

# PRACTICAL FILE



## MCA-223 CLOUD COMPUTING

**SUBMITTED BY:**

TARUN ANAND GOYAL  
00911104422  
MCA 3<sup>rd</sup> SEM

**SUBMITTED TO:**

Mr. Alok Mishra  
ASST.  
PROFFESOR

**BANARSIDAS CHANDIWALA INSTITUTE OF INFORMATION TECHNOLOGY**  
*AFFILIATED WITH*  
**GURU GOBIND SINGH INDRAAPRASTH UNIVERSITY, DELHI**

# Index

<b>Sr. no</b>	<b>Practical</b>	<b>Pageno</b>
1	Create an AWS account ,Azure account and google cloud account	4
2	Set up a budget to an AWS account	9
3	Launch a windows server instance with t2.micro instance type and create a security group by using EC2	13
4	Connect the launch instances 2/2 status check and decrypt password by using RDP client	17
5	Terminate the launched window server instance from AWS EC2.	19
6	Write the steps to launch a Linux server by using AWS EC2.	21
7	Write the steps to connect with the window server by using AWS EC2.	24
8	Launch a website on a windows server using EC2.	28
9	Terminate the launched Linux server instance from AWS EC2.	35
10	create a IAM (Identify and Access management) user.	37
11	connect with a launched instance of Linux (putty & putty gen software).	41
12	Create IAM user and grant in limited permission to IAM user by AWS route user	48
13	Create a bucket by using S3 aws service	52
14	Upload an object on bucket created by using S3 AWS service	55
15	Create a bucket and allow public access on uploaded objects by using object URL and S3 aws surface	58
16	Delete the object and bucket by using S 3 interface	63
17	Transfer the object file from S3 service to EC2 launched Linux server install GCC and wget commands in this regard on terminal	
18	Create a VPC and implement EC2 services on it	65
19	Implement and configure load balancing with all necessary steps	69

20	How to handle a cloud shell explain it	74
21	Create a private cloud on Google Drive and Grant restrict permissions for the user	75

## Practical 1: Create an AWS account.

**Objective :** The objective of creating an AWS (Amazon Web Services) account is to enable individuals, organizations, and businesses to access and use AWS cloud services, resources, and infrastructure. AWS provides a wide range of cloud computing services, including computing power, storage, databases, machine learning, analytics, networking, content delivery, and more. Creating an AWS account allows you to:

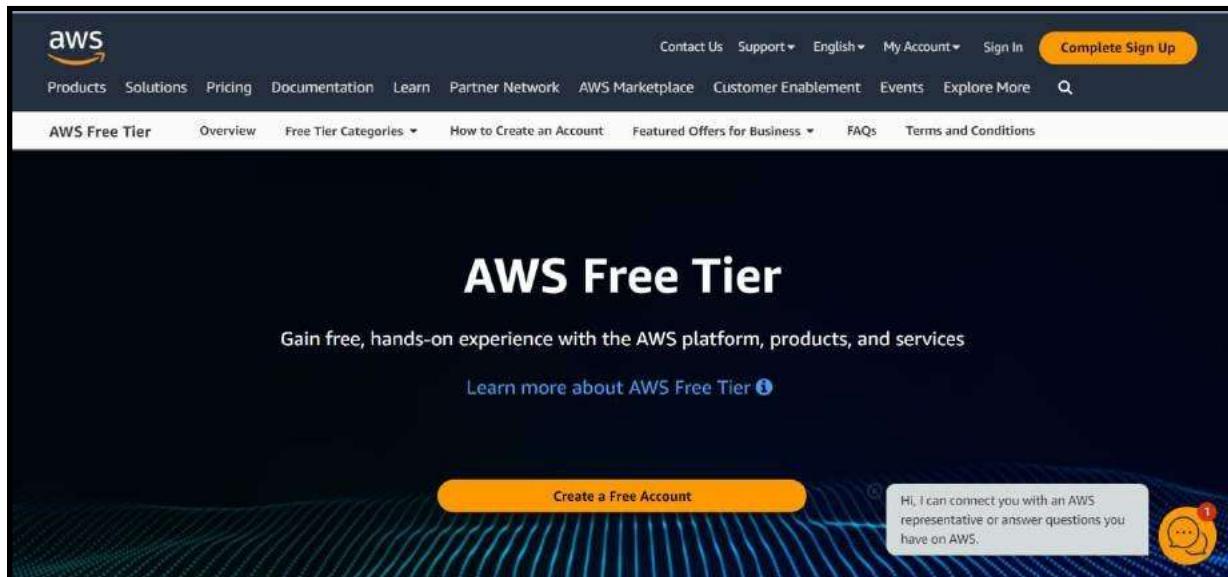
- Access AWS Services: Gain access to a vast array of cloud services, from virtual servers (EC2) to scalable storage (S3) and advanced machine learning (SageMaker).
- Scalability: Easily scale your infrastructure and resources up or down to meet your changing needs, ensuring that you only pay for what you use.
- Flexibility: Choose from a variety of operating systems, programming languages, databases, and other tools to run your applications.
- Security and Compliance: Use AWS's security features and compliance certifications to secure your data and applications, ensuring they meet industry standards and regulatory requirements.
- Cost-Efficiency: Optimize your infrastructure and reduce operational costs by utilizing AWS's pay-as-you-go pricing model.

### Step 1:

First Open your web browser and navigate to AWS Free Tier Page

### Step 2:

On middle click of Create a Free Account



### Step 3:

Issue the details which you want to use to log in to your AWS account and click on Continue

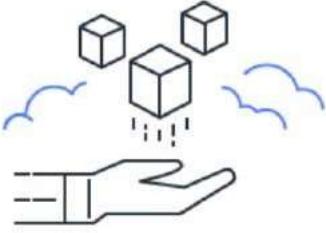
- **Email address:** Provide the mail id which hasn't been registered yet with Amazon AWS.
- **Password:** Type your password.
- **Confirm password:** Authenticate the password.
- **AWS Account name:** Choose a name for your account. You can change this name in your account settings after you sign up



## Sign up for AWS

**Explore Free Tier products with a new AWS account.**

To learn more, visit [aws.amazon.com/free](http://aws.amazon.com/free).



**Root user email address**  
Used for account recovery and some administrative functions

**AWS account name**  
Choose a name for your account. You can change this name in your account settings after you sign up.

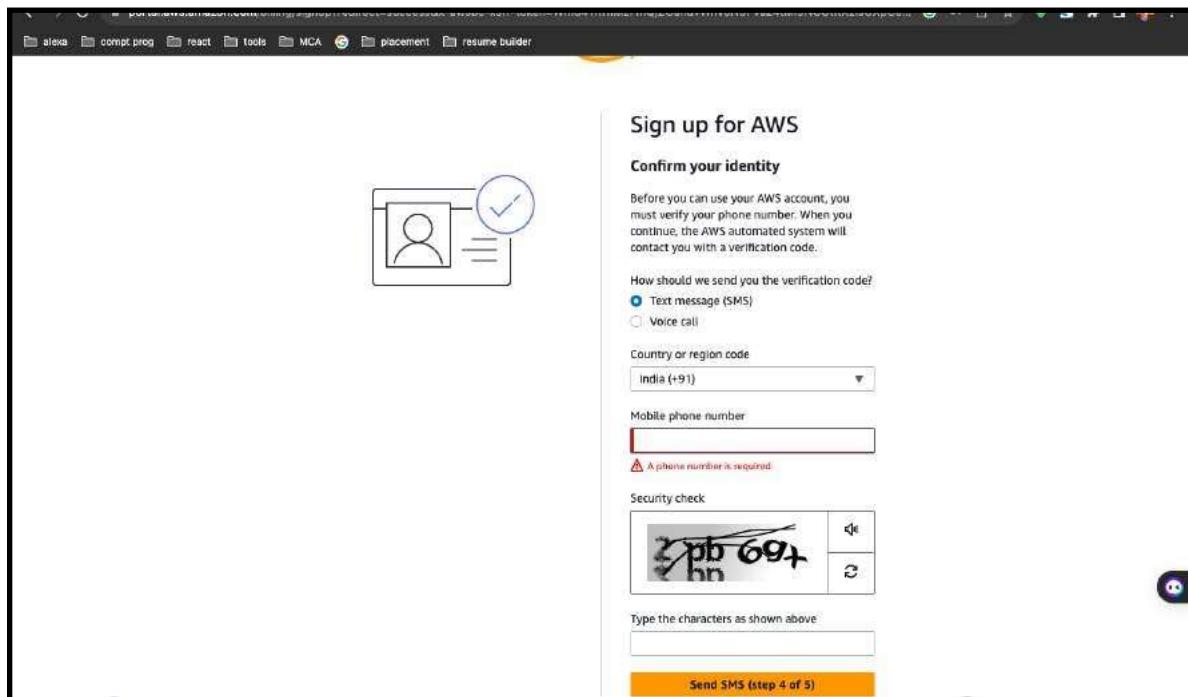
**Verify email address**

OR

**Sign in to an existing AWS account**

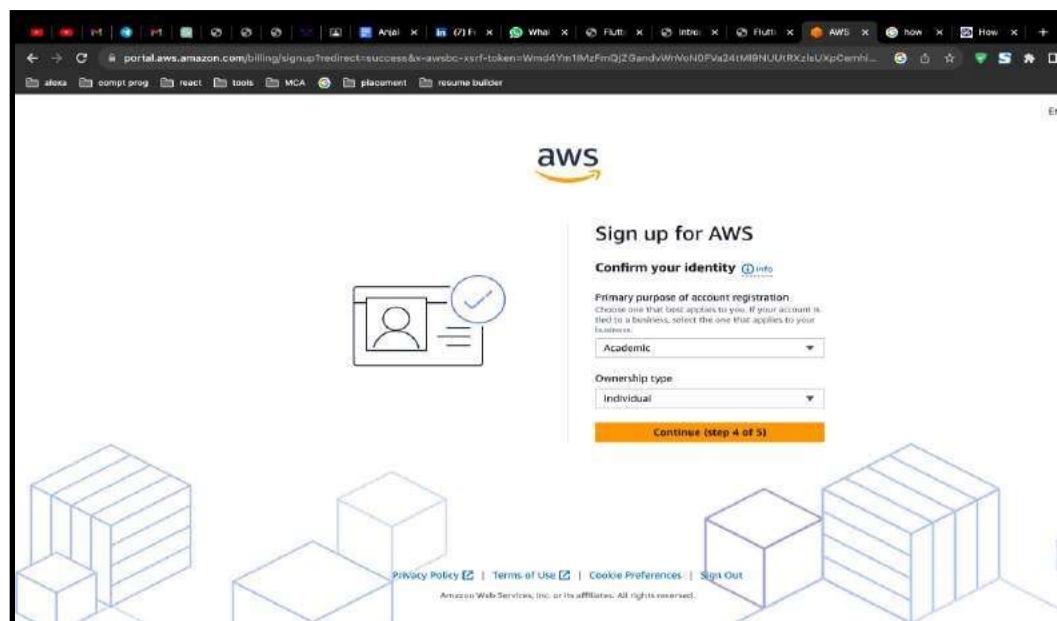
## Step 4:

**Phone verification:** Here you will be taken to an identity verification page that will already have your phone number, so you just have to select either “Text message or Voice call” Provide a valid phone number, Solve the captcha, and then click on Send SMS or Call Me Now(depending upon your selection).



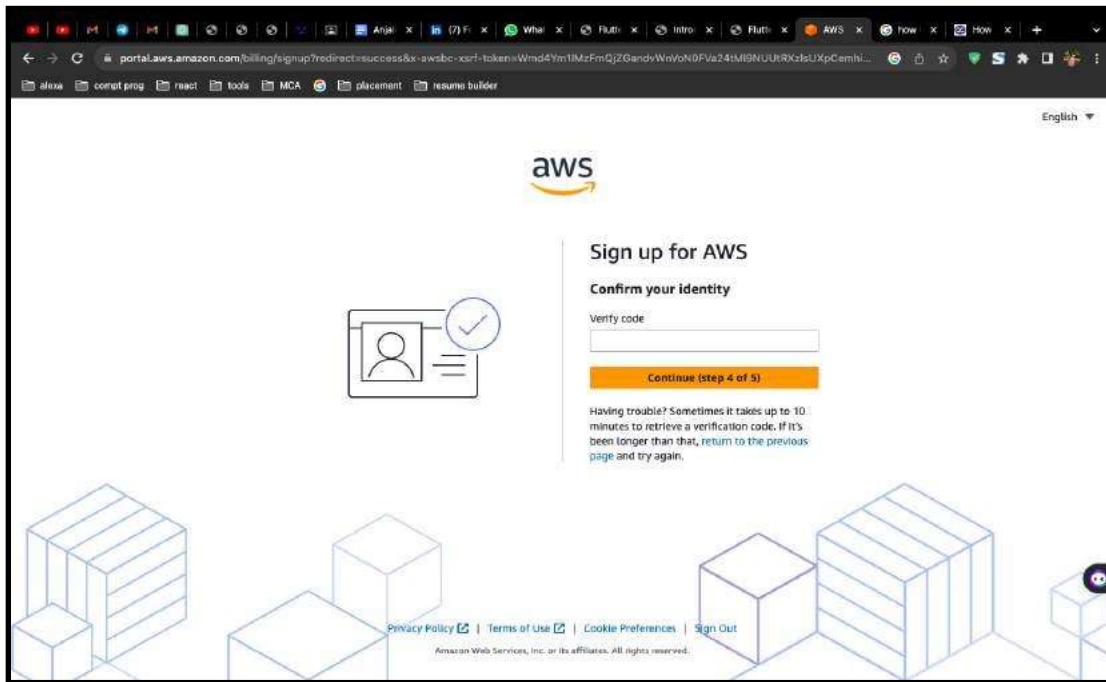
## Step 5 :

Enter your Purpose of Account Registration



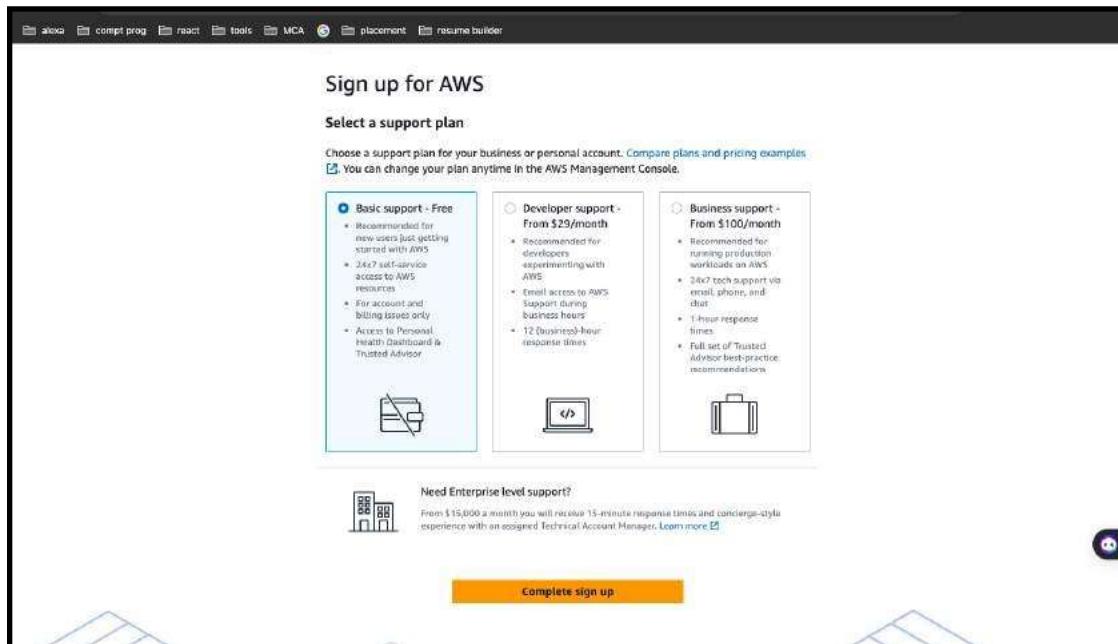
## Step 6:

After clicking on Send SMS or Call me Now, you will immediately receive a call or SMS from Amazon, for verification code, Enter your code then click on Verify Code.



## Step 7:

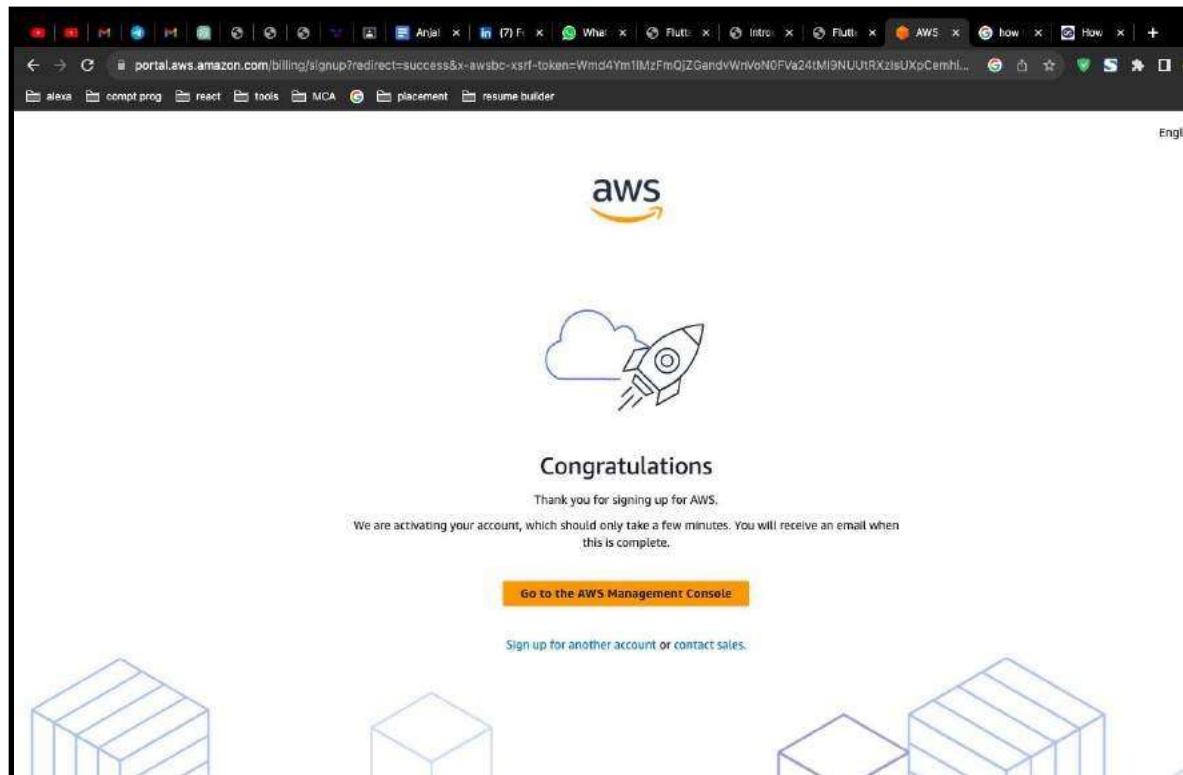
**Support plan:** AWS support offers a selection of plans to meet your business needs. Select your suitable plan then click continue.



