6a401b-9ae6 | Overview**

Your deployment is complete

Deployment name: Microsoft.Web-WebApp-Portal-1b6a401b-9ae6 Start time: 7/21/2021, 12:49:54 PM  
Subscription: Azure for Students Correlation ID: 76653cd2-c090-4d97-a1e5-21033aa42efc  
Resource group: Record

**Deployment details** (Download)

**Next steps**

Manage deployments for your app. Recommended  
Protect your app with authentication. Recommended

**Go to resource**

**Security Center**  
Secure your apps and infrastructure  
[Go to Azure security center >](#)

**Free Microsoft tutorials**  
[Start learning today >](#)

**Work with an expert**  
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.  
[Find an Azure expert >](#)

## STEP5: GOTO WEBSITE URL LINK.

**Record-app** App Service

Resource group (change) : Record

	URL
Status	: Running
Location	: Australia Central
Subscription (change)	: Azure for Students
Subscription ID	: db4eee0b-1e34-4be0-9c9c-65cc8d398405
Tags (change)	: Click here to add tags

**JSON View**

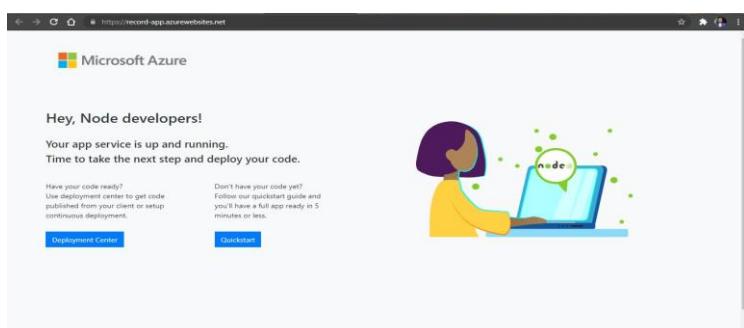
**Diagnose and solve problems**  
Our self-service diagnostic and troubleshooting experience helps you identify and resolve issues with your web app.

**Application Insights**  
Application Insights help you detect and diagnose quality issues in your apps, and helps you understand what your users actually do with it.

**App Service Advisor**  
App Service Advisor provides insights for improving app experience on the App Service platform. Recommendations are sorted by freshness, priority and impact to your app.

<https://portal.azure.com/>

## STEP6: THIS IS OUR WEBAPP SERVICE.



### Result:

- A simple website is created using the chosen public cloud service provider.
- Files are uploaded to the cloud storage service.
- The website is publicly accessible, demonstrating Storage as a Service, where users can store and access files over the internet without managing underlying infrastructure

## **7. CREATE A STORAGE SERVICE USING ANY PUBLIC CLOUD SERVICE PROVIDER (AZURE/GCP/AWS) AND CHECK THE PUBLIC ACCESSIBILITY OF THE STORED FILE TO DEMONSTRATE STORAGE AS A SERVICE.**

### **AIM:**

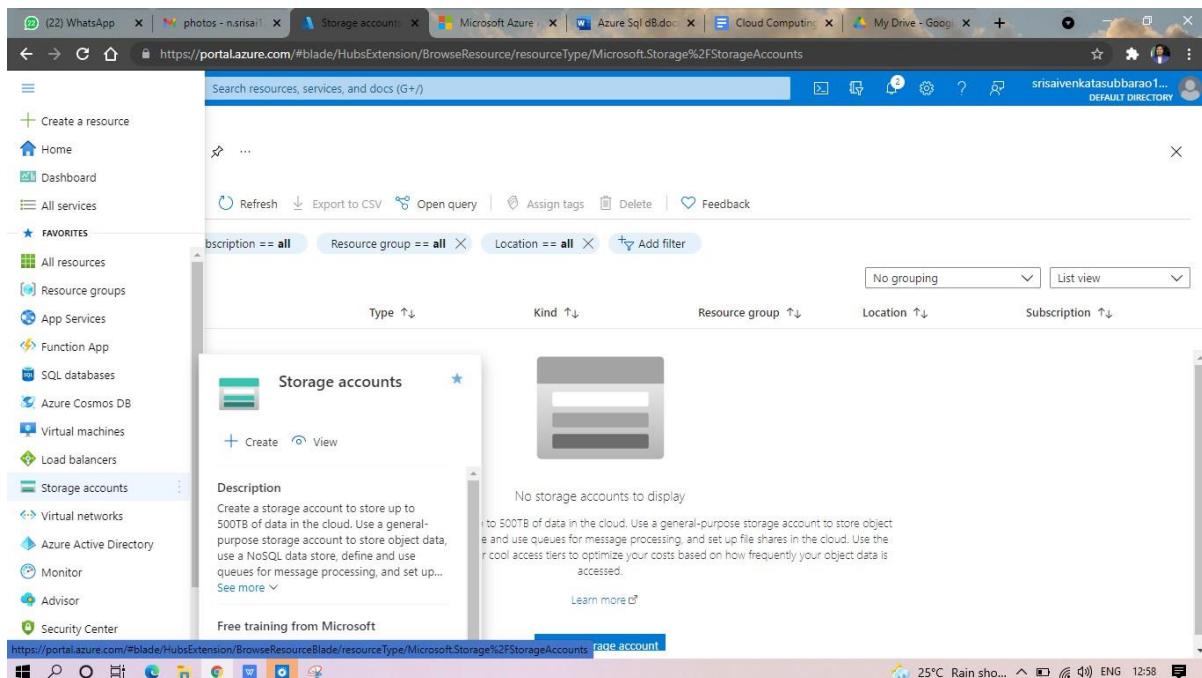
**Demonstrate Storage as a Service by creating a storage service using any public cloud service provider (Azure/GCP/AWS) and checking the public accessibility of stored files.**

### **PROCEDURE:**

- 1. Select Provider:** Choose a public cloud service provider (Azure, GCP, AWS).
- 2. Create Storage Service:** Utilize the provider's storage services to set up a storage container or bucket.
- 3. Upload File:** Upload a file to the created storage service.
- 4. Configure Access:** Adjust the permissions to allow public access to the uploaded file.
- 5. Check Accessibility:** Access the file using a web browser or a direct link to verify public accessibility.

### **IMPLEMENTATION:**

#### **STEP1: OPEN AZURE AND GOTO STORAGE ACCOUNTS AND CREATE STOROAGE ACCOUNT**



The screenshot shows the Microsoft Azure portal interface. The user is on the 'Storage accounts' page under the 'All services' category. The left sidebar includes links for Home, Dashboard, All resources, Resource groups, App Services, Function App, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, Monitor, Advisor, and Security Center. The main content area has a search bar at the top. Below it, there are filters for Subscription, Resource group, Location, and Type. A table lists a single storage account entry. The table columns are Type (General-Purpose), Kind (blob), Resource group (all), Location (East Asia), and Subscription (all). There is also a 'Create' button and a 'View' button. The bottom of the page has a 'Description' section with information about storage accounts and a 'Learn more' link. The URL in the address bar is https://portal.azure.com/#blade/HubsExtension/BrowseResourceBlade/resourceType/Microsoft.Storage%2FStorageAccounts/createAccount.