## Step 8:

### Registration Confirmation page.

Once you complete all the above steps and processes. You'll get the confirmation page below. Now your account will be processed for activation. It may take somewhere between 30 minutes to 1 hour for you to receive an email confirmation that your Amazon Cloud Services account has been activated



## Practical 2:

### Set up Budget to an AWS account.

**Objective :**The primary object of AWS Budgets is to help AWS customers manage and control their cloud spending effectively. AWS Budgets is a cost management tool provided by Amazon Web Services (AWS) to help organizations set and track budgets for their AWS spending. Its main objectives are:

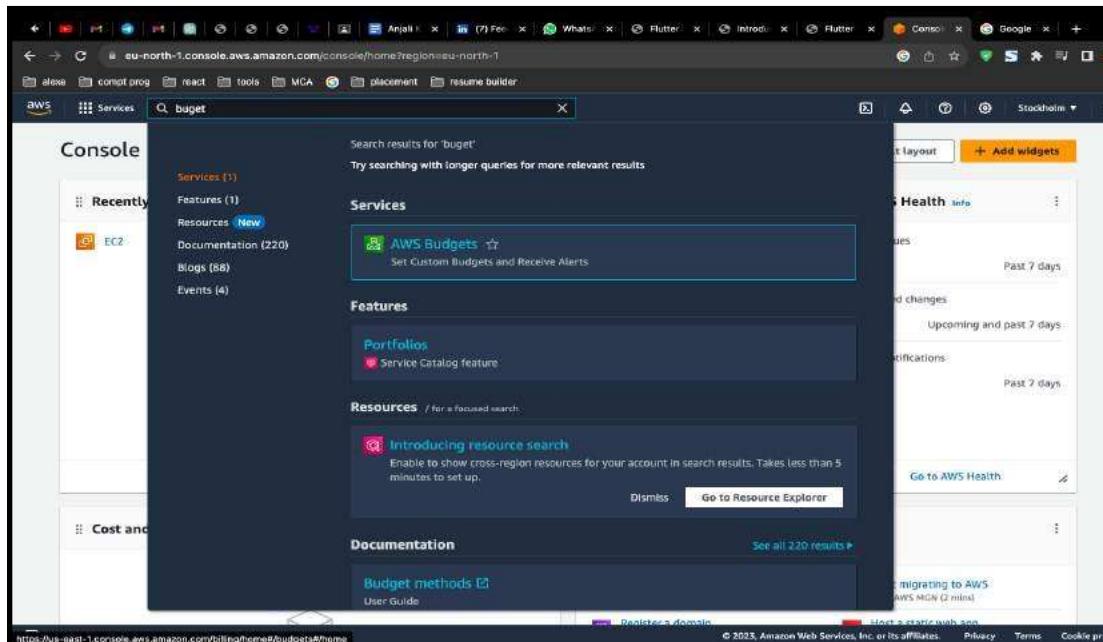
- Cost Monitoring: AWS Budgets provides insights into your AWS cost and usage data, allowing you to monitor your spending in real-time. This helps you keep track of your expenses and ensure they align with your budgetary goals.
- Budget Setting: The service allows you to set specific budgets for different aspects of your AWS usage, such as overall costs, service costs, or specific cost and usage patterns. You can create custom budgets that align with your business objectives.
- Cost Alerts: AWS Budgets enables you to set up cost and usage alerts. When your actual spending approaches or exceeds your budget thresholds, AWS Budgets will notify you via email or SNS (Simple Notification Service). This helps you take timely action to avoid unexpected overages.

#### Step 1:

Sign in to the AWS Management Console and open the AWS Cost Management console at <https://console.aws.amazon.com/cost-management/home>.

#### Step 2:

In the navigation pane, choose Budgets.



## Step 3:

At the top of the page, choose Create budget.

The screenshot shows the AWS Billing console with the 'Billing' service selected in the sidebar. The main page is titled 'AWS Budgets' and features a dark header with the text 'Set custom budgets that alert you when you exceed your budgeted thresholds'. Below the header, there's a section titled 'How it works' with a flowchart showing the process from creating a budget to receiving notifications. To the right, there are sections for 'Start tracking your AWS costs and usage' (with a 'Create a budget' button) and 'Pricing (US)' (noting no additional charge). At the bottom, there's a 'Getting started' section.

## Step 4:

Under Details, for Budget name, enter the name of your budget. Your budget name must be unique within your account. It can contain A-Z, a-z, spaces, and the following characters:

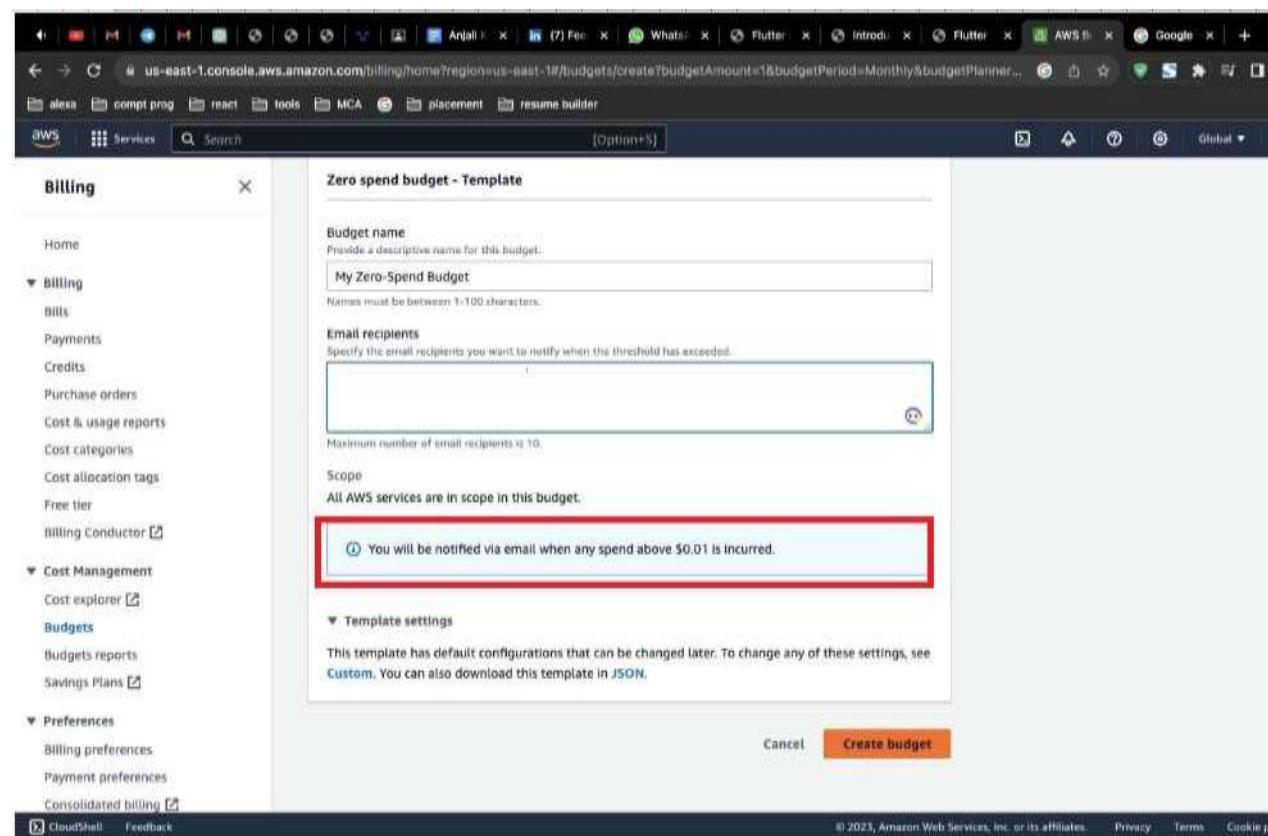
The screenshot shows the 'Create budget' wizard in the AWS Billing console. The left sidebar shows the 'Billing' service selected. The main area has two options: 'Use a template (simplified)' (selected) and 'Customize (advanced)'. Under 'Templates - new', there are four options: 'Zero spend budget', 'Monthly cost budget', 'Daily Savings Plans coverage budget', and 'Daily reservation utilization budget'. Below this, the 'Zero spend budget - Template' section is shown, with fields for 'Budget name' (containing 'My Zero-Spend Budget') and 'Email recipients'. The bottom of the page includes standard copyright and legal links.

## Step 5:

Under Set alert threshold, for Threshold, enter the amount that must be reached for you to be notified. This can be either an absolute value or a percentage.

(Optional) Under Notification preferences, for Email recipients, enter the email addresses that you want the alert to notify. Separate multiple email addresses with commas. A notification can be sent to a maximum of 10 email addresses.

(Optional) Under Notification preferences, for Amazon SNS Alerts, enter the Amazon Resource Name (ARN) for your Amazon SNS topic. For instructions on how to create a topic, see Creating an Amazon SNS topic for budget notifications.



The screenshot shows the AWS Billing console with the 'Billing' template selected. The left sidebar shows various navigation options like Home, Billing, Payments, Credits, Purchase orders, Cost & usage reports, Cost categories, Cost allocation tags, Free tier, Billing Conductor, Cost Management, Cost explorer, Budgets, Budgets reports, Savings Plans, Preferences, Billing preferences, Payment preferences, and Consolidated billing. The main panel is titled 'Zero spend budget - Template' and contains fields for 'Budget name' (set to 'My Zero-Spend Budget'), 'Email recipients' (empty), and 'Scope' (set to 'All AWS services are in scope in this budget'). A note at the bottom of the main panel states: 'You will be notified via email when any spend above \$0.01 is incurred.' This note is highlighted with a red box. At the bottom right of the main panel are 'Cancel' and 'Create budget' buttons. The footer includes links for CloudShell, Feedback, and copyright information: © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie policy.

## Step 6 :

Your Aws Budget is Created

The screenshot shows the AWS Billing console at the URL [us-east-1.console.aws.amazon.com/billing/home?region=us-east-1#/budgets/overview](https://us-east-1.console.aws.amazon.com/billing/home?region=us-east-1#/budgets/overview). The browser tab bar includes multiple tabs such as Alexa, Compt Prog, React, Tools, MCA, Placement, and Resume Builder. The AWS navigation bar has 'Services' selected. A green success message banner at the top right says: "Your budget My Zero-Spend Budget has been created successfully. After creating a budget, it can take up to 24 hours to populate all of your spend data." Below the banner, the left sidebar is open under the 'Billing' section, showing options like Home, Bills, Payments, Credits, Purchase orders, Cost & usage reports, Cost categories, Cost allocation tags, Free tier, Billing Conductor, Cost Management (Cost explorer, Budgets, Budgets reports, Savings Plans), Preferences (Billing preferences, Payment preferences, Consolidated billing), CloudShell, and Feedback. The main content area is titled 'Overview' and shows the 'Budgets (1) Info' table. The table has columns: Name, Thresholds, Budget, Amount used, Forecasted, and Current vs. budgeted. One row is listed: 'My Zero-Spend Budget' with 'OK' status, \$1.00 budget, \$0.00 used, and a neutral forecast. Buttons for 'Download CSV', 'Actions', and 'Create budget' are visible above the table. At the bottom of the page, there's a footer with links to 'Privacy' and 'Terms of Service'.

## Practical 3:

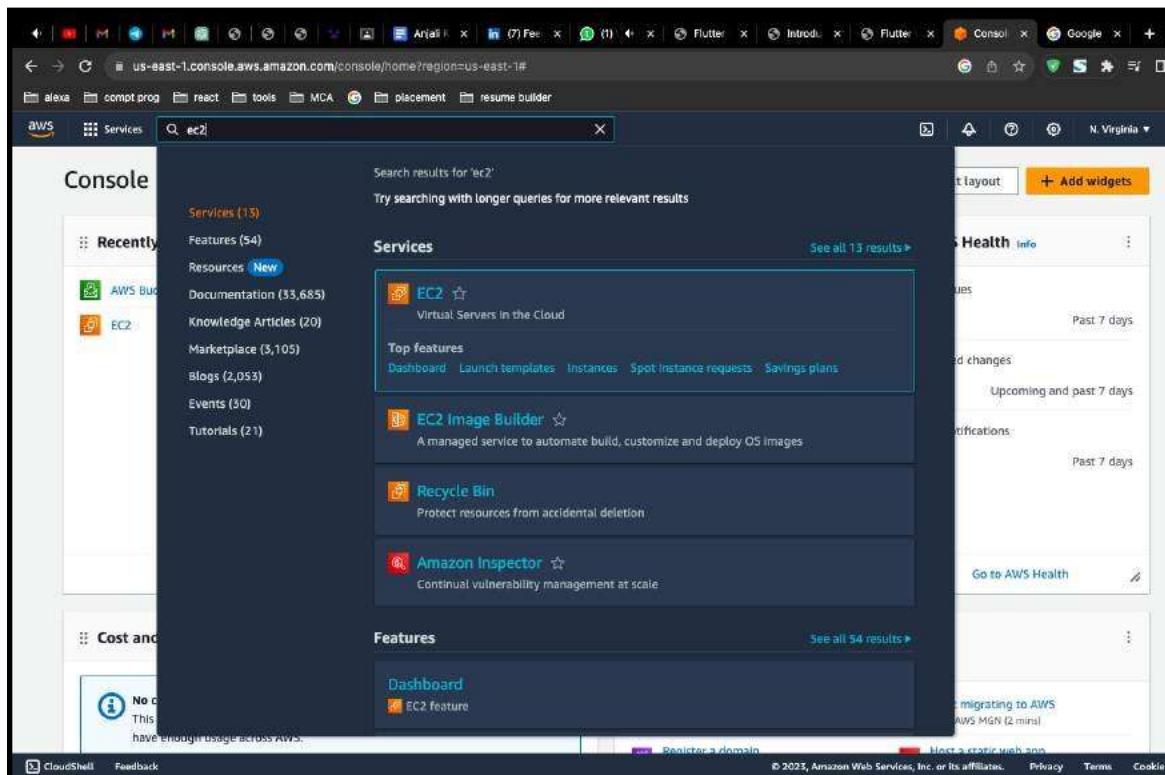
**Launch a WINDOW SERVER INSTANCE with t2.micro.instance type and create a security group by using EC2**

**Objective :**Launching a Windows Server instance in AWS EC2 serves a variety of purposes, depending on your specific needs and use cases. Here are some common reasons for launching a Windows Server in AWS EC2:

- Application Hosting: You can host Windows-based applications, including web servers, database servers, content management systems, and custom applications, on Windows Server instances in EC2.
  - Development and Testing: Windows Server instances are ideal for development and testing environments. You can create isolated development environments, test software, and simulate production environments on-demand.
  - Data Analysis and Reporting: Organizations often use Windows Servers in EC2 for data analysis, data warehousing, and generating reports using tools like SQL Server, Power BI, or custom analytics software.

## **Step1:**

Once logged in, navigate to the EC2 dashboard. You can do this by searching for "EC2" in the AWS Management Console's search bar or by selecting "Compute" and then "EC2" under the "Services" menu.

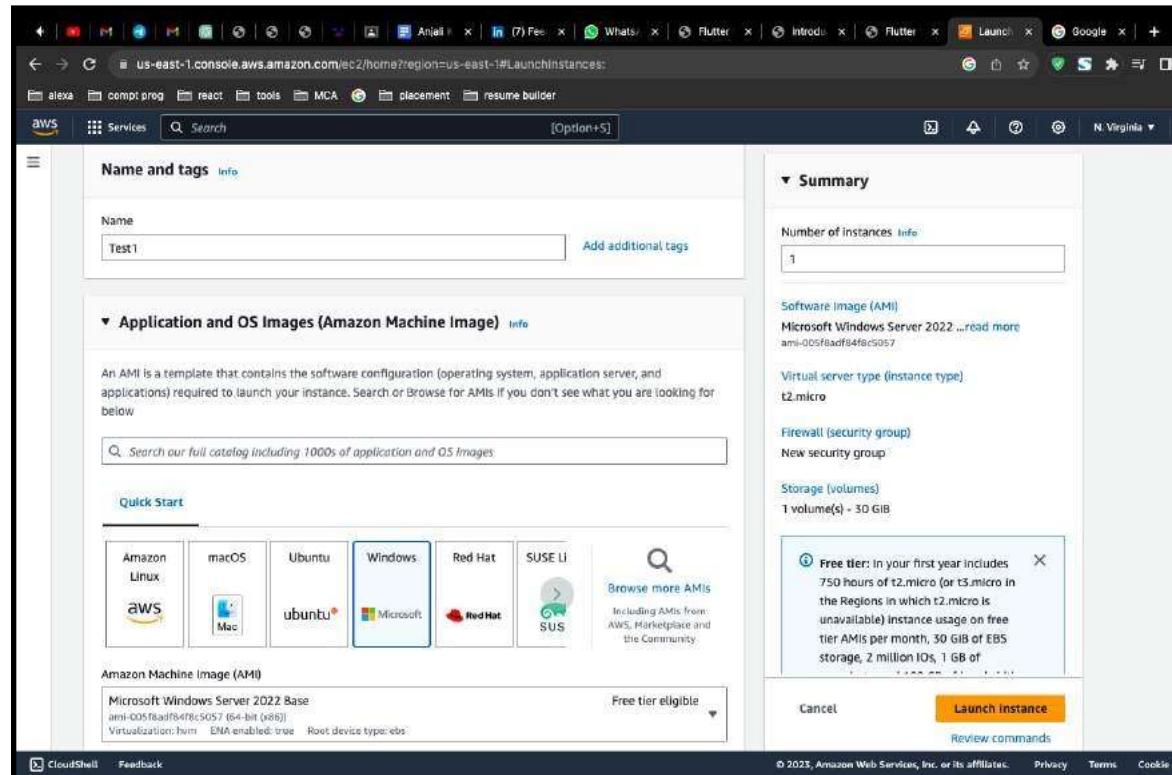


## Step 2:

Enter the name of your Aws Ec2 Instance

## Step 3:

In the "Choose an Amazon Machine Image (AMI)" step, search for a Windows Server AMI. AWS provides various Windows Server AMIs, including different Windows Server versions and editions. Select the one that suits your requirements.

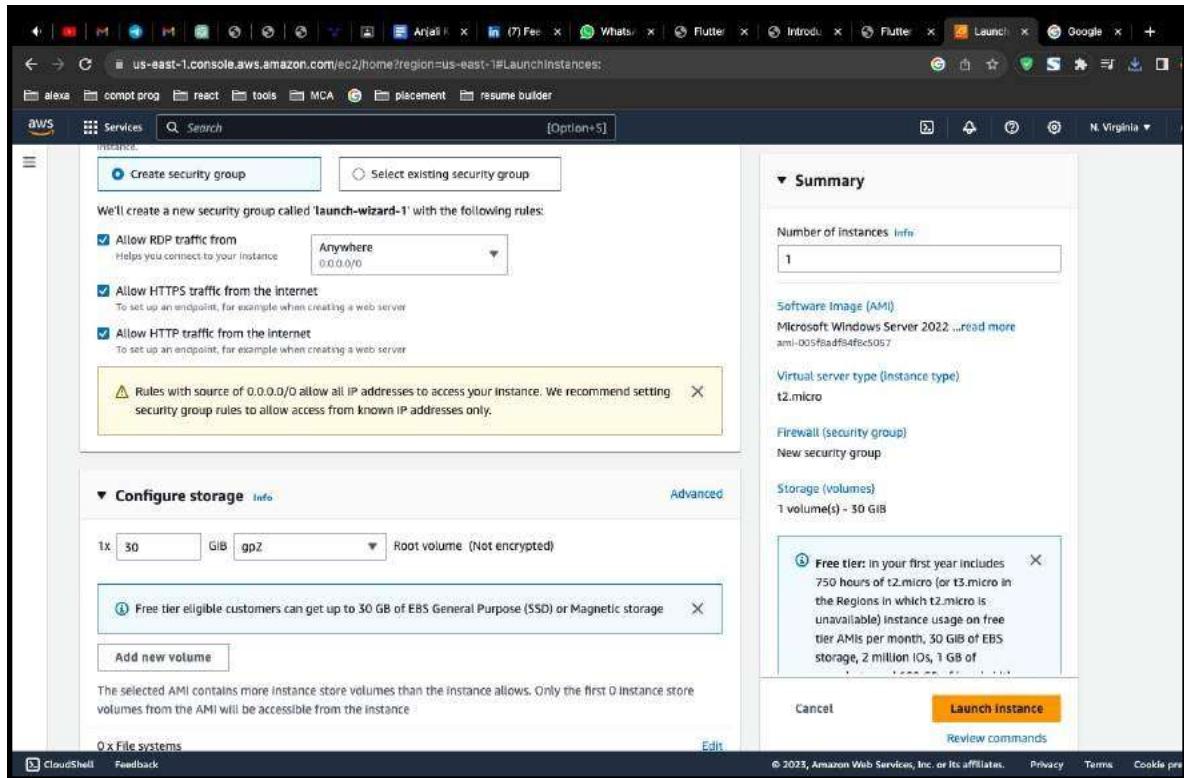


## Step 4:

In the "Add Storage" step, you can specify the size and type of the root volume for your instance. You can also add additional volumes if necessary.

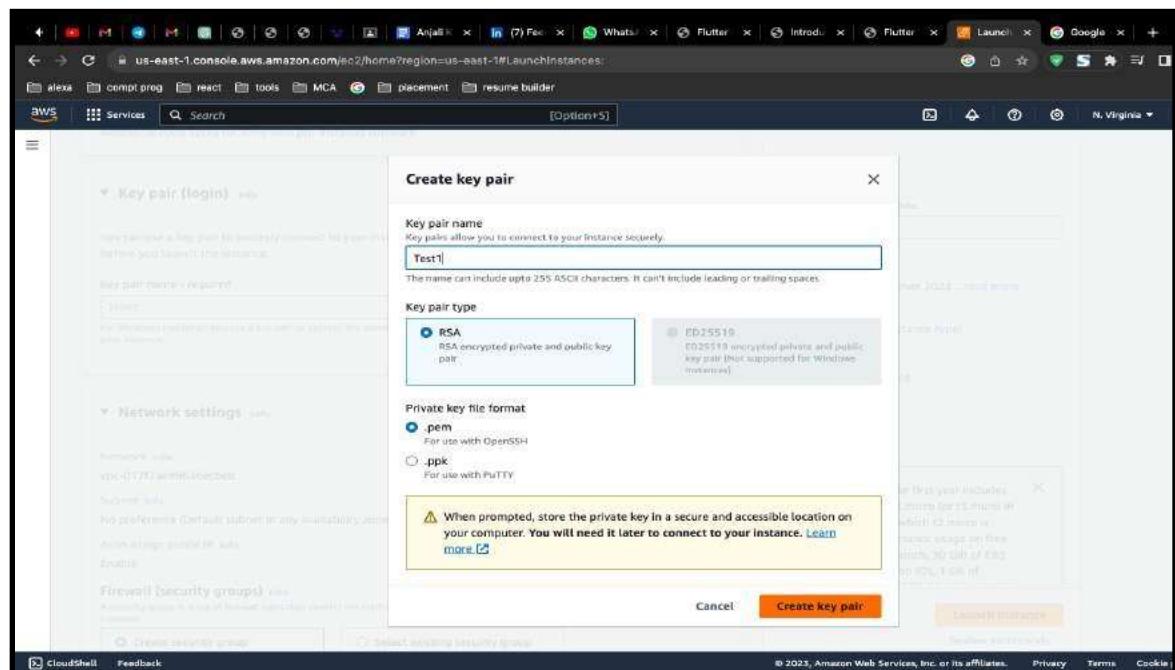
## Step 5:

In the "Configure Security Group" step, you'll need to configure the security group rules. Security groups act as firewalls to control inbound and outbound traffic to your instance. Ensure that you allow Remote Desktop Protocol (RDP) for Windows instances if you plan to access them remotely.



## Step 6:

If you haven't already created a key pair, you will be prompted to create one. This key pair is used to securely access your Windows Server instance. Download and save the key pair (.pem file) in a secure location.



## Step 6:

After selecting or creating a key pair, click the "Launch Instances" button.

The screenshot shows two consecutive screenshots of the AWS EC2 Instances launch process.

**Top Screenshot:** The progress bar is at 21% completion, showing the step: "Creating security group rules". Below the progress bar, there is a message: "Please wait while we launch your instance. Do not close your browser while this is loading."

**Bottom Screenshot:** The progress bar has completed at 100%, indicated by a green checkmark icon and the word "Success". The message says: "Successfully Initiated launch of Instance (i-02367495c4e5fe96)". Below this, there is a "Launch log" link and a "Next Steps" section.

**Next Steps Section:**

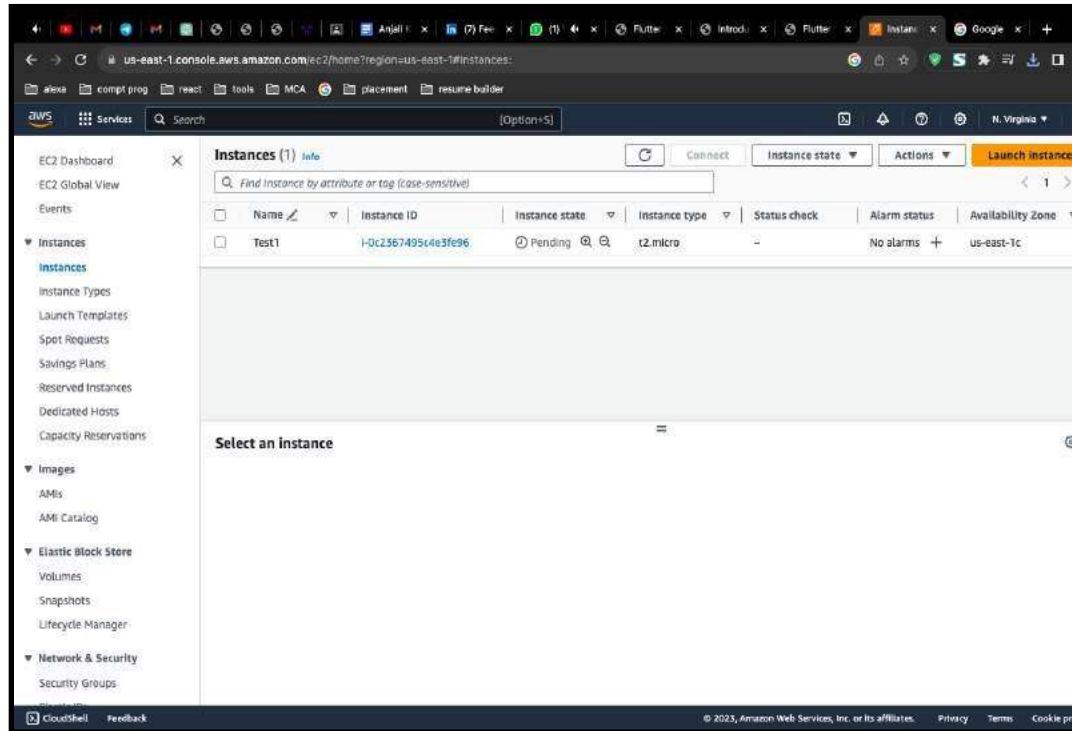
- Create billing and free tier usage alerts
- To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.  
[Create billing alerts](#)
- Connect to your instance
- Once your instance is running, log into it from your local computer.  
[Connect to Instance](#)  
[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 a new RDS database](#)  
[Learn more](#)
- Create EBS snapshot policy
- Create a policy that automates the creation, retention, and deletion of EBS snapshots.  
[Create EBS snapshot policy](#)
- Manage detailed monitoring
- Enable or disable detailed monitoring for
- Create Load Balancer
- Create a application, network gateway or
- Create AWS budget
- AWS Budgets allows you to create
- Manage CloudWatch alarms
- Create or update Amazon CloudWatch

## Practical 4:

### Connect the launch instance , 2/2 status check and decrypt password by using RDP client .

#### Step 1:

Select the Instance



The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), 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), CloudShell, and Feedback. The main content area has a header with 'Instances (1) Info' and a search bar. Below it is a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. A single row is selected for 'Test1' (Instance ID: i-0c2567495c4e3fe96, State: Pending, Type: t2.micro, Status Check: -). There are 'Connect', 'Actions', and 'Launch instance' buttons at the top right of the table. A large 'Select an instance' button is centered below the table.

#### Step 2:

You'll be taken to the "Instances" view, where you can see the status of your Windows Server instance as it starts. Once the instance is in a "running" state, you can connect to it using RDP.

EC2 > Instances > i-0ed023039c00b4423 > Connect to instance

## Connect to instance Info

Connect to your instance i-0ed023039c00b4423 (experiment) using any of these options

**Session Manager** | **RDP client** **EC2 serial console**

**Instance ID**  
i-0ed023039c00b4423 (experiment)

**Connection Type**

- Connect using RDP client**  
Download a file to use with your RDP client and retrieve your password.
- Connect using Fleet Manager**  
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#)

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following details:

<b>Public DNS</b> <input type="checkbox"/> ec2-54-197-166-238.compute-1.amazonaws.com	<b>User name</b> <input type="checkbox"/> Administrator
<b>Password</b> <a href="#">Get password</a>	

**Info** If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

### Step 3 :

In the "Instances" view, you can see the status of your instance(s) as they transition from "pending" to "running." You can monitor the status changes there. Once the instance is in the "running" state, it means it has successfully launched, and you can then proceed to connect to it or use it for your intended purposes.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances (with sub-links for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main content area has a header 'Instances (1) Info' with a search bar and filters for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. A 'Connect' button is also in the header. Below the header is a table with one row for 'Test1'. The table columns include Name, Instance ID, Instance state (which shows 'Running'), Instance type (t2.micro), Status check (2/2 checks passed), Alarm status (No alarms), and Availability Zone (us-east-1c). At the bottom of the main content area, there's a large text input field labeled 'Select an instance'.

## Practical 5:

### Terminate the launched window server instance from AWS EC2.

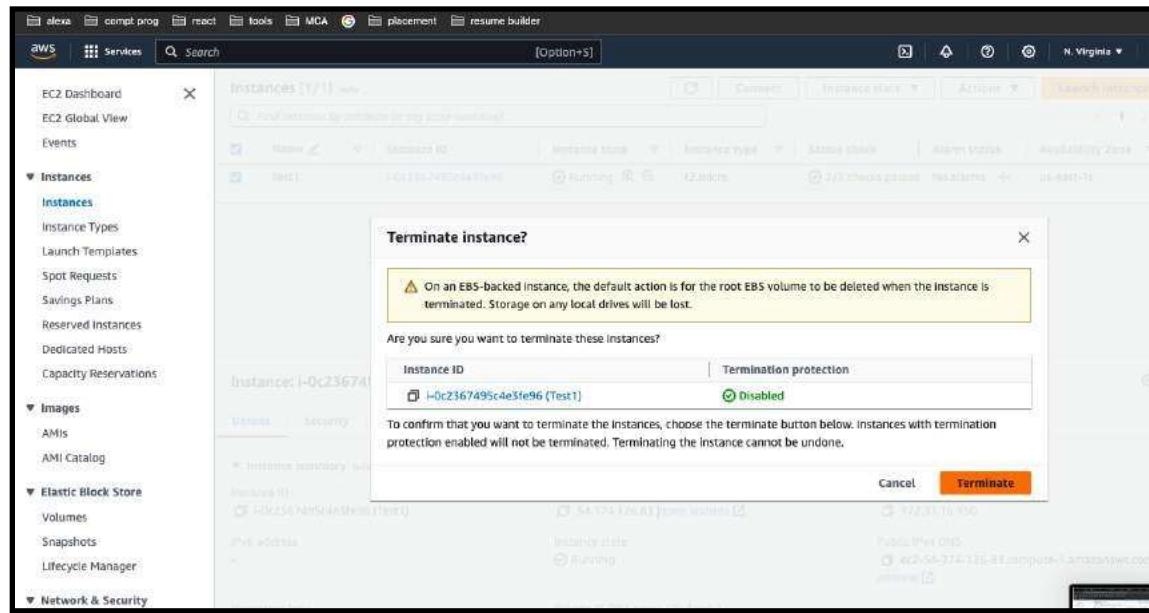
#### Step 1:

In the EC2 dashboard, click on "Instances" in the left navigation pane to view a list of your running instances.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation pane with various options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), 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), CloudShell, and Feedback. The main content area shows a table of instances. One instance is selected, named 'Test1' with the ID 'i-0c2367495c4e3fe96'. The instance is listed as 'Running' with the type 't2.micro'. At the top right of the table, there are buttons for 'Stop instance', 'Start instance', 'Reboot instance', 'Hibernate instance', and 'Actions'. The 'Actions' button is highlighted in orange. Below the table, there's a detailed view for the selected instance 'i-0c2367495c4e3fe96 (Test1)'. This view includes tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. Under the Details tab, there's an 'Instance summary' section with fields for Instance ID (i-0c2367495c4e3fe96 (Test1)), IPv6 address (empty), Hostname type (IP name: ip-172-31-16-150.ec2.internal), Public IPv4 address (54.174.126.83), Instance state (Running), and Private IP/DNS (Private IPv4 address 172.31.16.150, Public IPv4 DNS ec2-54-174-126-83.compute-1.amazonaws.com). A small screenshot of the Windows desktop is shown in the bottom right corner of the instance details.

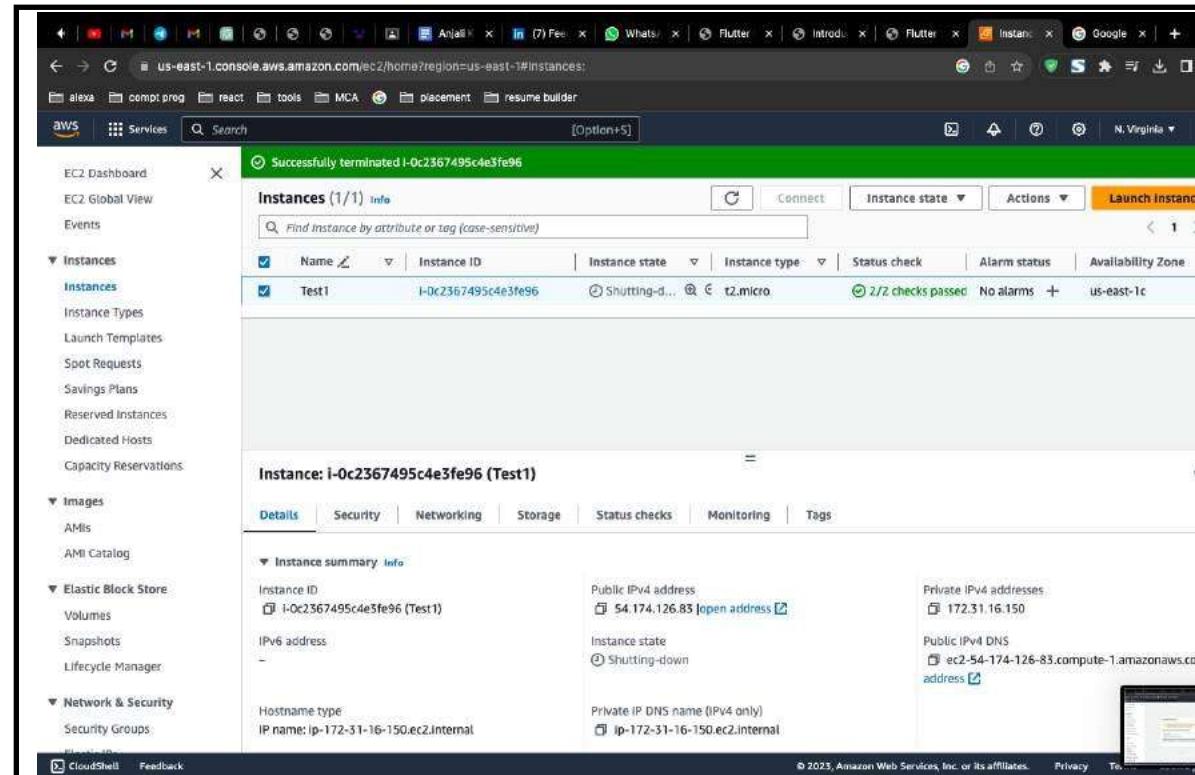
#### Step 2:

With the instance selected, click the "Actions" button at the top of the dashboard, and from the dropdown menu, select "Instance State" and then choose "Terminate."



### Step 3:

AWS will now initiate the termination process. The instance will first be stopped if it was running, and then it will be permanently deleted. This process may take a few minutes.

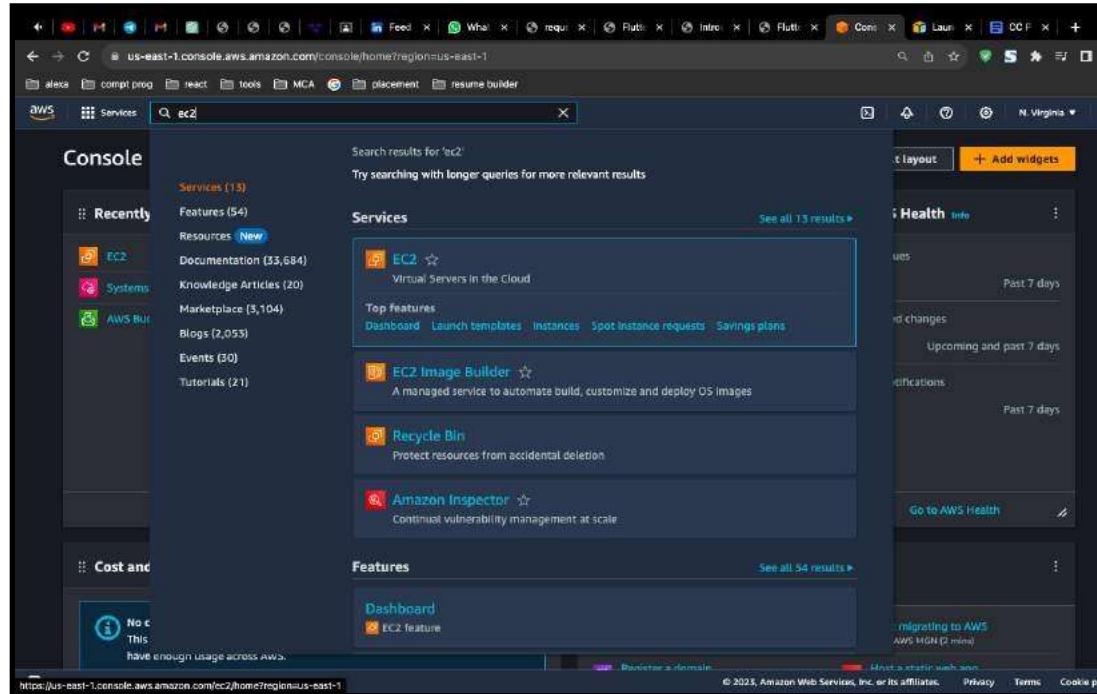


# **Practical 6:**

## **Write the steps to launch a Linux server by using AWS EC2.**

## **Step 1:**

Once logged in, navigate to the EC2 dashboard. You can do this by searching for "EC2" in the AWS Management Console's search bar or by selecting "Compute" and then "EC2" under the "Services" menu.