#### **STEP2: ENTER THE RESOURC GROUP AND AND STORAGE ACCOUNT NAME AND REVIEW AND CREATE AND CLICK TH CREATE AND YOUR STORAGE ACCOUNT WILL BE DEPLOYED SUCESSFULLY.**

The screenshot shows the Microsoft Azure Deployment Overview page for a deployment named 'recordstoragesubbarao\_1626852653220'. The status bar at the top right indicates 'Deployment succeeded' and 'Deployment 'recordstoragesubbarao\_1626852653220' to resource group 'Record' was successful.' The main content area displays deployment details: Deployment name: recordstoragesubbarao\_1626852653220, Start time: 7/21/2021, 1:00:57 PM, Subscription: Azure for Students, Resource group: Record. Below this, there are sections for 'Deployment details' (with a download link) and 'Next steps' (with a 'Go to resource' button). To the right, there are links to 'Security Center', 'Free Microsoft tutorials', and 'Work with an expert'. The URL in the browser bar is https://portal.azure.com/#.

## STP3: AND OUR STORAGE ACCOUNT IS CREATED.

The screenshot shows the Microsoft Azure Storage account overview page for 'recordstoragesubbarao'. The left sidebar lists various storage services: Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage Explorer (preview), Data storage (Containers, File shares, Queues, Tables), Security + networking, and Networking. The main content area displays the storage account's properties: Resource group (change) : Record, Location : East US, Primary/Secondary Location : Primary: East US, Secondary: West US, Subscription (change) : Azure for Students, Subscription ID : db4eee0b-1e34-4be0-9c9c-65cc8d398405, Disk state : Primary: Available, Secondary: Available. It also shows the Blob service configuration (Hierarchical namespace: Disabled, Default access tier: Hot, Blob public access: Enabled, Blob soft delete: Enabled (7 days)) and Security settings (Require secure transfer for REST API operations: Enabled, Storage account key access: Enabled, Minimum TLS version: Version 1.2). The URL in the browser bar is https://portal.azure.com/#.

## STEP5: GOTO STATIC WEBSITE

This screenshot shows the Microsoft Azure Storage account details page for a storage account named 'deeksha'. The left sidebar contains navigation links for Home, deeksha (Storage account), Encryption, Security, Data management (Geo-replication, Data protection, Object replication, Blob inventory (preview), Static website, Lifecycle management, Azure search), Settings (Configuration, Resource sharing (CORS), Advisor recommendations, Endpoints, Locks), and Monitoring. The main content area displays the following details:

Essentials	
Resource group (change)	: Gopi
Location	: East US
Primary/Secondary Location	: Primary: East US, Secondary: West US
Subscription (change)	: Azure for Students
Subscription ID	: 88bd0e11-e431-4a2a-8040-bdf7d22901aa
Disk state	: Primary: Available, Secondary: Available
Tags (change) :	

Below the Essentials section are tabs for Properties, Monitoring, Capabilities (?), Recommendations, Tutorials, and Developer Tools. Under Properties, there are two sections: Blob service and Security.

Blob service	
Hierarchical namespace	Disabled
Default access tier	Hot
Blob public access	Enabled
Blob soft delete	Enabled (7 days)
Container soft delete	Enabled (7 days)
Versioning	Disabled

Security	
Require secure transfer for REST API operations	Enabled
Storage account key access	Enabled
Minimum TLS version	Version 1.2
Infrastructure encryption	Disabled

The status bar at the bottom shows system information: 32°C, ENG, 10:20, 22-06-2021.

## STEP6: AND ENABLE AND ENTER YOUR INDEX AND ERROR HTML FILENAMES.

This screenshot shows the Microsoft Azure Static website configuration page for the 'deeksha' storage account. The left sidebar contains the same navigation links as the previous screenshot. The main content area shows the configuration for a static website:

Static website status: Enabled

Primary endpoint: https://deeksha.z13.web.core.windows.net/

Secondary endpoint: https://deeksha-secondary.z13.web.core.windows.net/

Index document name: index.html

Error document path: 404.html

The status bar at the bottom shows system information: 33°C, ENG, 10:43, 22-06-2021.

## STEP7: AND GOTO STORAGE EXPLORER(Review) AND AND GOTO BLOB CONTAINERS AND WEB AND UPLOAD THE TWO HTML FILES INIT

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, there's a sidebar with various options like Data migration, Events, Storage Explorer (preview), Data storage (Containers, File shares, Queues, Tables), Security + networking (Networking, Azure CDN, Access keys, Shared access signature, Encryption), and a link to the portal. The main area shows a tree view with 'BLOB CONTAINERS' expanded, and '\$web' selected. Under '\$web', there are 'FILE SHARES', 'QUEUES', and 'TABLES'. A table lists 11 files in the '\$web' container:

NAME	ACCESS TIER	ACCESS TIER LAST MODIFIED	LAST MODIFIED	BLOB TYPE	CONTENT TYPE	SIZE	STATUS	RI
about.jpg	Hot (inferred)		7/21/2021, 1:30:36 PM	Block Blob	image/jpeg	188.6 KB	Active	
hero-bg.png	Hot (inferred)		7/21/2021, 1:30:35 PM	Block Blob	image/png	7.1 KB	Active	
hero-img.png	Hot (inferred)		7/21/2021, 1:30:35 PM	Block Blob	image/png	22.1 KB	Active	
index.html	Hot (inferred)		7/21/2021, 1:30:34 PM	Block Blob	text/html	9.6 KB	Active	
logo.png	Hot (inferred)		7/21/2021, 1:30:34 PM	Block Blob	image/png	1.0 KB	Active	
README.md	Hot (inferred)		7/21/2021, 1:30:34 PM	Block Blob	application/octet-stream	14 B	Active	
script.js	Hot (inferred)		7/21/2021, 1:30:34 PM	Block Blob	text/javascript	447 B	Active	
style.css	Hot (inferred)		7/21/2021, 1:30:34 PM	Block Blob	text/css	3.8 KB	Active	
values-1.png	Hot (inferred)		7/21/2021, 1:30:35 PM	Block Blob	image/png	203 KB	Active	
values-2.png	Hot (inferred)		7/21/2021, 1:30:35 PM	Block Blob	image/png	22.2 KB	Active	
values-3.png	Hot (inferred)		7/21/2021, 1:30:35 PM	Block Blob	image/png	19.6 KB	Active	

Showing 1 to 11 of 11 cached items

## STEP8: AND AGAIN RETURN TO STATIC WEBSITE AND OPEN THE PRIMARY LINK AND YOUR WEB PAGE IS CREATED

The screenshot shows a static website page titled 'React'. The URL in the address bar is <https://recordstoragesubbarao.z13.web.core.windows.net>. The page has a header with links to Home, About, Services, Testimonials, and Contact. The main content features a large blue heading: 'We offer modern solutions for growing your business'. Below it is a subtext: 'We are team of talented designers making websites with Bootstrap'. At the bottom is a blue 'Get Started' button. To the right of the text is a large, stylized 3D illustration of three people (two men and one woman) working together on a massive smartphone screen. The screen displays some code and a small logo. The background of the page is white.

### RESULT:

- A storage service is set up using the chosen public cloud service provider.
- A file is uploaded to the storage service.
- The file is publicly accessible, showcasing Storage as a Service, where users can store and retrieve data over the internet without managing underlying infrastructure.

## **8. CREATE A SQL STORAGE SERVICE AND PERFORM A BASIC QUERY USING ANY PUBLIC CLOUD SERVICE PROVIDER (AZURE/GCP/AWS) TO DEMONSTRATE DATABASE AS A SERVICE (DAAS)**

### **AIM:**

Demonstrate Database as a Service (DBaaS) by creating a SQL storage service and performing a basic query using any public cloud service provider (Azure/GCP/AWS).