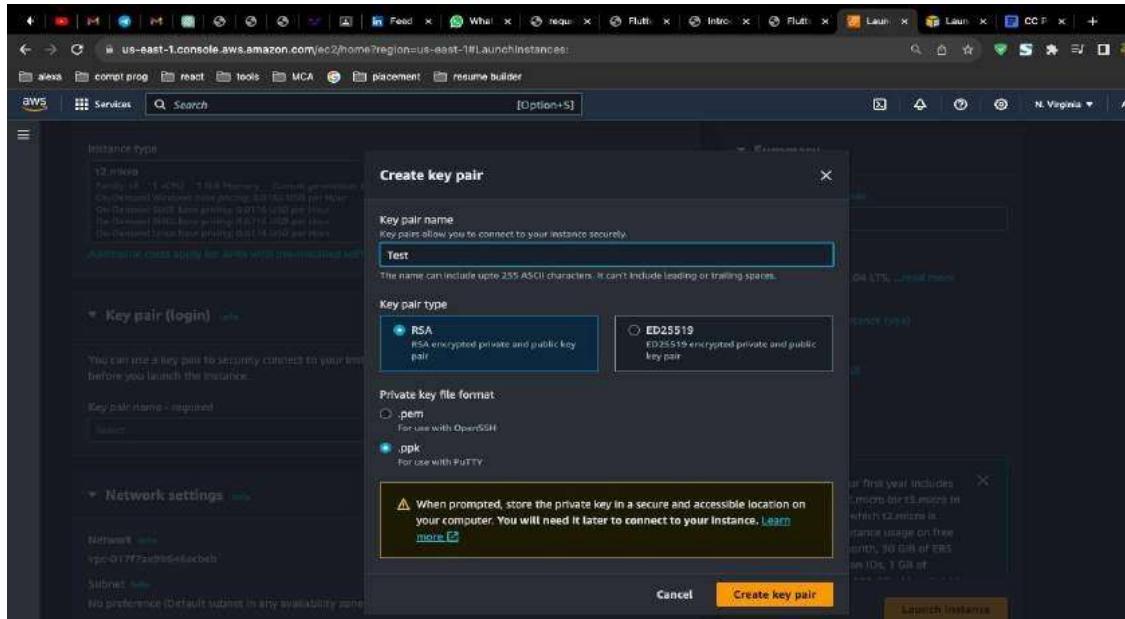


## Step 2 :

In the "Choose an Amazon Machine Image (AMI)" step, select a Linux AMI. AWS provides various Linux distributions and versions. Choose the one that suits your requirements (e.g., Amazon Linux, Ubuntu, CentOS, etc.).

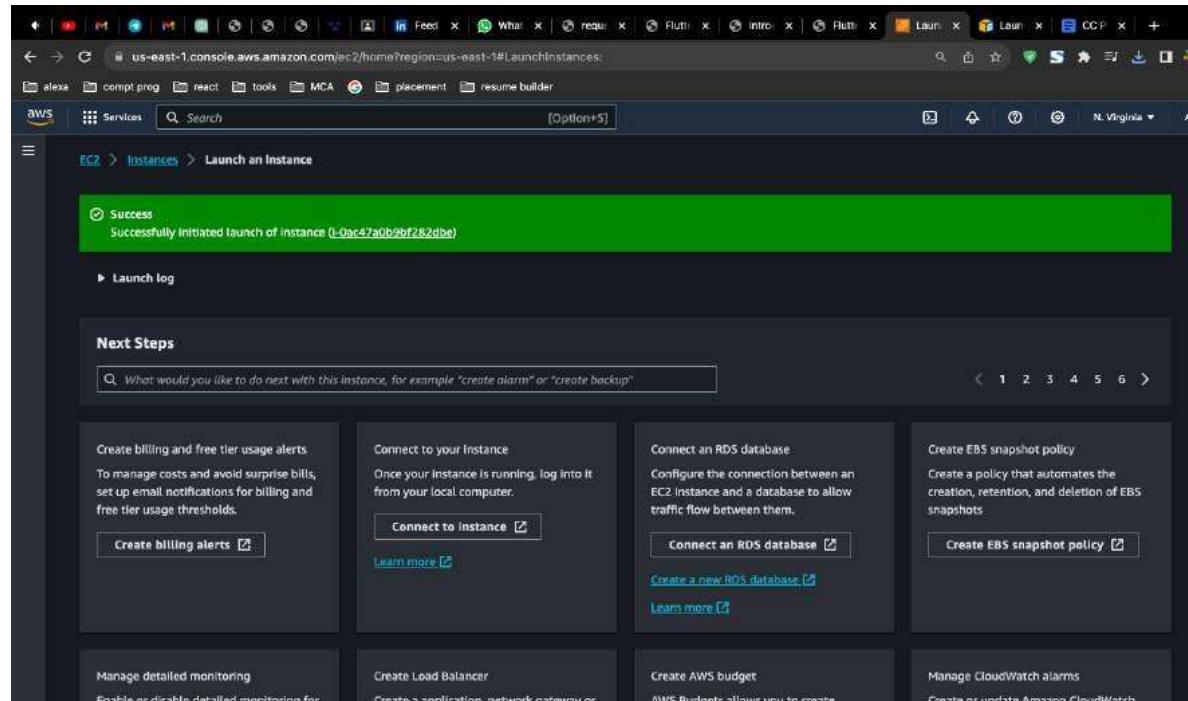
## Step 3:

If you haven't already created a key pair, you will be prompted to create one. This key pair is used to securely access your Linux server instance. Download and save the key pair (.pem file) in a secure location



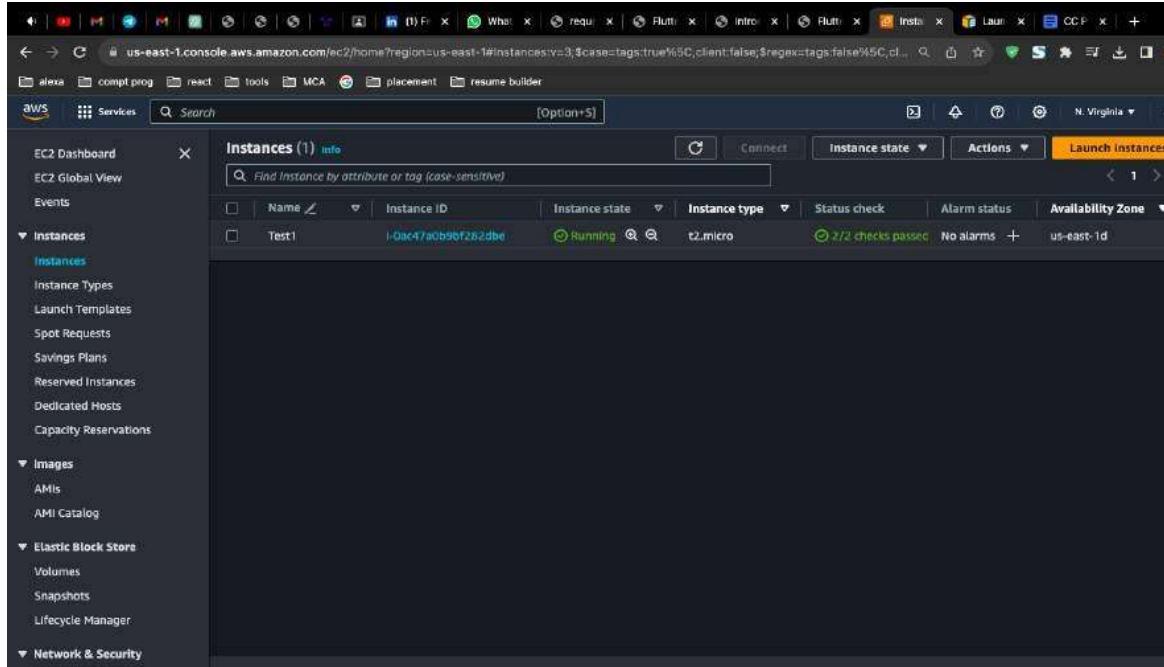
#### Step 4 :

After selecting or creating a key pair, click the "Launch Instances" button.



#### Step 5 :

You'll be taken to the "Instances" view, where you can see the status of your Linux server instance as it starts. Once the instance is in a "running" state, it means it has successfully launched, and you can proceed to access it via SSH.

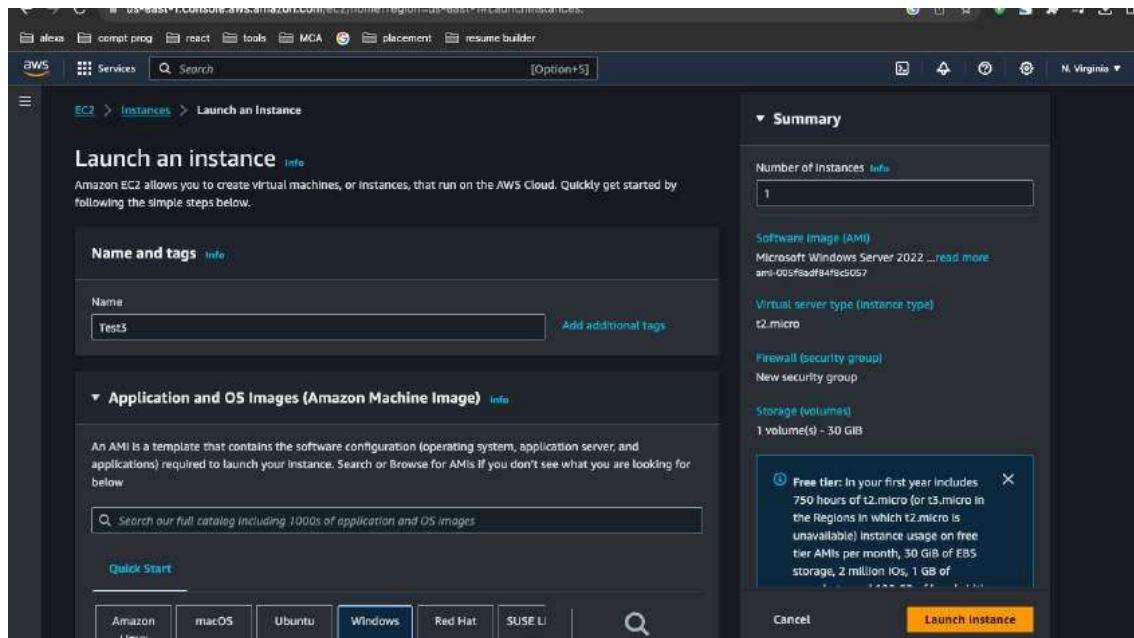


## Practical 7:

### Write the steps to connect with the window server by using AWS EC2.

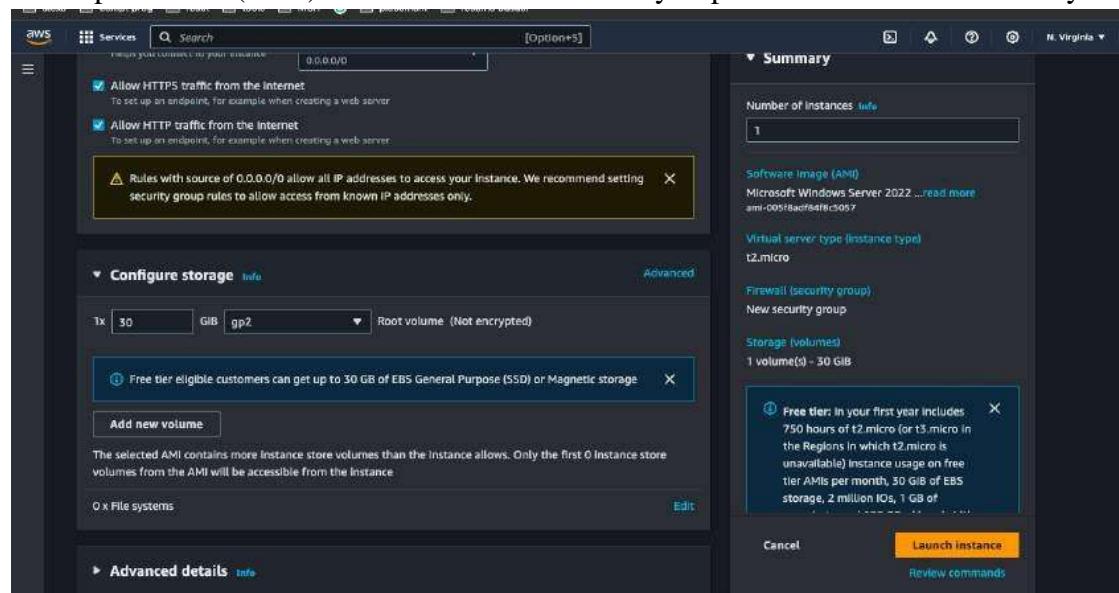
#### Step 1:

Create a Aws Ec2 Window Instance



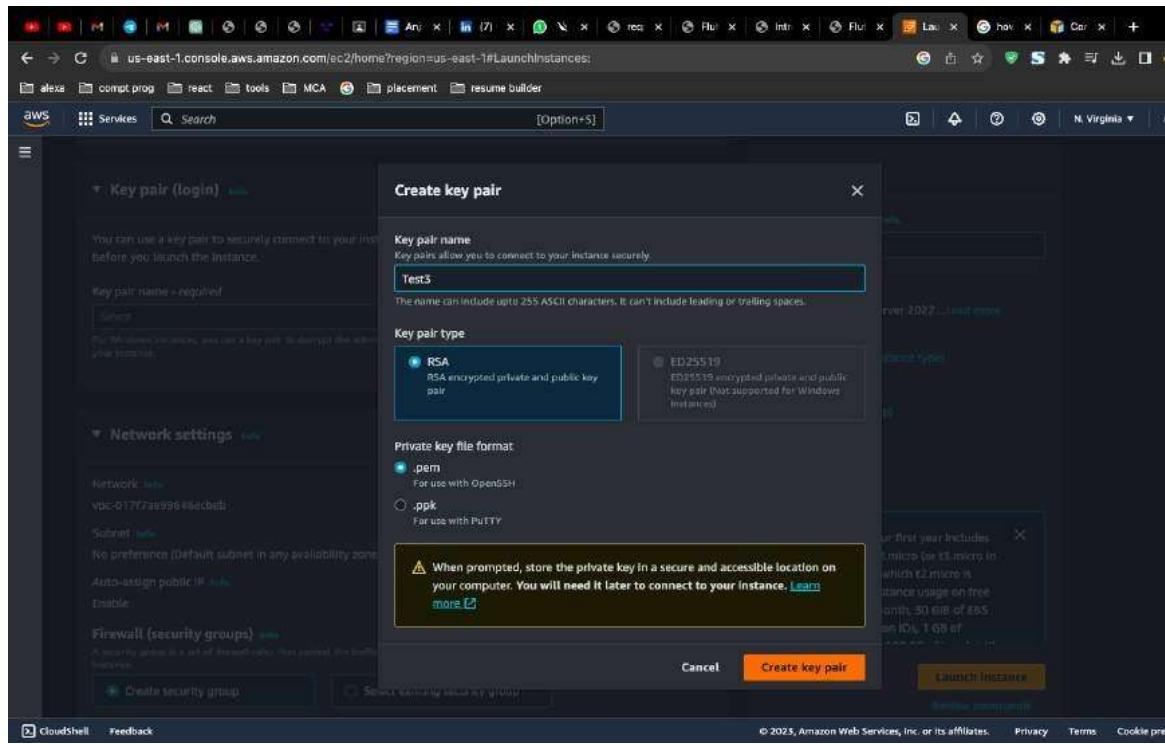
#### Step 2:

In the "Configure Security Group" step, you'll need to configure the security group rules. Security groups act as firewalls to control inbound and outbound traffic to your instance. Ensure that you allow Remote Desktop Protocol (RDP) for Windows instances if you plan to access them remotely.



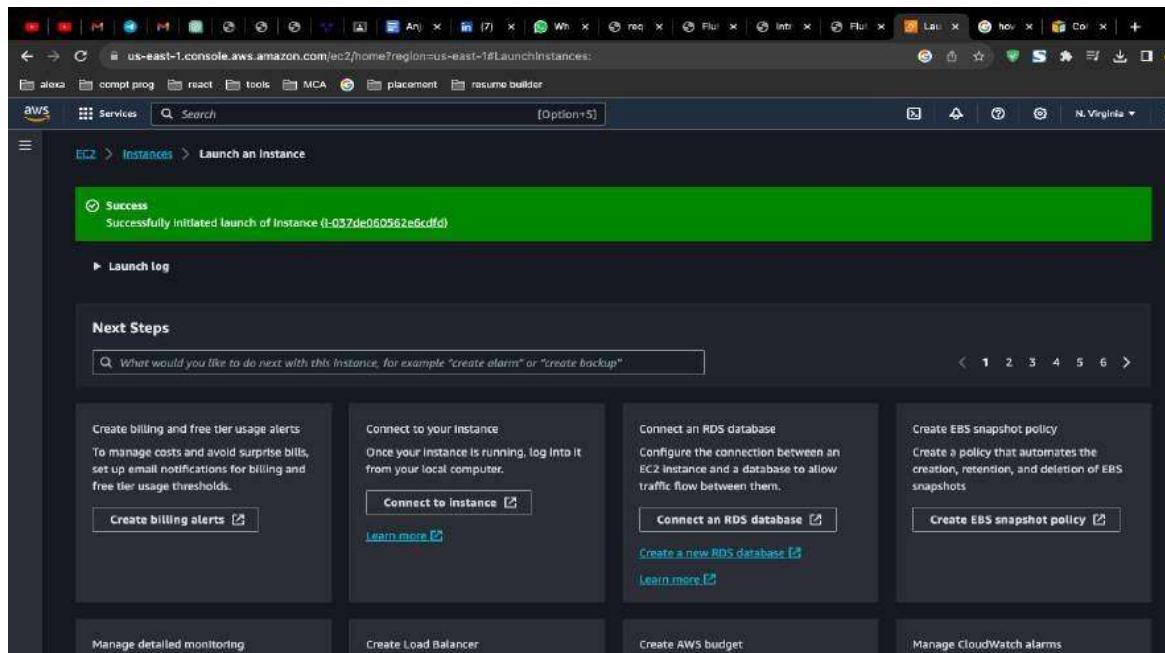
### Step 3:

Select The key pair for Instance



### Step 4 :

Launch Instance



The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), 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, and Security Groups. The main content area has a header 'Instances (1) Info' with a search bar and filters for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. A table lists one instance: 'Test3' (Instance ID: i-037de060562e6cdff, State: Pending, Type: t2.micro, No alarms, us-east-1c). Below the table is a section titled 'Select an instance' with a dropdown menu. At the bottom, there are links for CloudShell and Feedback.