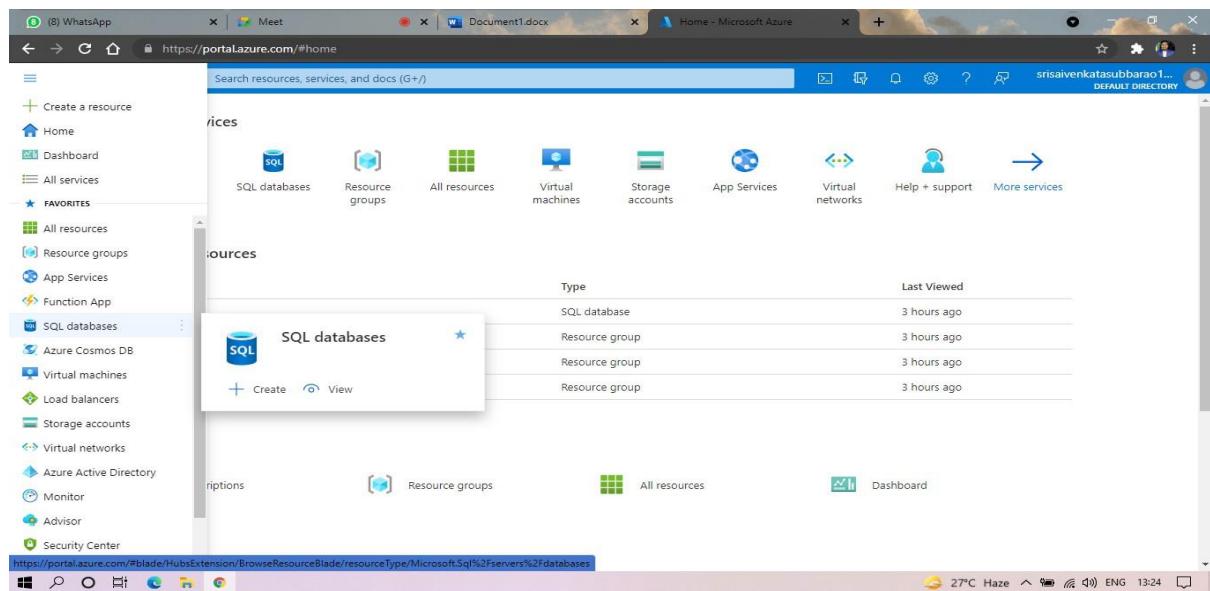
### **PROCEDURE:**

1. Select Provider: Choose a public cloud service provider (Azure, GCP, AWS).
2. Create SQL Storage Service: Utilize the provider's database services to create a SQL database instance.
3. Define Schema: Define a basic database schema with tables and columns.
4. Insert Data: Insert some sample data into the database tables.
5. Perform Query: Write and execute a basic SQL query (e.g., SELECT statement) to retrieve data from the database.

### **IMPLEMENTATION:**

#### **STEP1: GOTO AZURE AND GOTO SQLDATABASE.**

#### **STEP 02:- Now Create a Sql Databse**

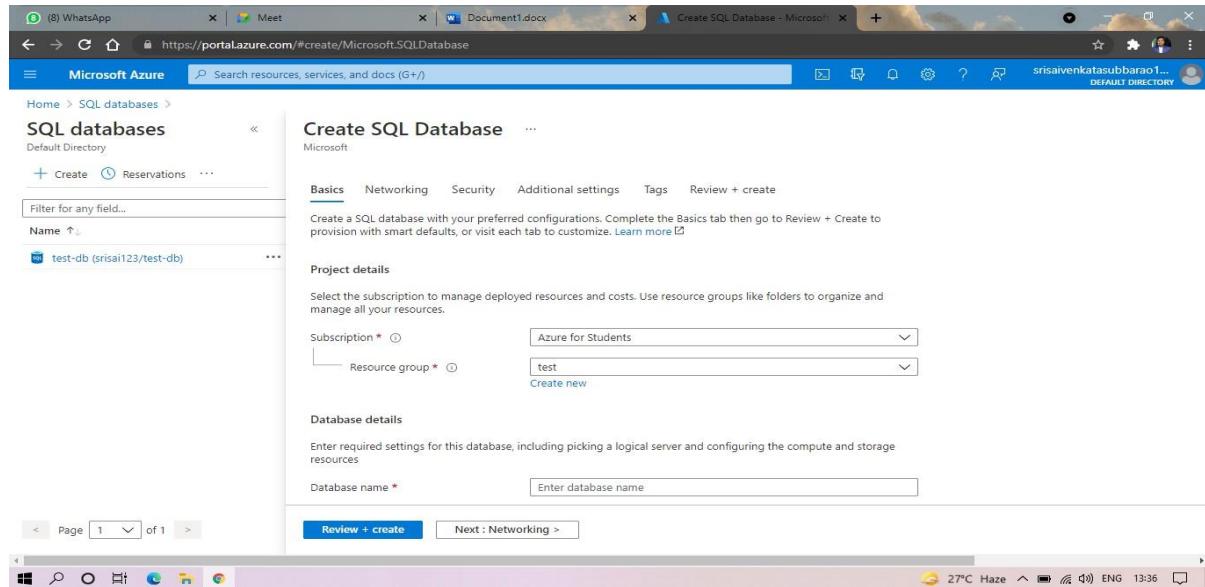


The screenshot shows the Microsoft Azure portal interface. The left sidebar is collapsed, and the main navigation bar includes 'WhatsApp', 'Meet', 'Document1.docx', and 'Home - Microsoft Azure'. The search bar at the top says 'Search resources, services, and docs (G+/-)'. The main content area has a blue header 'Services' with icons for 'SQL databases', 'Resource groups', 'All resources', 'Virtual machines', 'Storage accounts', 'App Services', 'Virtual networks', 'Help + support', and 'More services'. Below this is another header 'Sources' with a table:

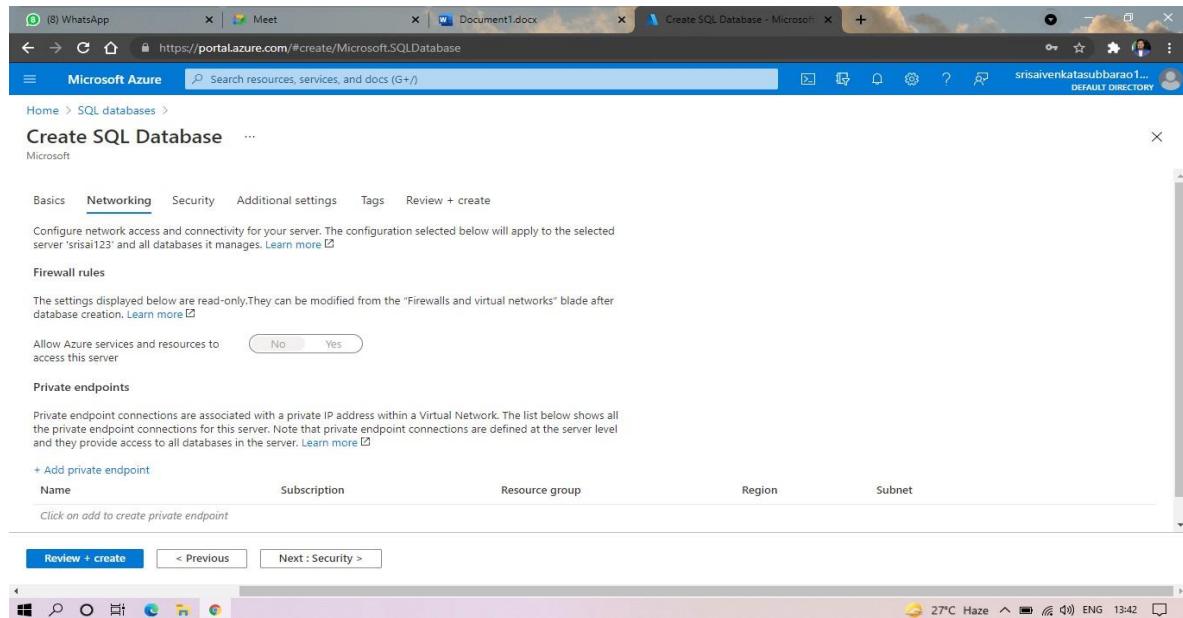
Type	Last Viewed
SQL database	3 hours ago
Resource group	3 hours ago
Resource group	3 hours ago
Resource group	3 hours ago