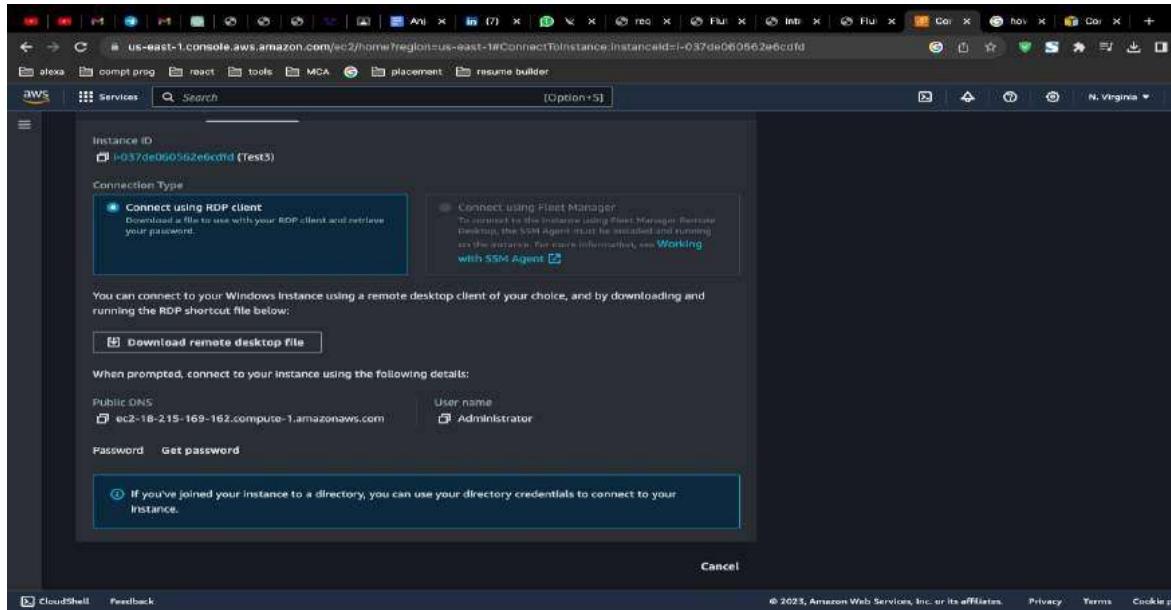
## Step 5 :

### Check Instance Summary

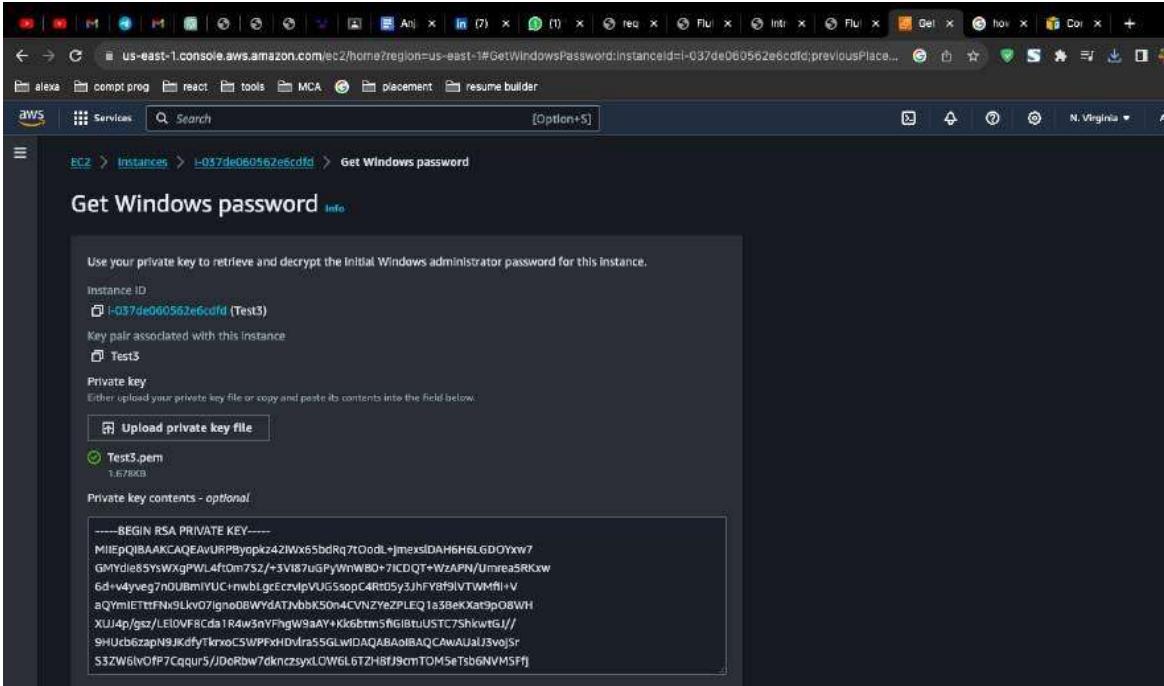
The screenshot shows the AWS EC2 Instance Details page for instance 'Test3'. The left sidebar is identical to the previous screenshot. The main content area shows the 'Instance summary for i-037de060562e6cdff (Test3)' with the message 'Updated less than a minute ago'. It displays various details: Instance ID (i-037de060562e6cdff (Test3)), Public IPv4 address (18.215.169.162), Instance state (Running), Private IP DNS name (IPv4 only) (ip-172-51-31-179.ec2.internal), Instance type (t2.micro), VPC ID (vpc-012f7ae99646ecbeb), Subnet ID (subnet-0944dd8d0c7f2ac2), and other fields like Auto-assigned IP address (18.215.169.162 [Public IP]), IAM Role (None), and IMDSv2 (Optional). Below the summary, tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags are visible. At the bottom, there's a 'Learn more' link for AWS Compute Optimizer.

## Step 6 :

Open your RDP client and configure a new connection with the public IP address or public DNS name of your Windows Server instance. You may also need to specify the username you want to use for the remote desktop session. By default, this is usually "Administrator."



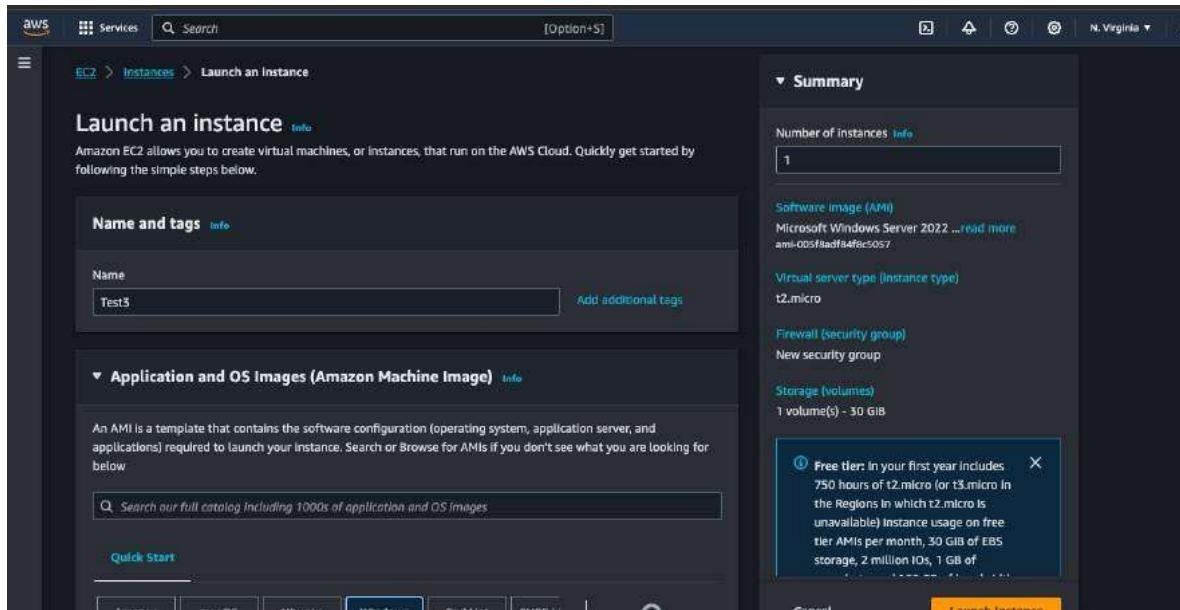
**Step 7 :** After configuring the connection, click "Connect" or "Connect" in your RDP client. You will be prompted to enter the administrator's username and password. If you haven't changed the password, you can retrieve it from the AWS Management Console.



## Practical 8:

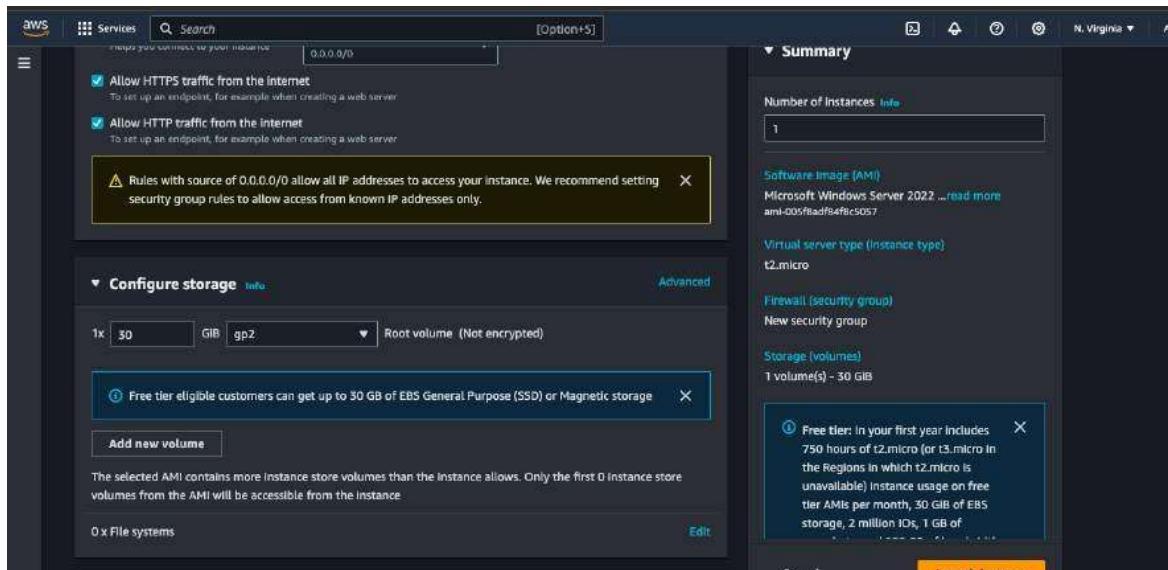
### How to launch a website on a windows server using EC2.

#### Step 1: Create a Instance



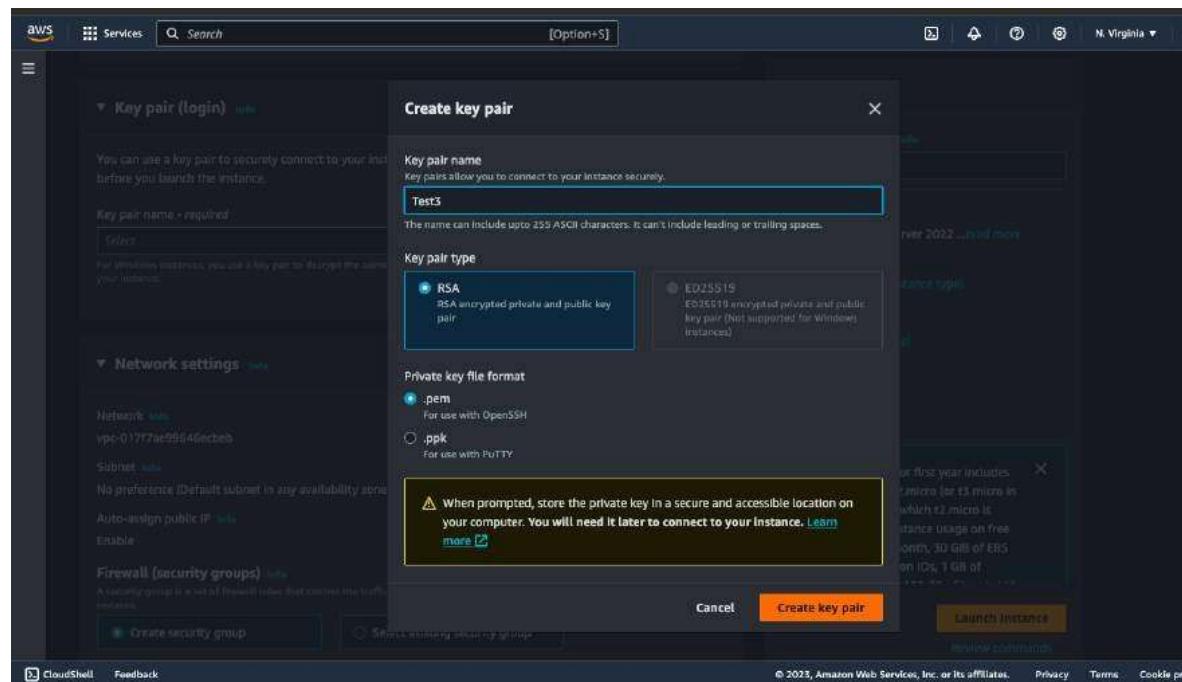
#### Step 2 :

In the "Configure Security Group" step, you'll need to configure the security group rules. Security groups act as firewalls to control inbound and outbound traffic to your instance. Ensure that you allow Remote Desktop Protocol (RDP) for Windows instances if you plan to access them remotely.



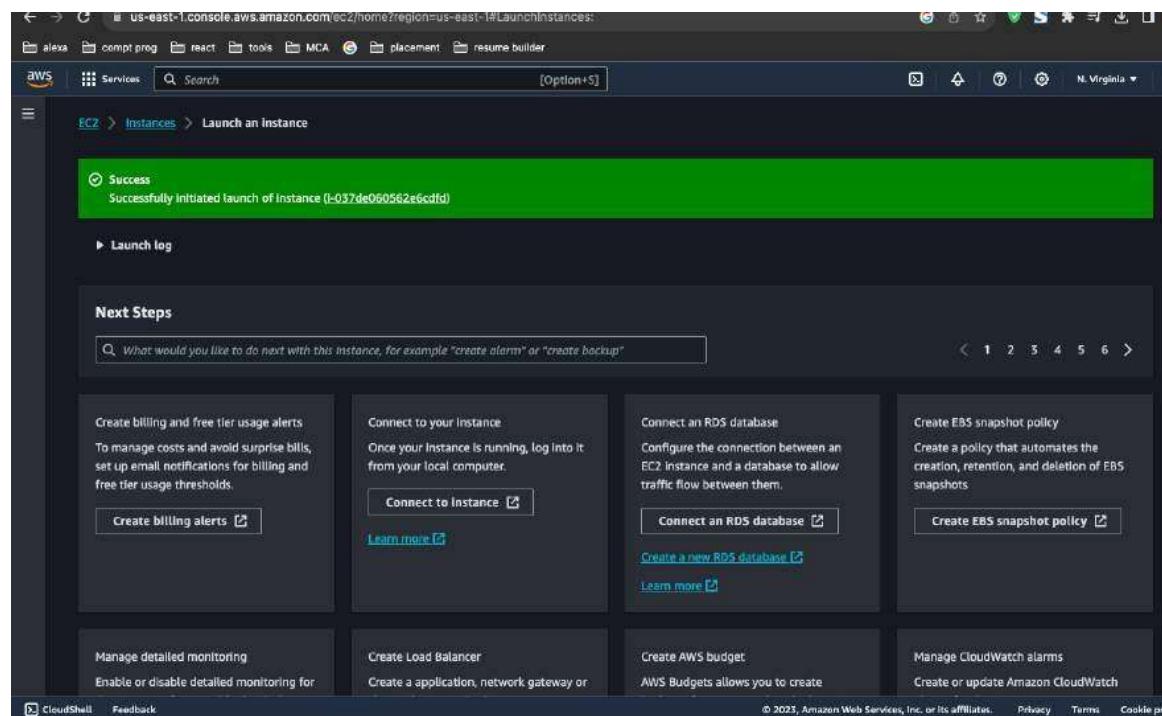
## Step 3:

Create a key pair for instance.



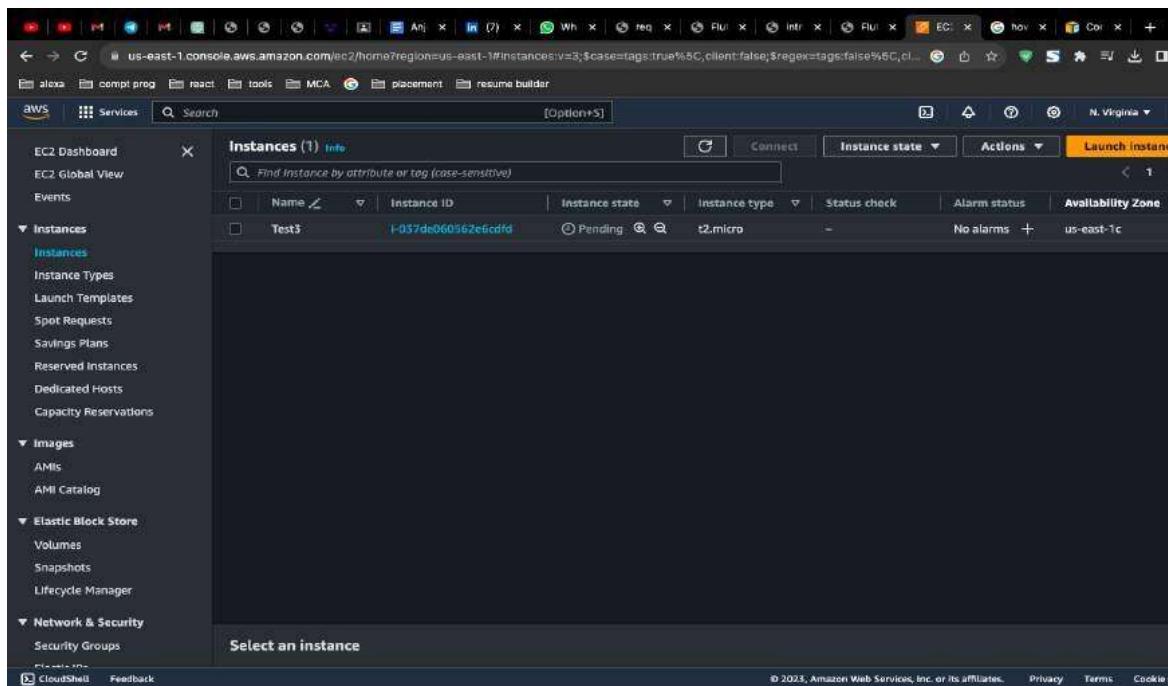
## Step 4:

Launch the instance



## Step 5 :

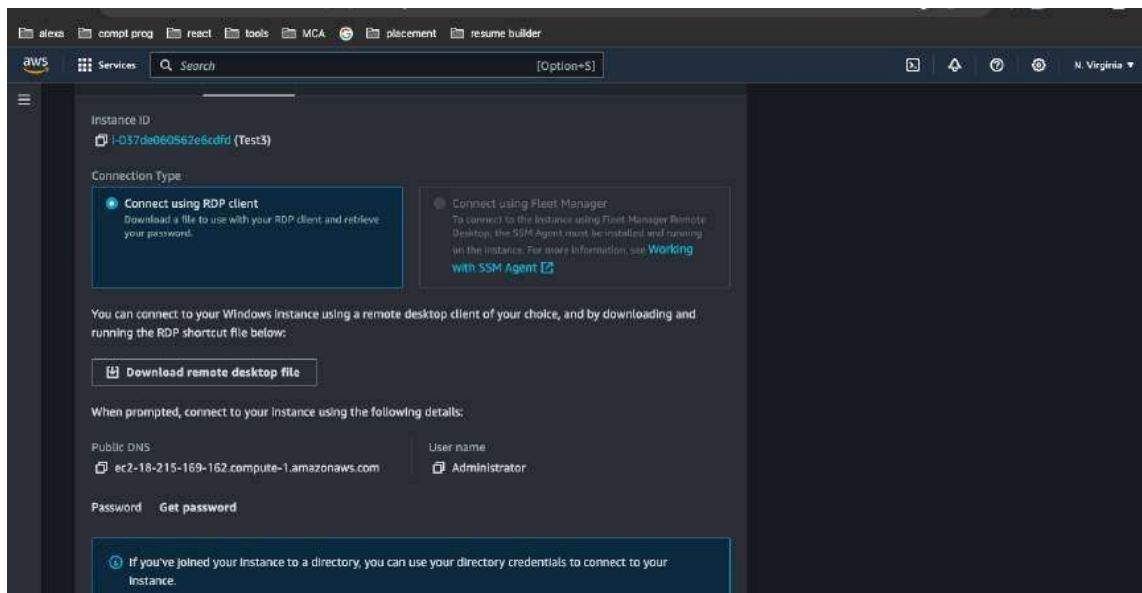
Launch the instance



The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), 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), and Network & Security (Security Groups). The main content area has a header 'Instances (1) info' with a search bar. A table lists one instance: 'Test3' with Instance ID 'i-057de060562e6cdff', State 'Pending', Type 't2.micro', Status check ' - ', Alarm status 'No alarms', and Availability Zone 'us-east-1c'. At the bottom, there's a 'Select an instance' dropdown and a note about CloudShell and Feedback.