The 'SQL databases' row is highlighted with a blue star icon. At the bottom of this table are 'Create' and 'View' buttons. The footer of the page shows 'https://portal.azure.com/#blade/HubsExtension/BrowseResourceBlade/resourceType/Microsoft.Sql%2Fservers%2Fdatabases' and system status indicators like '27°C Haze' and 'ENG 13:24'.

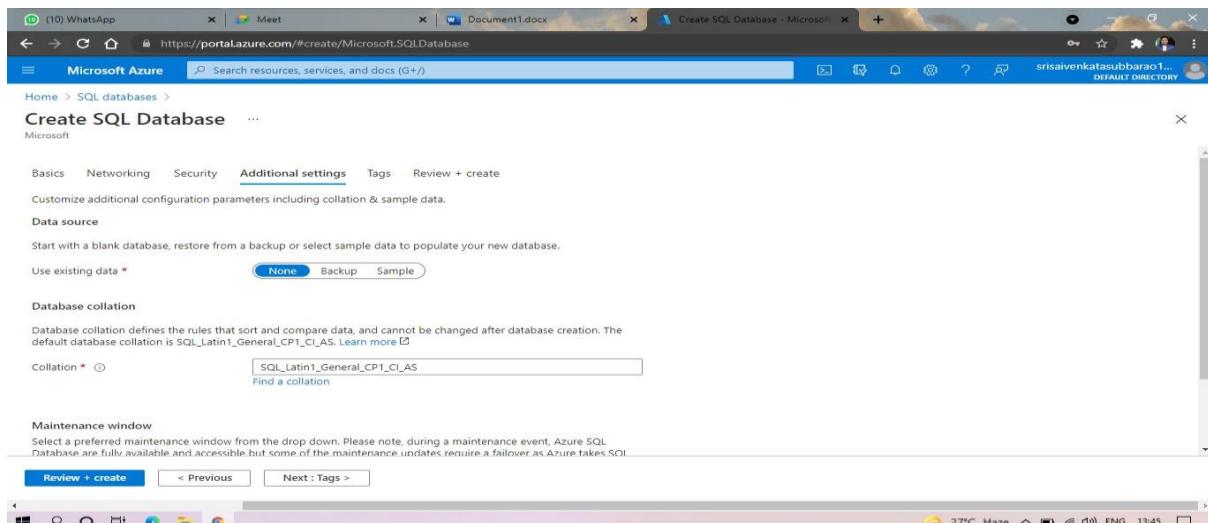
### STEP3: SELECT THE RESOURCE GROUP AND ENTER THE SERVERNAME THAT APPLICABLE.



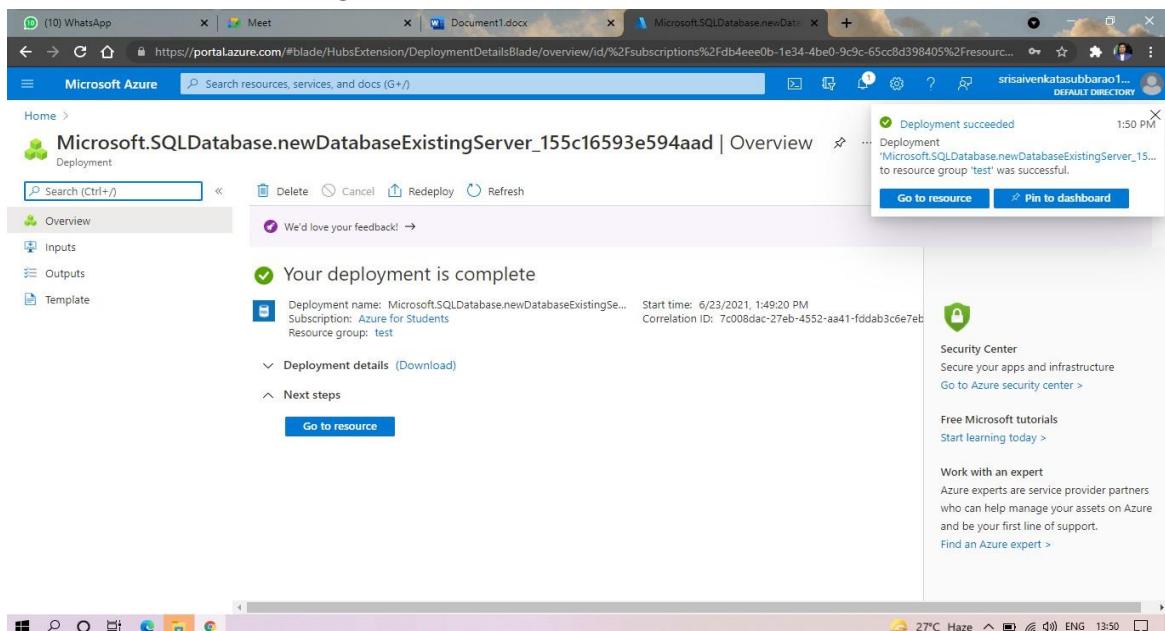
### STEP4: IN NETWORKING SELECT ALLOW AZURE SERVICES AND RESOURCES TO ACCESS THIS SERVER.