## Step 6 :

Open your RDP client and configure a new connection with the public IP address or public DNS name of your Windows Server instance. You may also need to specify the username you want to use for the remote desktop session. By default, this is usually "Administrator."



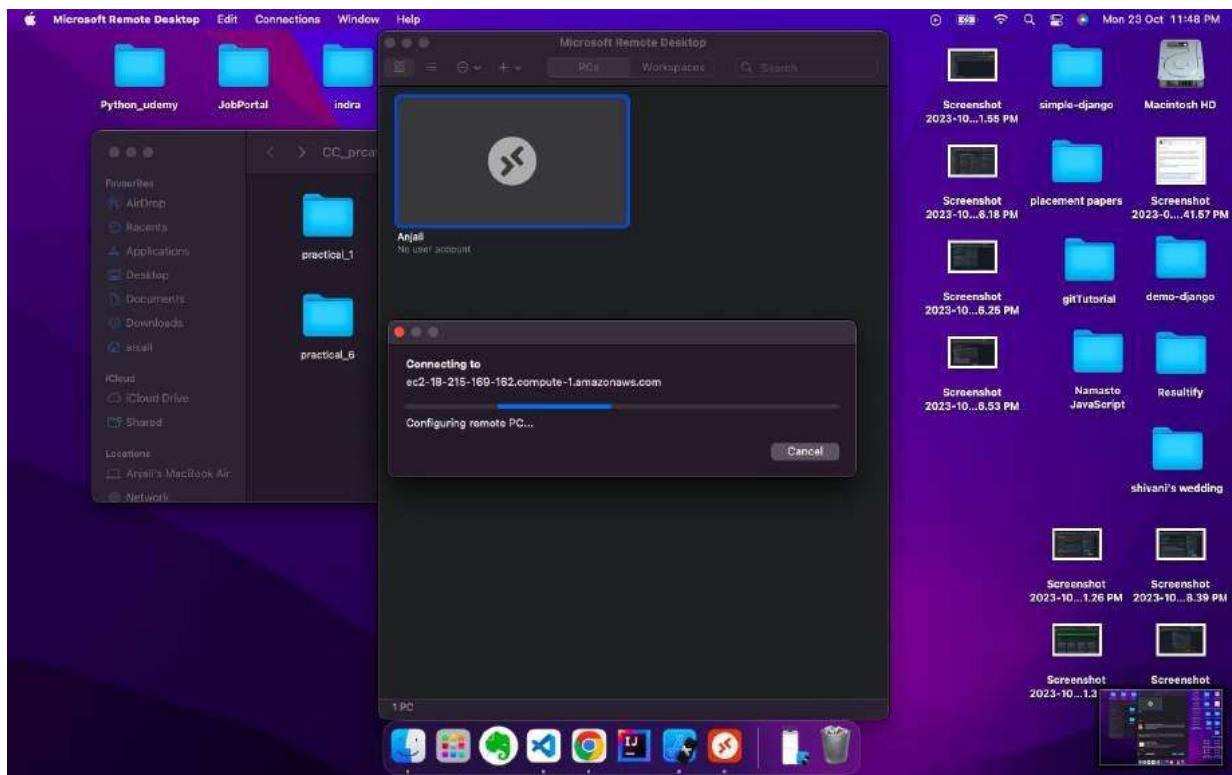
This screenshot shows the 'Connect using RDP client' details for instance 'Test3'. It includes a download link for a remote desktop file ('Download remote desktop file'), connection details ('Public DNS: ec2-18-215-169-162.compute-1.amazonaws.com', 'User name: Administrator'), and a note about directory credentials ('If you've joined your instance to a directory, you can use your directory credentials to connect to your instance').

## Step 7:

After configuring the connection, click "Connect" or "Connect" in your RDP client. You will be prompted to enter the administrator's username and password. If you haven't changed the password, you can retrieve it from the AWS Management Console.

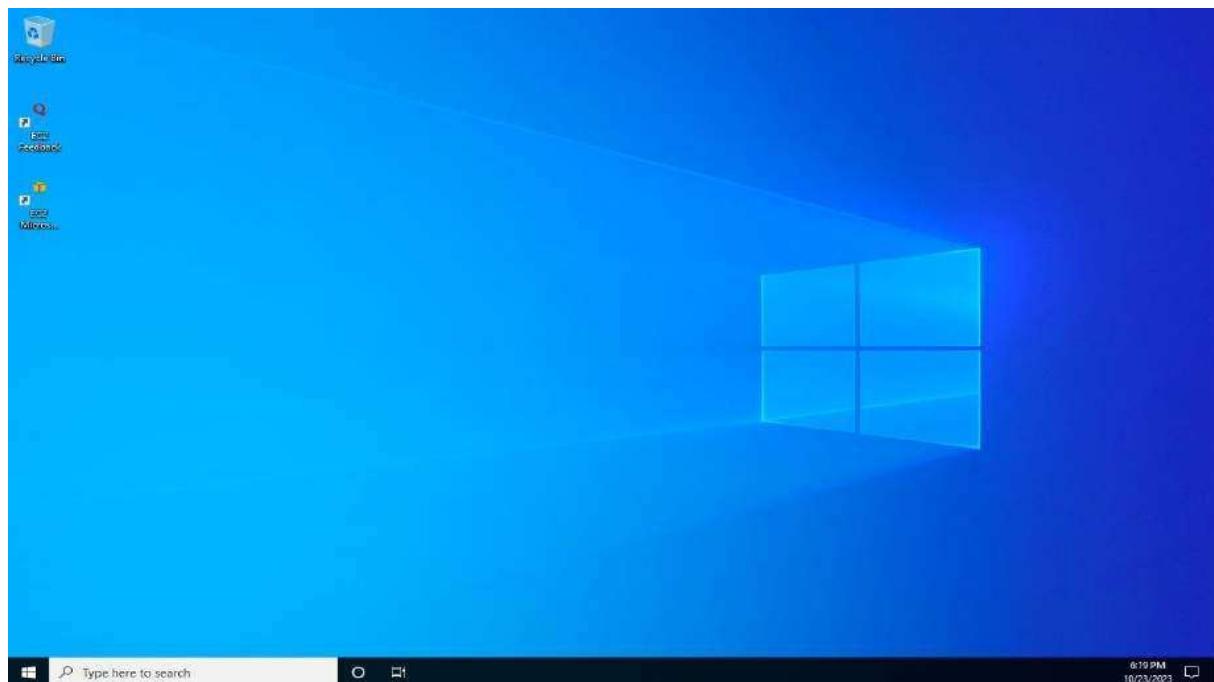
The screenshot shows the AWS Management Console with the URL <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#GetWindowsPassword:instanceId=i-037de060562e6ccfd;previousPlace...>. The page title is "Get Windows password". It displays the Instance ID (i-037de060562e6ccfd (Test3)), Key pair associated with this instance (Test3), and a private key file uploaded (Test3.pem). The private key contents are shown in a large text area. At the bottom, there are "Cancel" and "Decrypt password" buttons.





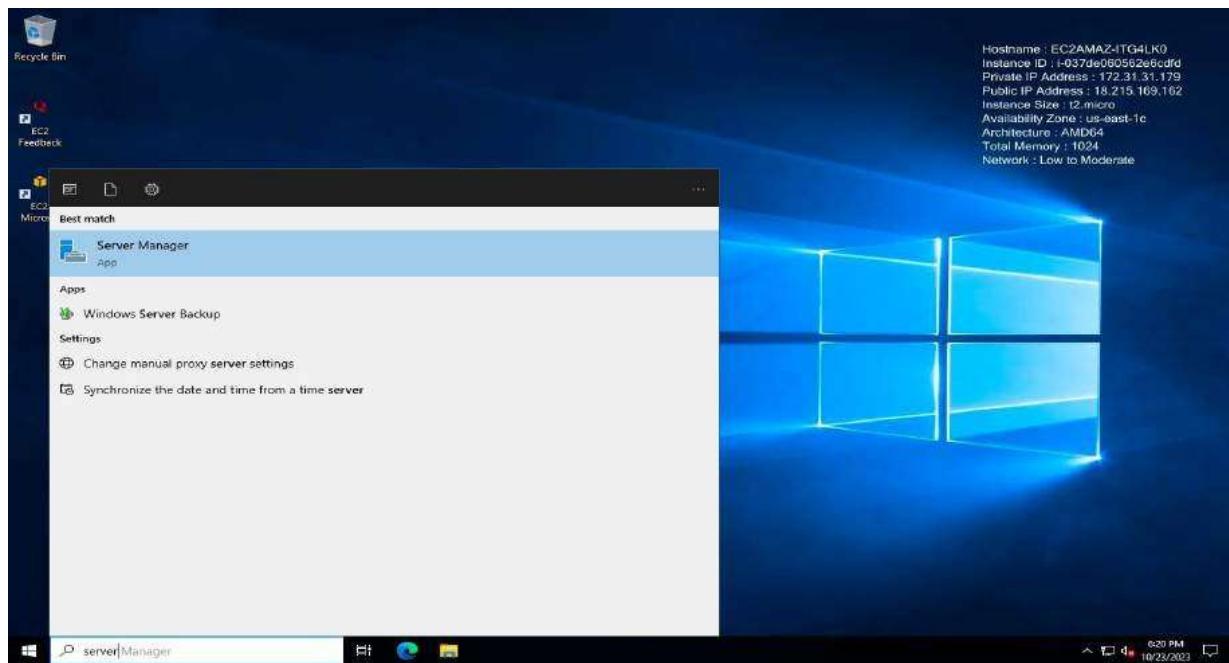
### Step 8:

To host a website, you'll need web server software. You can use Microsoft Internet Information Services (IIS) as a common choice for hosting websites on Windows Server. Follow these steps to install and configure IIS:



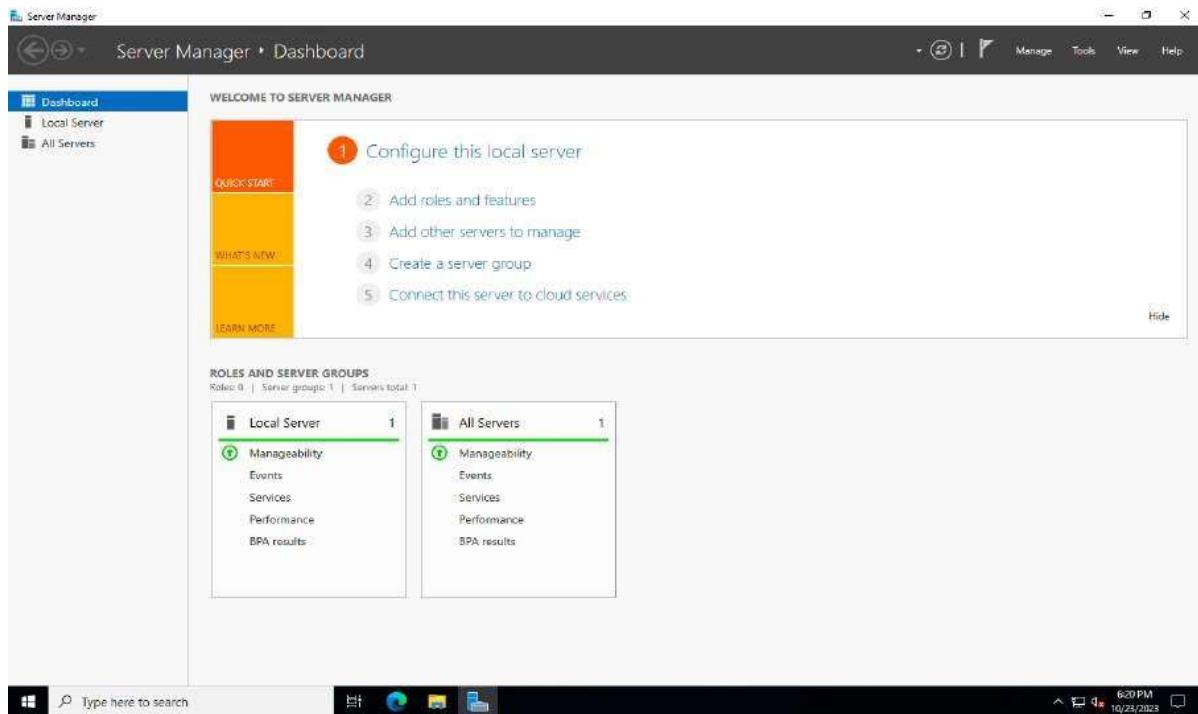
## Step 9 :

In your Windows Server instance, open "Server Manager."



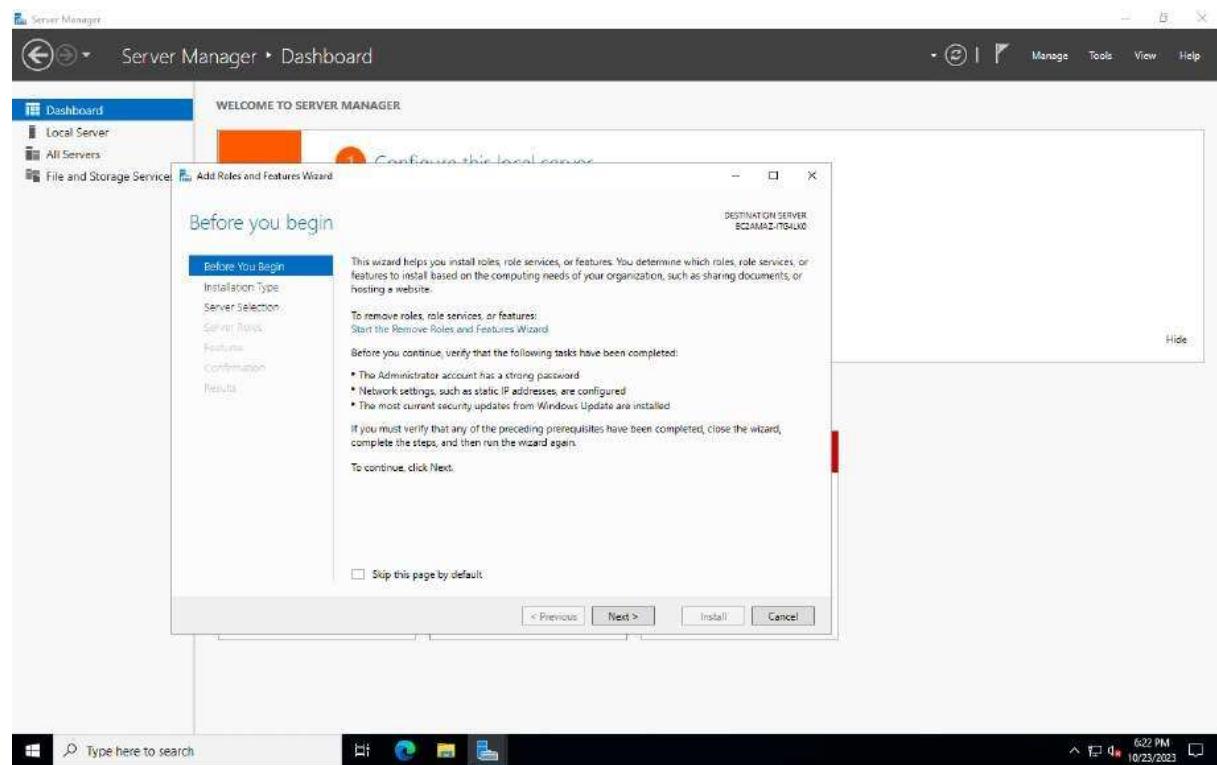
## Step 10 :

Click on "Add roles and features" and follow the wizard to install the Web Server (IIS) role.



## Step 11 :

Once IIS is installed, you can configure your website by creating a new site, specifying the content directory, and setting up bindings (e.g., domain names or IP addresses).



## Practical 9:

### Terminate the launched Linux server instance from AWS EC2.

#### Step 1:

In the EC2 dashboard, click on "Instances" in the left navigation pane to view a list of your running instances. Locate the Linux server instance you want to terminate.

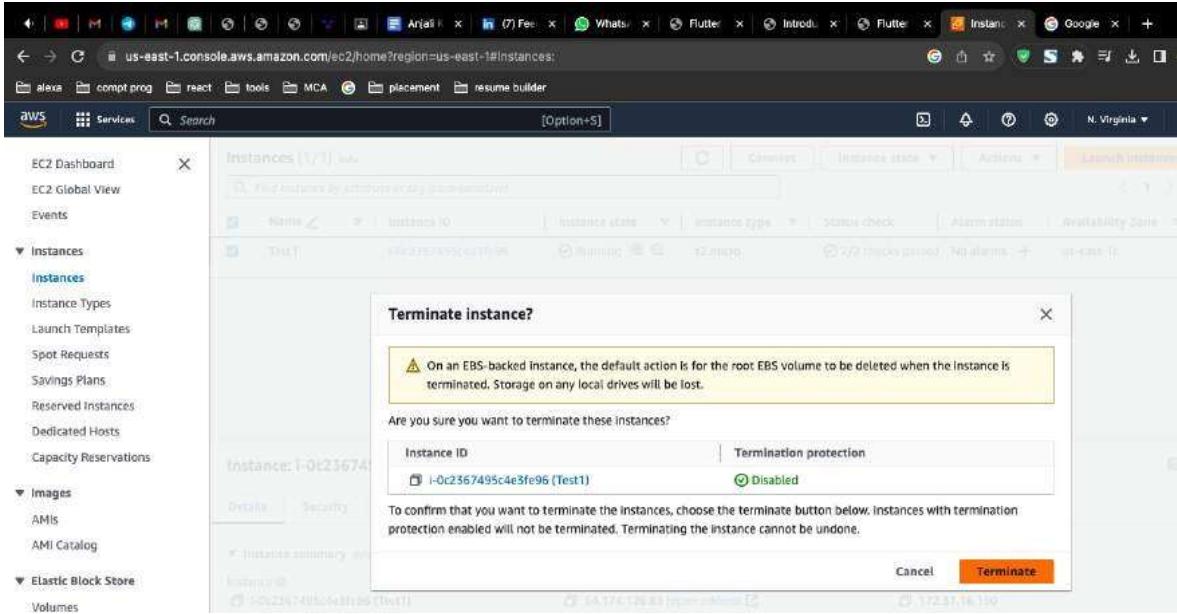
#### Step 2:

Click the checkbox next to the Linux server instance you want to terminate. It will become selected.