### STEP5: IN ADDITIONAL SETTINGS SELECT SAMPLE.



## STEP6:AND THE SQL DATABASE IS DEPLOYED.



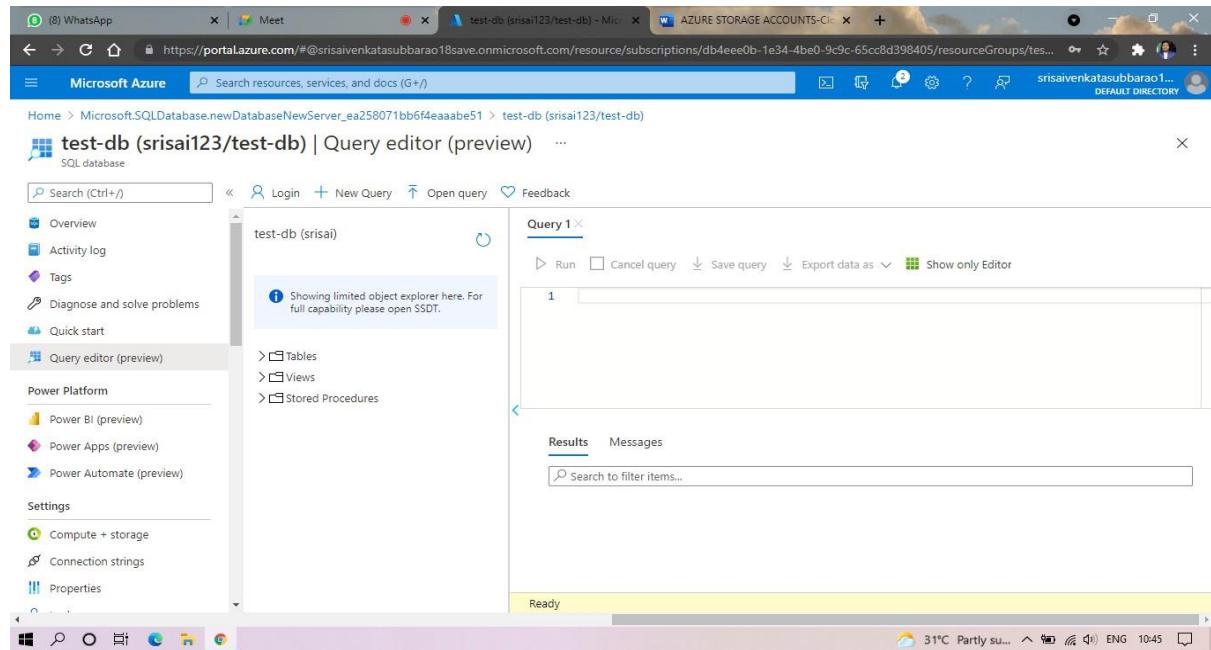
## STEP7:AND NOW GOTO QUERY EDITOR.

The screenshot shows the Microsoft Azure portal interface. The left sidebar navigation includes 'Overview', 'Activity log', 'Tags', 'Diagnose and solve problems', 'Quick start', and 'Query editor (preview)'. Under 'Power Platform', there are links for 'Power BI (preview)', 'Power Apps (preview)', and 'Power Automate (preview)'. Under 'Settings', there are links for 'Compute + storage', 'Connection strings', 'Properties', and 'Locks'. The main content area displays 'Essentials' information for the database, such as Resource group (change) : test, Status : Paused, Location : West US 3, Subscription (change) : Azure for Students, Subscription ID : db4eee0b-1e34-4be0-9c9c-65cc8d398405, Tags (change) : Click here to add tags, Server name : srисai123.database.windows.net, Connection strings : Show database connection strings, Pricing tier : General Purpose: Serverless, Gen5, 1 vCore, Auto-pause delay : 1 hour, and Earliest restore point : 2021-06-23 05:19 UTC. Below this is a 'Compute utilization' chart. A search bar at the top left says 'Search (Ctrl+)' and a top navigation bar has tabs for 'Home', 'Meet', 'Document1.docx', and 'test-db (srисai123/test-db) - Microsoft Edge'.

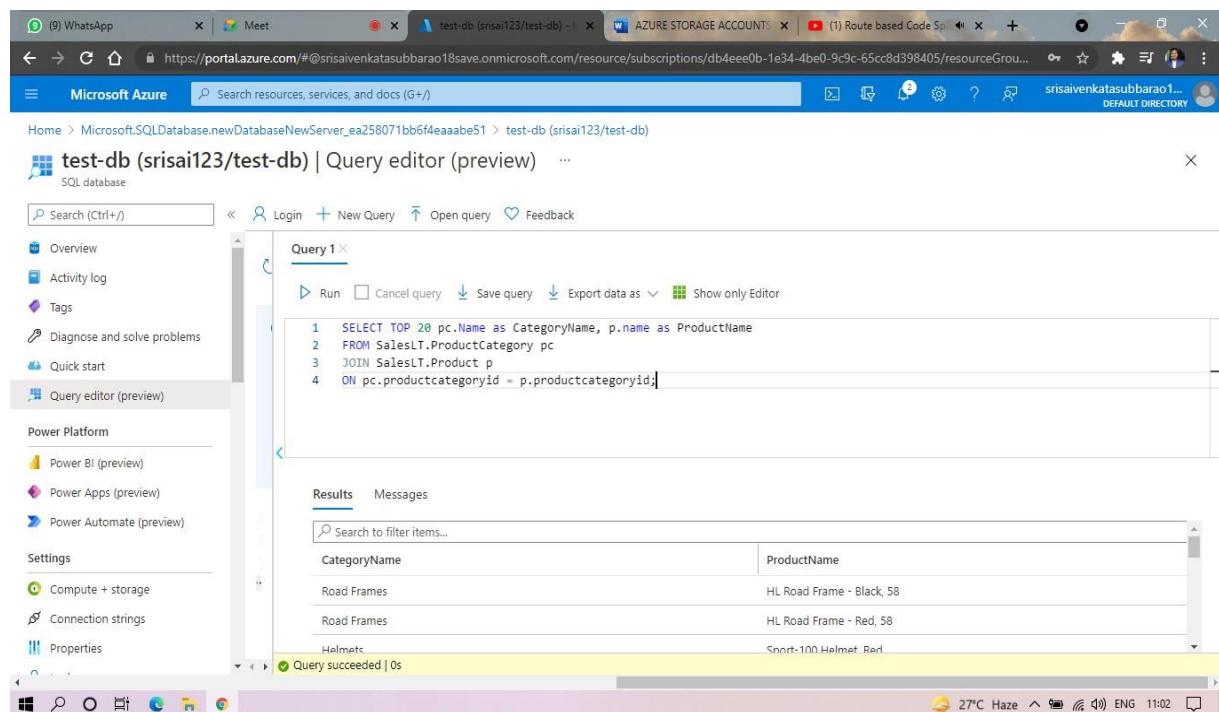
## STEP8:AND NOW AGAIN LOGIN TO THE SQLDATADBATABASE

The screenshot shows the Microsoft Azure portal interface, specifically the 'Query editor (preview)' section for the test-db. The left sidebar navigation is identical to the previous screenshot. The main content area shows the 'Welcome to SQL Database Query Editor' screen. It features two authentication options: 'SQL server authentication' (Login: srисai, Password: [redacted], OK button) and 'Active Directory authentication' (Continue as srисaivenkatasubbarao10@...). A large 'SQL' icon is centered above the authentication fields. The top navigation bar includes tabs for 'Home', 'Meet', 'Document1.docx', and 'test-db (srисai123/test-db) - Microsoft Edge'.

## STEP: AND OUR TABLES WILL SHOWN AND TYPE THE QUERY TO EXECUTED.



## STEP10: AND OUR OUTPUT IS READY.



## RESULT:

- A SQL storage service is created using the chosen public cloud service provider.
- A basic database schema is defined and populated with sample data.
- A SQL query is successfully executed, demonstrating Database as a Service, where users can perform database operations without managing the underlying infrastructure.

## **EXP. 9: PERFORM THE BASIC CONFIGURATION SETUP FOR INSTALLING HADOOP 2.X LIKE CREATING THE HDUSER AND SSH LOCALHOST**

### **AIM:**

**Perform basic configuration setup for installing Hadoop 2.x, including creating the HDUSER and configuring SSH localhost.**

### **PROCEDURE:**

#### **Step 1 – System Update**

```
$ sudo apt-get update
```

#### **Step 2 – Install Java and Set JAVA\_HOME**

//This first thing to do is to setup the webupd8 ppa on your system. Run the following command and proceed.

```
$ sudo apt-add-repository ppa:webupd8team/java
```

```
$ sudo apt-get update
```

//After setting up the ppa repository, update the package cache as well.

//Install the Java 8 installer

```
$ sudo apt-get install oracle-java8-installer
```

// After the installation is finished, Oracle Java is setup. Run the java command again to check the version and vendor.

[or]

```
$ sudo apt-get install default-jdk
```

```
$ java -version
```

#### **Step 3 – Add a dedicated Hadoop user**

```
$ sudo addgroup hadoop
```

```
$ sudo adduser --ingroup hadoop hduser
```

// Add hduser to sudo user group

```
$ sudo adduser hduser sudo
```

#### **Step 4 – Install SSH and Create Certificates**

```
$ sudo apt-get install ssh
```

```
$ su hduser
```

```
$ ssh-keygen -t rsa -P ""  
  
// Set Environmental variables  
$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

**Step 5 – Check if SSH works**

```
$ ssh localhost
```

**Step 6 – Install Hadoop**

```
// Extract Hadoop-2.7.2  
$ sudo tar xvzf hadoop-2.7.2.tar.gz
```

```
// Create a folder ‘hadoop’ in /usr/local  
$ sudo mkdir -p /usr/local/hadoop
```

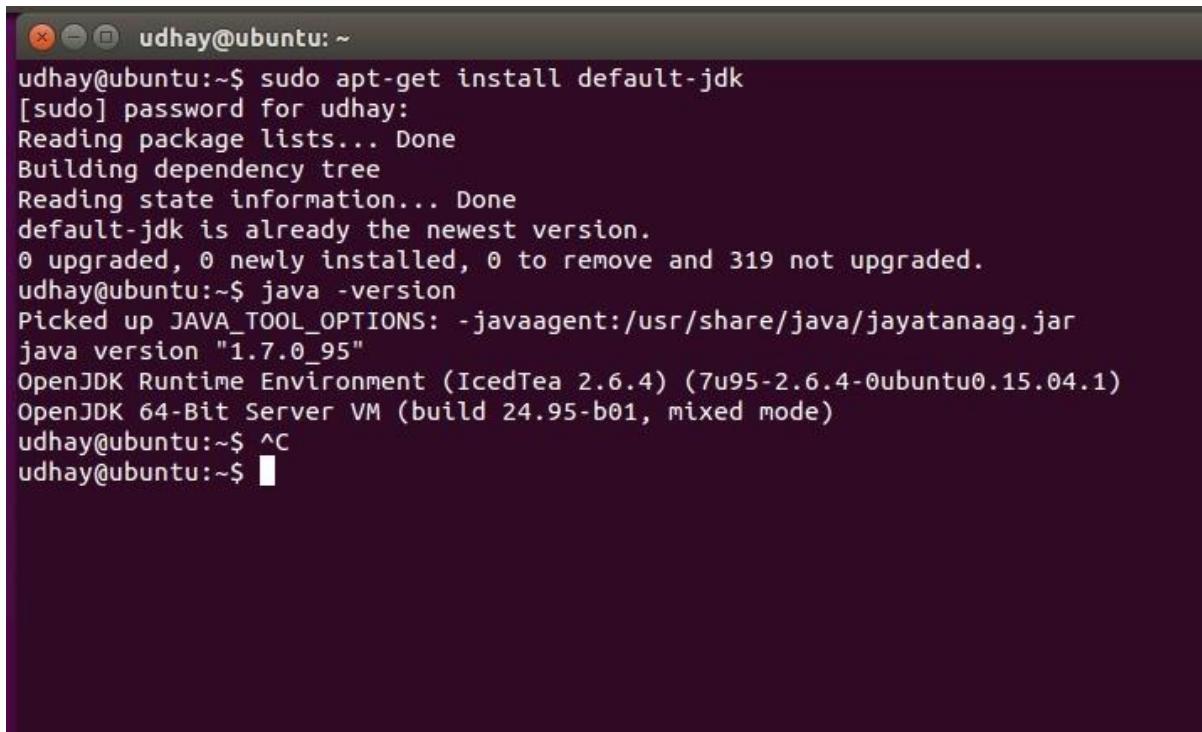
```
// Move the Hadoop folder to /usr/local/hadoop
```

```
$ sudo mv hadoop-2.7.2 /usr/local/hadoop
```

```
// Assigning read and write access to Hadoop folder
```

```
$ sudo chown -R hduser:hadoop /usr/local/hadoop
```

**Implementation:**



The screenshot shows a terminal window titled "udhay@ubuntu: ~". It displays the output of several commands:

```
udhay@ubuntu:~$ sudo apt-get install default-jdk
[sudo] password for udhay:
Reading package lists... Done
Building dependency tree
Reading state information... Done
default-jdk is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 319 not upgraded.
udhay@ubuntu:~$ java -version
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
java version "1.7.0_95"
OpenJDK Runtime Environment (IcedTea 2.6.4) (7u95-2.6.4-0ubuntu0.15.04.1)
OpenJDK 64-Bit Server VM (build 24.95-b01, mixed mode)
udhay@ubuntu:~$ ^C
udhay@ubuntu:~$
```

```
udhay@ubuntu:~$ sudo apt-get install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
ssh is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 319 not upgraded.
udhay@ubuntu:~$ su hduser
Password:
hduser@ubuntu:/home/udhay$
```

```
udhay@ubuntu:~$ su hduser
Password:
hduser@ubuntu:/home/udhay$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
/home/hduser/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
09:0f:15:f2:b2:b7:5e:11:1a:6c:d3:2f:c3:09:02:15 hduser@ubuntu
The key's randomart image is:
+---[RSA 2048]---+
|   ..E.o.      |
|   . = .       |
|   = B o      |
|   O B +      |
|   . S * .    |
|   . . +     |
|   . .        |
|   .          |
+-----+
hduser@ubuntu:/home/udhay$
```

```
hduser@ubuntu:/home/udhay$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
hduser@ubuntu:/home/udhay$ ssh localhost
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-84-generic x86_64)
```

\* Documentation: <https://help.ubuntu.com/>

```
Last login: Thu Jul 15 22:00:14 2021 from localhost
hduser@ubuntu:~$
```

**About the Cluster**

**Cluster Metrics**

	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	ID
0	0	0	0	0	0	0 B	8 GB	0 B	0	8	0	1	0

**Scheduler Metrics**

Scheduler Type	Scheduling Resource Type	Minimum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>

**Cluster Configuration**

- Cluster ID: 1626414170591
- ResourceManager state: STARTED
- ResourceManager HA state: active
- ResourceManager HA zookeeper connection state: ResourceManager HA is not enabled.
- ResourceManager RMStateStore: org.apache.hadoop.yarn.server.resourcemanager.recovery.NullRMStateStore
- ResourceManager started on: Thu Jul 15 22:42:50 -0700 2021
- ResourceManager version: 2.7.2 from b165c4fe8a74265c792ce23f546c64604acf0e41 by Jenkins source checksum 2016-01-26T00:16Z
- Hadoop version: 2.7.2 from b165c4fe8a74265c792ce23f546c64604acf0e41 by Jenkins source checksum 2016-01-26T00:08Z

Activate Windows

## RESULT:

**HDUSER** is created with necessary permissions for Hadoop installation.  
SSH localhost configuration is set up, allowing passwordless SSH authentication for **HDUSE**

## **EXP. 10: INSTALL HADOOP 2.X AND CONFIGURE THE NAME NODE AND DATA NODE.**

### **AIM:**

Install Hadoop 2.x and configure the NameNode and DataNode.

### **PROCEDURE:**

#### **Step 7 - Modify Hadoop config files**

//Hadoop Environmental variable setting – The following files will be modified

1. `~/.bashrc`
2. `/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/hadoop-env.sh`
3. `/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/core-site.xml`
4. `/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/hdfs-site.xml`
5. `/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/yarn-site.xml`
6. `/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/mapred-site.xml.template`