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, 'Instances' is selected. In the main content area, the 'Instances (1/1)' table has one row for 'Test1' (Instance ID: i-0c2367495c4e3fe96). The 'Actions' column for this instance contains several buttons: 'Stop Instance', 'Start Instance', 'Reboot Instance', 'Hibernate instance', and 'Terminate Instance'. The 'Terminate Instance' button is highlighted with a blue border. Below the table, the details for 'Instance: i-0c2367495c4e3fe96 (Test1)' are shown, including its public IPv4 address (54.174.126.83) and private IPv4 address (172.31.16.150).

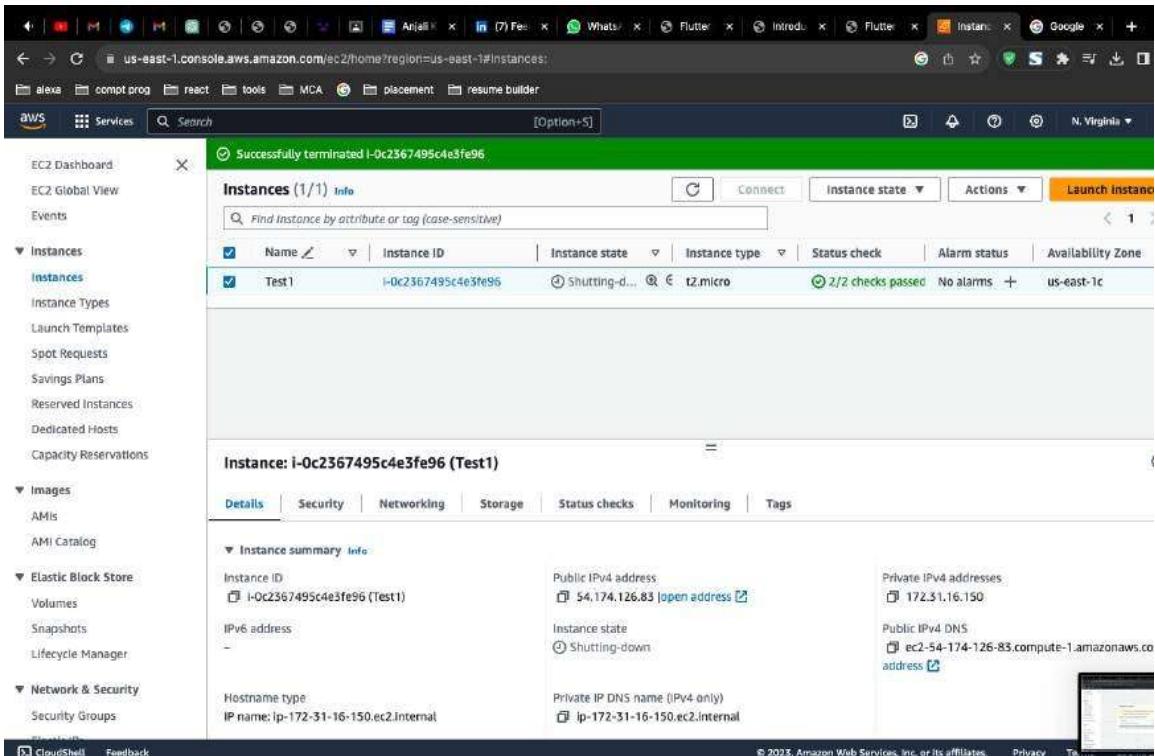
#### Step 3 :

With the instance selected, click the "Actions" button at the top of the dashboard, and from the dropdown menu, select "Instance State" and then choose "Terminate."



#### Step 4 :

AWS will now initiate the termination process. The instance will be stopped if it was running, and then it will be permanently deleted. This process may take a few minutes.

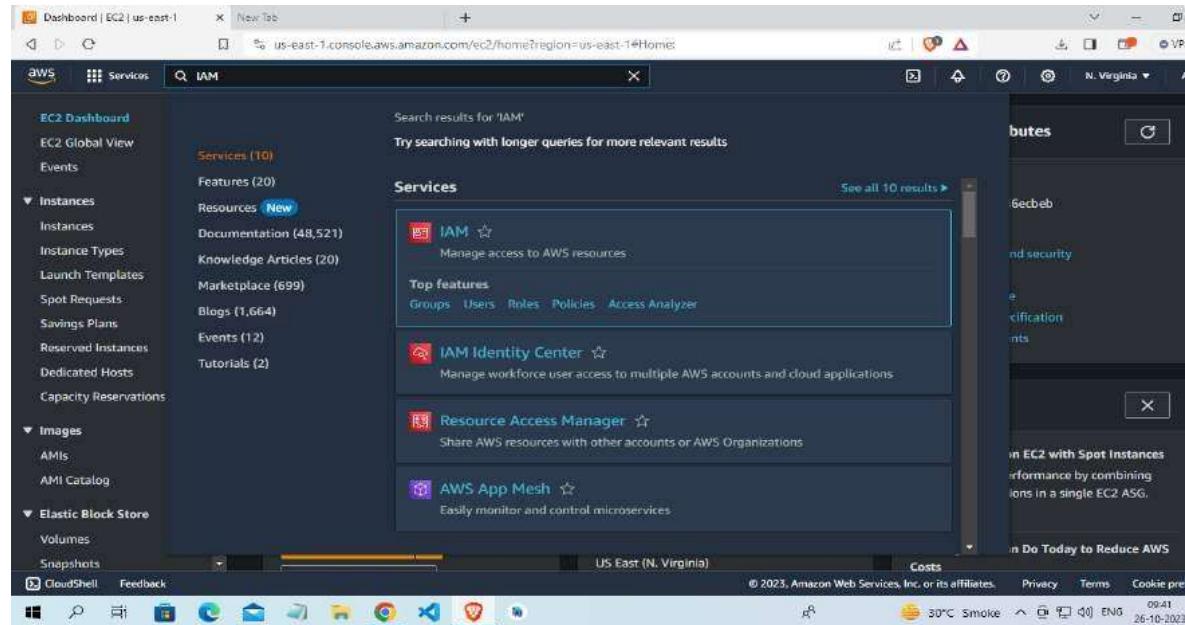


## Practical 10:

### How to create a IAM (Identity and Access management) user.

#### Step 1 :

Once logged in, navigate to the IAM (Identity and Access Management) Console. You can do this by searching for "IAM" in the AWS Management Console's search bar or by selecting "Security, Identity, & Compliance" and then "IAM" under the "Services" menu.



#### Step 2:

In the IAM dashboard, click on "Users" in the left navigation pane to view the list of existing IAM users.

The screenshot shows the AWS Identity and Access Management (IAM) dashboard. On the left, a navigation sidebar lists various IAM management options like Access management, User groups, Roles, Policies, and Access reports. The main area displays 'Security recommendations' with two items: 'Add MFA for root user' (yellow warning icon) and 'Root user has no active access keys' (green success icon). Below this is a summary of 'IAM resources' with counts: User groups (0), Users (0), Roles (2), Policies (0), and Identity providers (0). To the right, there's a section for the 'AWS Account' with details like Account ID (804008392614), Account Alias (Create), and Sign-in URL (https://804008392614.signin.aws.amazon.com/console). A 'Quick Links' section includes a link to 'My security credentials'. The bottom of the screen shows the Windows taskbar with various pinned icons.

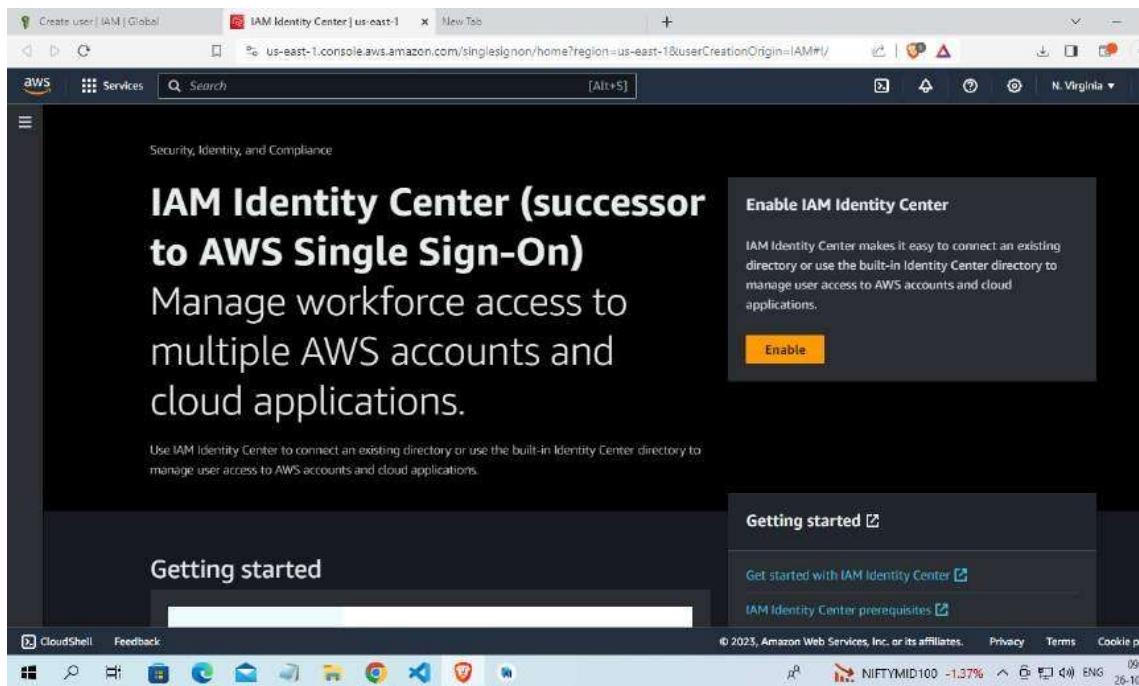
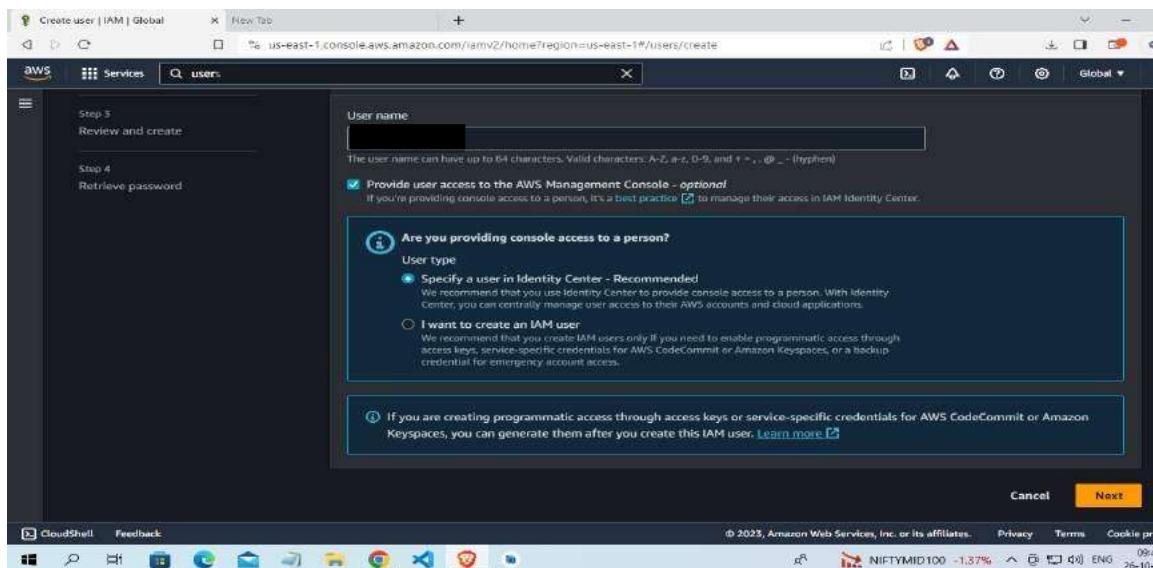
### Step 3 :

To create a new IAM user, click the "Add user" button at the top of the dashboard.

The screenshot shows the 'Users' page within the IAM service. The left sidebar is identical to the dashboard, with 'Users' currently selected. The main content area is titled 'Users (0) Info' and contains the text 'An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.' Below this is a search bar and a table header with columns: User name, Path, Group, Last activity, MFA, and Password age. A message 'No resources to display' is centered in the table body. The bottom of the screen shows the Windows taskbar.

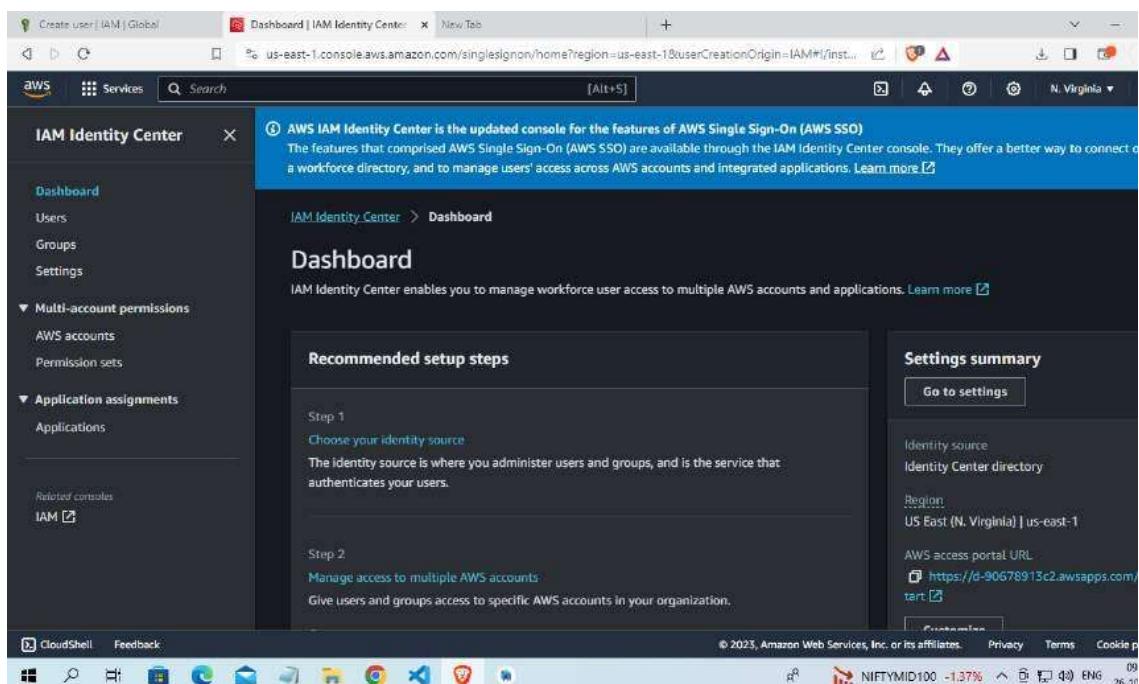
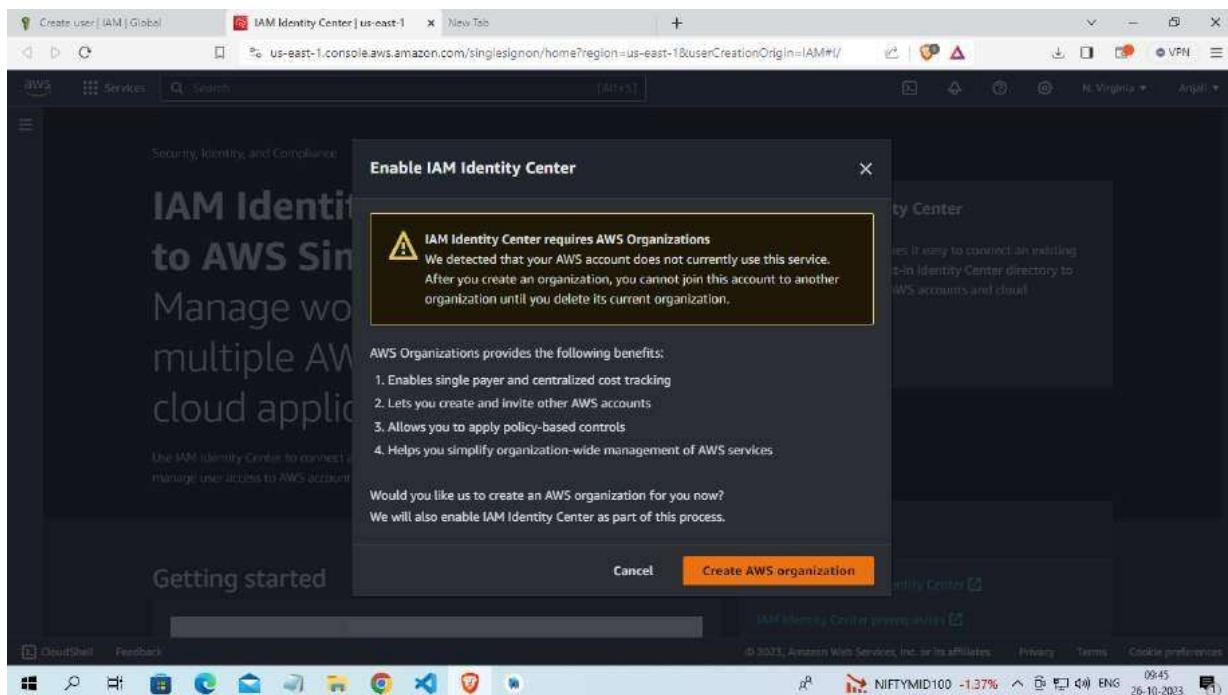
## Step 4 :

User name: Enter a unique name for the IAM user.



## Step 5 :

After the user is successfully created, you'll see a confirmation page. This page provides important information, such as the user's access key and secret access key (if you selected "Programmatic access"). Make sure to download and securely store the access keys because they will only be displayed once.