```
$ sudo nano ~/.bashrc
```

// Add the following lines at the end of the file

```
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
export HADOOP_HOME=/usr/local/hadoop/hadoop-2.7.2
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
export PATH=$PATH:/usr/local/hadoop/hadoop-2.7.2/bin
```

// Configure Hadoop Files

```
$ cd /usr/local/hadoop/hadoop-2.7.2/etc/hadoop/
```

```
$ sudo nano hadoop-env.sh
```

// Add following line in hadoop-env.sh – Set JAVA variable in Hadoop

```
# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
```

// Create datanode and namenode

```
$ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/namenode  
$ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/datanode  
// Changing ownership to hadoop_tmp  
$ sudo chown -R hduser:hadoop /usr/local/hadoop_tmp  
// Edit hdfs-site.xml  
$ sudo nano hdfs-site.xml
```

// Add the following lines between <configuration> ..... </configuration>

```
<configuration>  
<property>  
<name>dfs.replication</name>  
<value>1</value>  
</property>  
<property>  
<name>dfs.namenode.name.dir</name>  
<value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>  
</property>  
<property>  
<name>dfs.datanode.data.dir</name>  
<value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>  
</property>  
</configuration>
```

// Edit core-site.xml

```
$ sudo nano core-site.xml
```

// Add the following lines between <configuration> ..... </configuration>

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

// Edit yarn-site.xml

```
$ sudo nano yarn-site.xml
```

// Add the following lines between <configuration> ..... </configuration>

```
<configuration>  
<property>  
<name>yarn.nodemanager.aux-services</name>  
<value>mapreduce_shuffle</value>  
</property>  
<property>
```

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

// Edit mapred-site.xmsudo

```
$ cp /usr/local/hadoop/hadoop-2.7.2/etc/hadoop/mapred-site.xml.template
/usr/local/hadoop/hadoop-2.7.2/etc/hadoop/mapred-site.xml
```

```
$ sudo nano mapred-site.xml
```

// Add the following lines between <configuration> ..... </configuration>

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

**Step-8** – Format Hadoop File System

```
$ cd /usr/local/hadoop/hadoop-2.7.2/bin
$ hadoop namenode -format
```

**Step 9** - Start Hadoop

```
$ cd /usr/local/hadoop/hadoop-2.7.2/sbin
```

// Starting dfs services

```
$ start-dfs.sh
```

// Starting mapreduce services

```
$ start-yarn.sh
```

```
$ jps
```

**Step 10** - Check Hadoop through web UI

Go to browser type <http://localhost:8088> – All Applications Hadoop Cluster

Go to browser type <http://localhost:50070> – Hadoop Namenode

**Step 11** - Stop Hadoop

```
$ stop-dfs.sh
```

```
$ stop-yarn.sh
```

## IMPLEMENTATION:

```
GNU nano 2.2.6                                         File: /home/hduser/.bashrc

# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END
```

```
hduser@ubuntu:/home$ cd ..
hduser@ubuntu:/$ cd usr
hduser@ubuntu:/usr$ cd local
hduser@ubuntu:/usr/local$ cd hadoop
hduser@ubuntu:/usr/local/hadoop$ cd etc
hduser@ubuntu:/usr/local/hadoop/etc$ cd hadoop
hduser@ubuntu:/usr/local/hadoop/etc/hadoop$ ls
capacity-scheduler.xml      httpfs-env.sh          mapred-env.sh
configuration.xsl           httpfs-log4j.properties  mapred-queues.xml.template
container-executor.cfg       httpfs-signature.secret mapred-site.xml
core-site.xml                httpfs-site.xml        mapred-site.xml.template
hadoop-env.cmd               kms-acls.xml         slaves
hadoop-env.sh                kms-env.sh           ssl-client.xml.example
hadoop-metrics2.properties   kms-log4j.properties  ssl-server.xml.example
hadoop-metrics.properties    kms-site.xml         yarn-env.cmd
hadoop-policy.xml            log4j.properties     yarn-env.sh
hdfs-site.xml                mapred-env.cmd       yarn-site.xml
hduser@ubuntu:/usr/local/hadoop/etc/hadoop$
```

```
hduser@ubuntu: /usr/local/hadoop/etc/hadoop
GNU nano 2.2.6          File: hadoop-env.sh

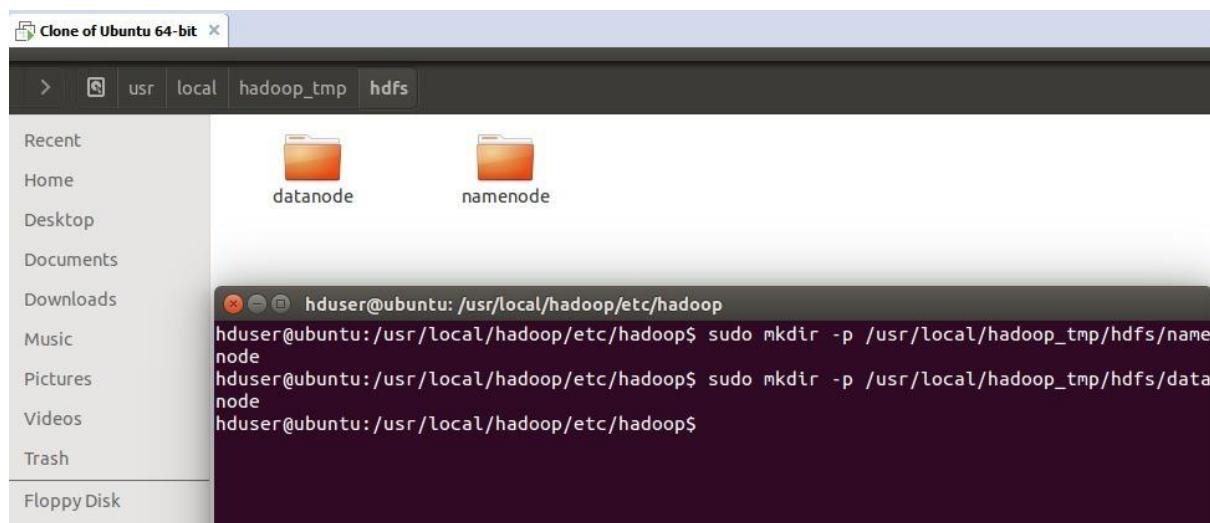
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# Set Hadoop-specific environment variables here.

# The only required environment variable is JAVA_HOME. All others are
# optional. When running a distributed configuration it is best to
# set JAVA_HOME in this file, so that it is correctly defined on
# remote nodes.

# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export JAVA_HOME=${JAVA_HOME}

# The jsvc implementation to use. Jsvc is required to run secure datanodes
# that bind to privileged ports to provide authentication of data transfer
# protocol. Jsvc is not required if SASL is configured for authentication of
# data transfer protocol using non-privileged ports.
```



## RESULT:

- Hadoop 2.x is installed and configured successfully.
- NameNode and DataNode are properly configured and running.
- Hadoop cluster is ready to process and store big data efficiently.

## **EXP. 11 LAUNCH THE HADOOP 2.X AND PERFORM MAPREDUCE PROGRAM FOR A WORD COUNT PROBLEM**

### **AIM:**

Launch Hadoop 2.x and perform a MapReduce program for a word count problem.

### **PROCEDURE:**

#### **Step 1 - Open Terminal**

```
$ su hduser
```

Password:

#### **Step 2 - Start dfs and mapreduce services**

```
$ cd /usr/local/hadoop/hadoop-2.7.2/sbin
```

```
$ start-dfs.sh
```

```
$ start-yarn.sh
```

```
$ jps
```

#### **Step 3 - Check Hadoop through web UI**

// Go to browser type <http://localhost:8088> – All Applications Hadoop Cluster

// Go to browser type <http://localhost:50070> – Hadoop Namenode

#### **Step 4 – Open New Terminal**

```
$ cd Desktop/
```

```
$ mkdir inputdata
```

```
$ cd inputdata/
```

```
$ echo "Hai, Hello, How are you? How is your health?" >> hello.txt
```

```
$ cat >> hello.txt
```

#### **Step 5 – Go back to old Terminal**

```
$ hadoop fs –copyFromLocal /home/hduser/Desktop/inputdata/hello.txt /folder/hduser
```

// Check in hello.txt in Namenode using Web UI

#### **Step 6 – Download and open eclipse by creating workspace**

Create a new java project.

#### **Step 7 – Add jar to the project**

You need to remove dependencies by adding jar files in the hadoop source folder. Now Click on **Project** tab and go to Properties.Under Libraries tab, click Add External JARs and select all the

jars in the folder (click on 1st jar, and Press Shift and Click on last jar to select all jars in between and click ok)

/usr/local/hadoop/hadoop-2.7.2/share/hadoop/common and  
/usr/local/hadoop/hadoop-2.7.2/share/hadoop/mapreduce folders.

### Step -8 – WordCount Program

Create 3 java files named

- WordCount.java
- WordCountMapper.java
- WordCountReducer.java

#### WordCount.java

```
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.mapred.FileInputFormat;
import org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobClient;
import org.apache.hadoop.mapred.JobConf;

import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
import org.apache.hadoop.io.Text;

public class WordCount extends Configured implements Tool {

    @Override
    public int run(String[] args) throws Exception {
        // TODO Auto-generated method stub
        if(args.length<2)
        {
            System.out.println("check the command line arguments");
        }
        JobConf conf=new JobConf(WordCount.class);
        FileInputFormat.setInputPaths(conf, new Path(args[0]));
        FileOutputFormat.setOutputPath(conf, new Path(args[1]));
        conf.setMapperClass(WordMapper.class);
        conf.setReducerClass(WordReducer.class);
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(IntWritable.class);
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(IntWritable.class);
        JobClient.runJob(conf);

        return 0;
    }
    public static void main(String args[]) throws Exception
    {
        int exitcode=ToolRunner.run(new WordCount(), args);
        System.exit(exitcode);
    }
}
```

```
    }
}
```

### WordCountMapper.java

```
import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.Mapper;

public class WordCountMapper extends MapReduceBase implements
Mapper<LongWritable,Text,Text,IntWritable>
{
    @Override
    public void map(LongWritable arg0, Text arg1, OutputCollector<Text, IntWritable> arg2,
Reporter arg3)
        throws IOException {
        // TODO Auto-generated method stub

        String s=arg1.toString();
        for(String word:s.split(" "))
        {
            arg2.collect(new Text(word),new IntWritable(1));
        }
    }
}
```

### WordCountReducer.java

```
import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
import org.apache.hadoop.io.Text;

public class WordCountReducer implements Reducer<Text,IntWritable,Text,IntWritable> {
    @Override
    public void configure(JobConf arg0) {
        // TODO Auto-generated method stub
    }
    @Override
    public void close() throws IOException {
        // TODO Auto-generated method stub
    }
}
```

```

    }
    @Override
    public void reduce(Text arg0, Iterator<IntWritable> arg1, OutputCollector<Text, IntWritable>
arg2, Reporter arg3)
        throws IOException {
        // TODO Auto-generated method stub
        int count=0;
        while(arg1.hasNext())
        {
            IntWritable i=arg1.next();
            count+=i.get();
        }
        arg2.collect(arg0,new IntWritable(count));
    }
}

```

### Step 9 - Create JAR file

Now Click on the Run tab and click Run-Configurations. Click on New Configuration button on the left-top side and Apply after filling the following properties.

### Step 10 - Export JAR file

Now click on File tab and select Export. under Java, select Runnable Jar.

In Launch Config – select the config file you created in **Step 9** (WordCountConfig).

Select an export destination (lets say desktop.)

Under Library handling, select Extract Required Libraries into generated JAR and click Finish.

Right-Click the jar file, go to Properties and under **Permission** tab, Check Allow executing file as a program. and give Read and Write access to all the users

### Step 11 – Go back to old Terminal for Execution of WordCount Program

**\$hadoop jar wordcount.jar/usr/local/hadoop/input/usr/local/hadoop/output**

Name	Type	Owner	Group	Size	Last Modified	Replication	BlockSize
cloud	drwxr-xr-x	hduser	supergroup	0 B	8/12/2016, 12:20:50 AM	0	0 B
cse	drwxr-xr-x	hduser	supergroup	0 B	8/11/2016, 1:47:41 AM	0	0 B
folder	drwxr-xr-x	hduser	supergroup	0 B	8/4/2016, 11:37:37 PM	0	0 B
grid	drwxr-xr-x	hduser	supergroup	0 B	8/11/2016, 9:52:15 PM	0	0 B
output	drwxr-xr-x	hduser	supergroup	0 B	8/11/2016, 9:54:38 PM	0	0 B
project	drwxr-xr-x	hduser	supergroup	0 B	8/11/2016, 11:54:23 PM	0	0 B
tmp	drwx-----	hduser	supergroup	0 B	8/4/2016, 11:40:37 PM	0	0 B

### Step 12 – To view results in old Terminal

```
$hdfs dfs -cat /usr/local/hadoop/output/part-r-00000
```

```
hadoop1@ubuntu-1:~/project$ hadoop fs -cat /output/wordcount4/part-r-00000
.
a 1
and 1
as 1
count 1
counts 1
file 2
for 1
input 1
is 1
job 1
job. 1
map 1
returns 1
sample 1
takes 1
```

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	hduser	supergroup	0 B	8/11/2016, 9:54:38 PM	1	128 MB	_SUCCESS
-rw-r--r--	hduser	supergroup	44 B	8/11/2016, 9:54:38 PM	1	128 MB	part-00000

### Step 13 - To Remove folders created using hdfs

```
$ hdfs dfs -rm -R /usr/local/hadoop/output
```

#### RESULT:

- Hadoop 2.x is successfully launched.
- The MapReduce program for the word count problem is executed on the Hadoop cluster.
- The program accurately counts the occurrences of each word in the input data.
- Output containing word counts is obtained, demonstrating the power of Hadoop for processing large-scale data using MapReduce paradigm.