## Practical 11:

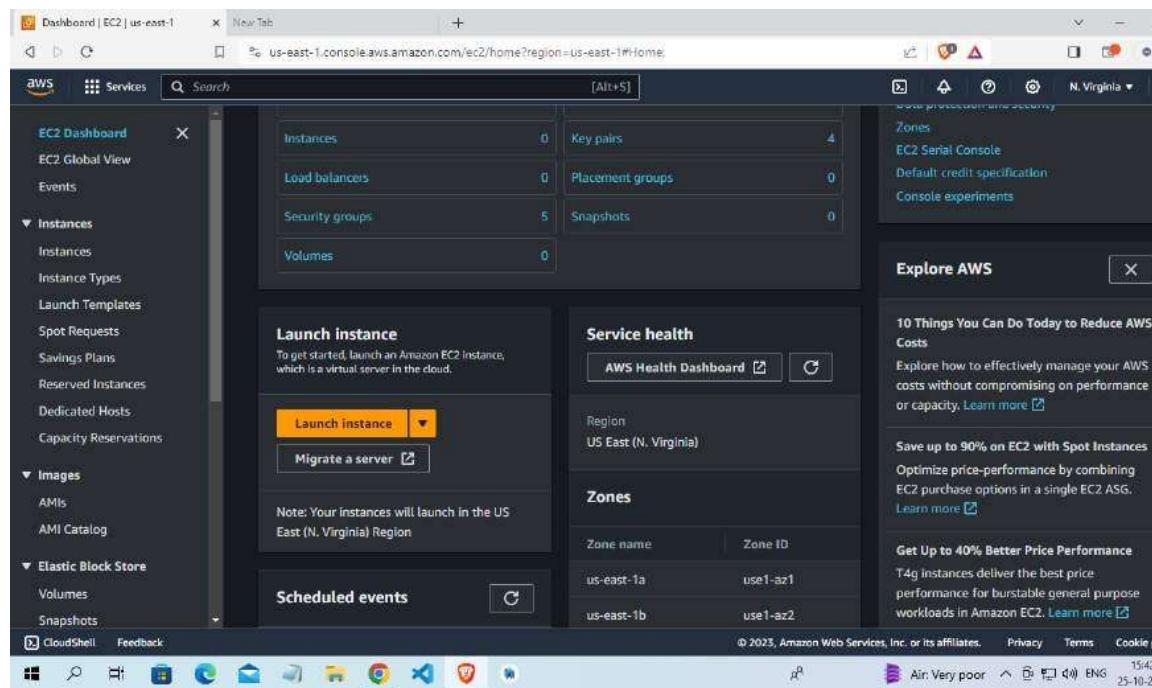
### How to connect with a launched instance of Linux (putty & putty gen software).

The objective of the practical steps I provided for connecting to a Linux server using PuTTY and PuTTYgen is to enable users to securely access and manage a remote Linux server from a Windows machine. Here's a breakdown of the objectives:

- Connect to a Remote Server: The primary objective is to establish a secure connection to a remote Linux server. This is often necessary for system administrators, developers, or users who need to perform tasks on a server located in a different location.
- Security: By using SSH (Secure Shell) and, optionally, generating an SSH key pair, the objective is to ensure a secure and encrypted connection. The SSH key pair enhances security by eliminating the need for passwords, making it difficult for unauthorized users to gain access.
- Manage the Linux Server: Once connected, the user can interact with the Linux server through the terminal window provided by PuTTY. This allows users to perform various tasks, such as software installation, file management, server configuration, and troubleshooting.

#### Step 1:

Open EC2 Dashboard: Click on "Services" in the top navigation and select "EC2" under the Compute section. This will take you to the EC2 Dashboard.



## Step 2:

Choose an Amazon Machine Image (AMI):

Select the Linux distribution you want to use (e.g., Amazon Linux, Ubuntu, CentOS, etc.).

Choose the AMI and click "Select."

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags** Info

Name  
Test5 Add additional tags

**Application and OS Images (Amazon Machine Image)** Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

**Quick Start**

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Li... CloudShell Feedback

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.2.2... read more  
ami-0dbc3d7c846c8516

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Free tier: In your first year X

Cancel **Launch instance** Review commands

**Amazon Machine Image (AMI)**

Ubuntu Server 22.04 LTS (HVM, SSD Volume Type)  
ami-0fc5d935ebf8bc3bc (64-bit (x86)) / ami-01648516ec7fa705 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible

Description  
Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-09-19

Architecture  
64-bit (x86) AMI ID ami-0fc5d935ebf8bc3bc Verified provider

**Instance type** Info

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ... read more  
ami-0fc5d935ebf8bc3bc

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Free tier: In your first year X

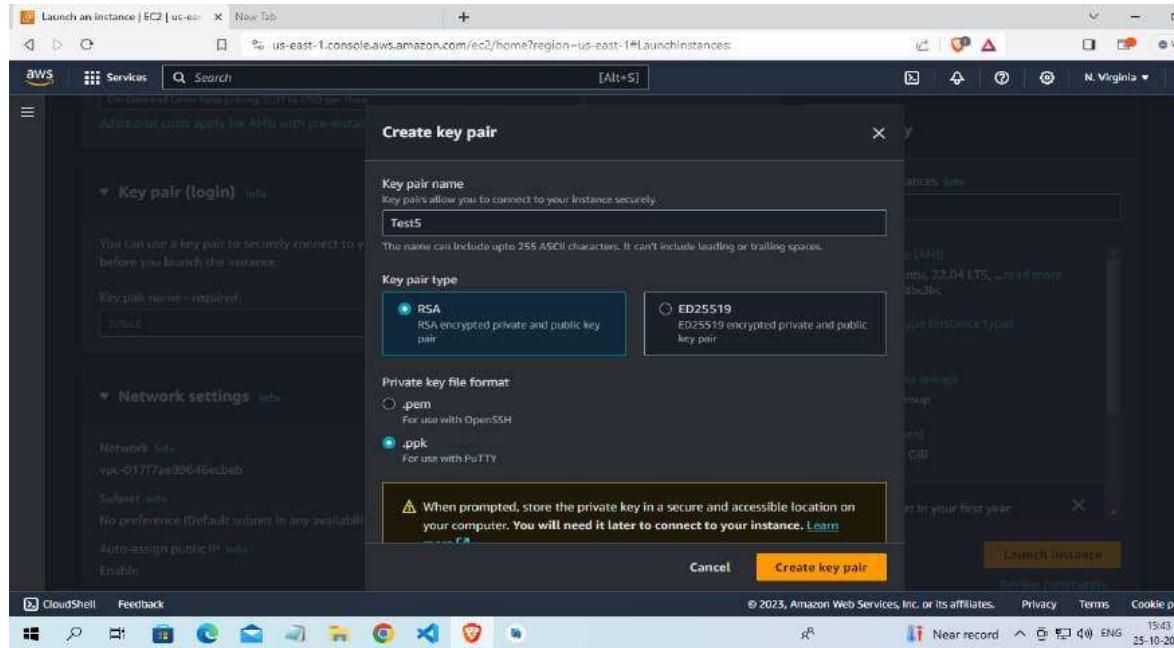
Cancel **Launch instance** Review commands

### Step 3:

Select Key Pair or Create a New Key Pair:

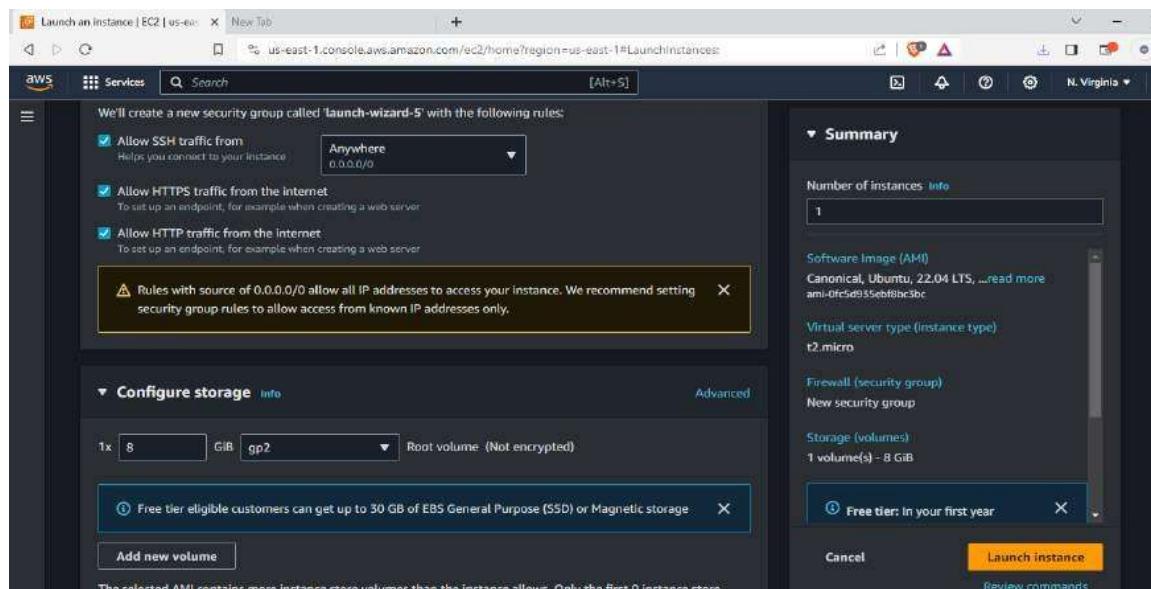
You'll need a key pair to securely connect to your instance. If you already have one, select it. If not, create a new key pair.

Download the key pair (.ppk) and save it in a secure location.



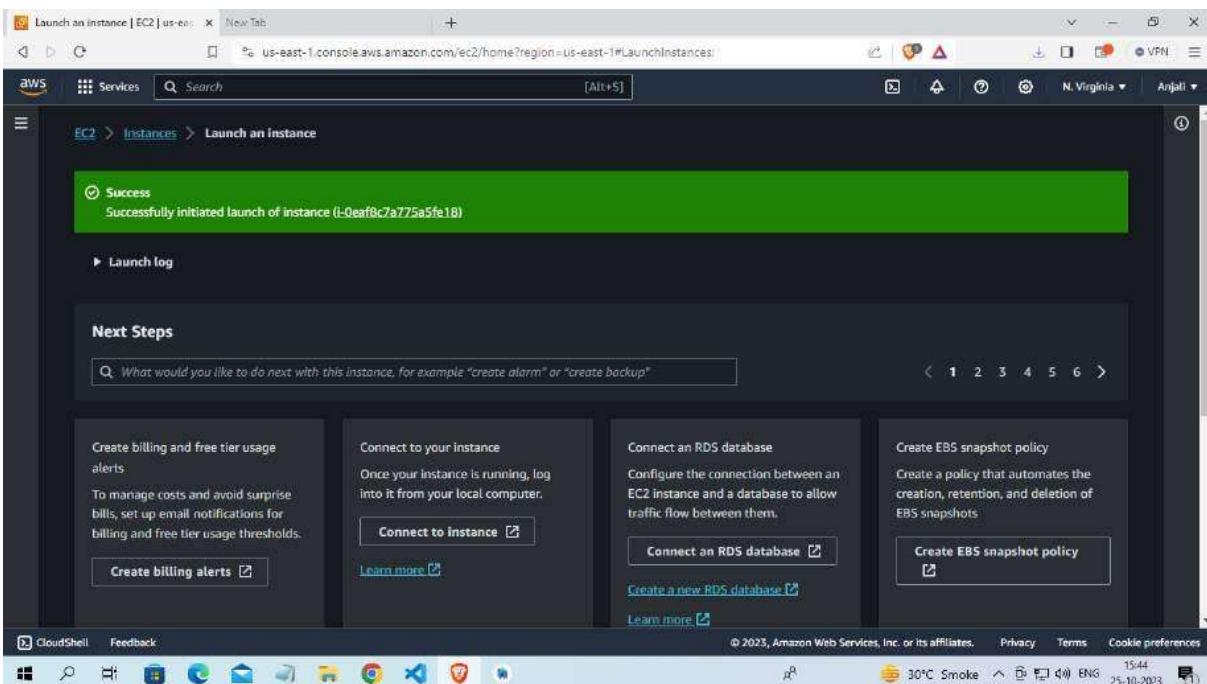
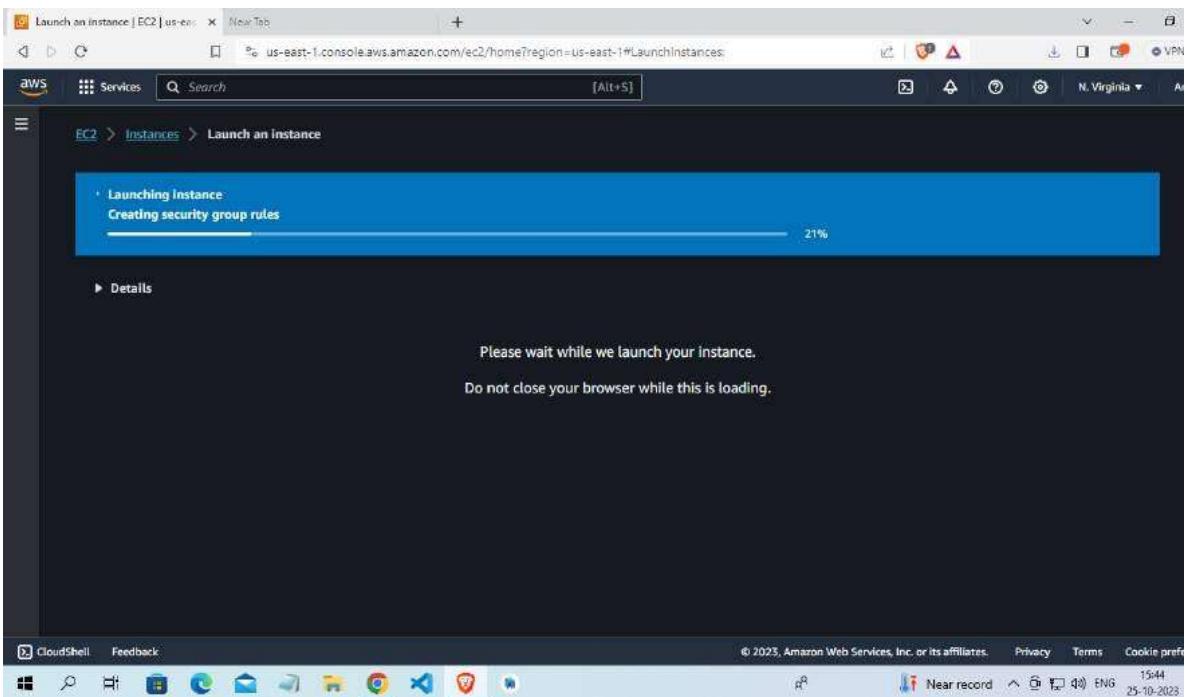
### Step 4:

Review your instance



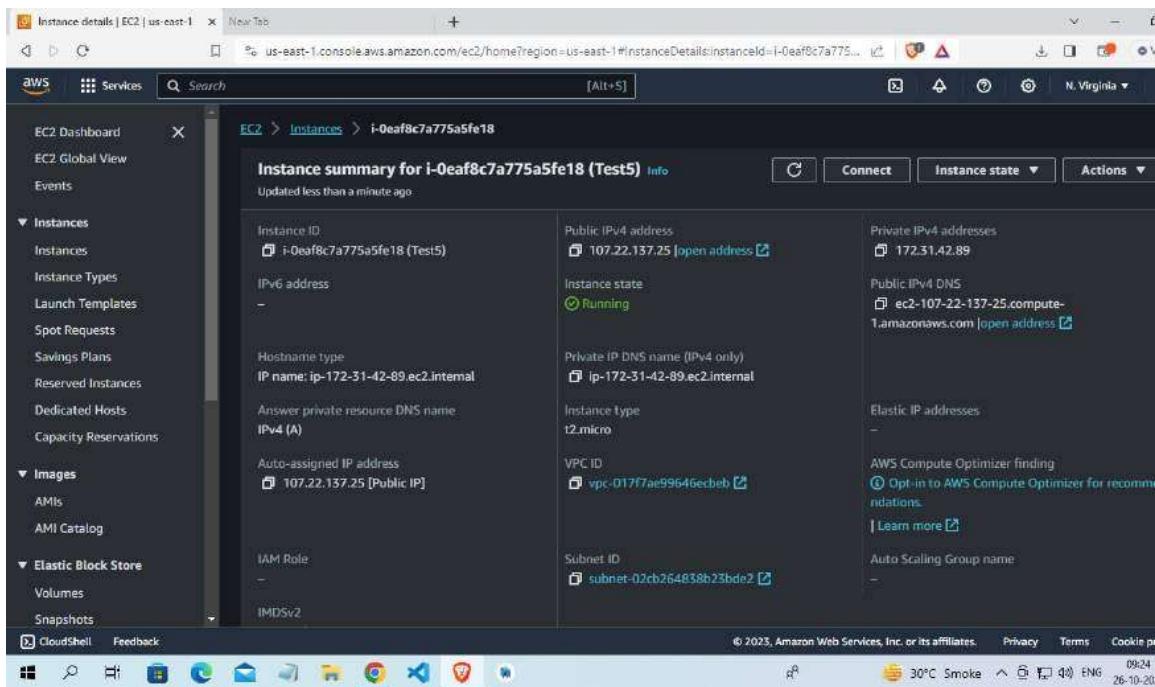
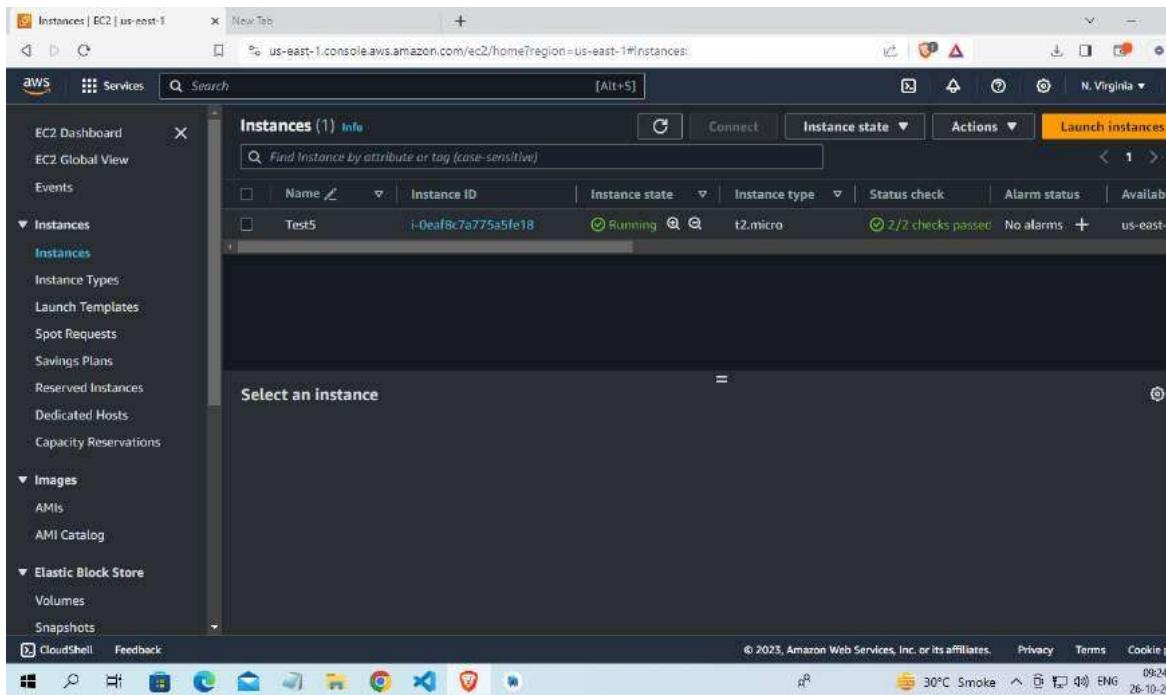
## Step 5:

### Launch the Instance



## Step 6:

Launch an Instance: Click the "Instances" link in the left sidebar and then click the "Launch Instance" button.

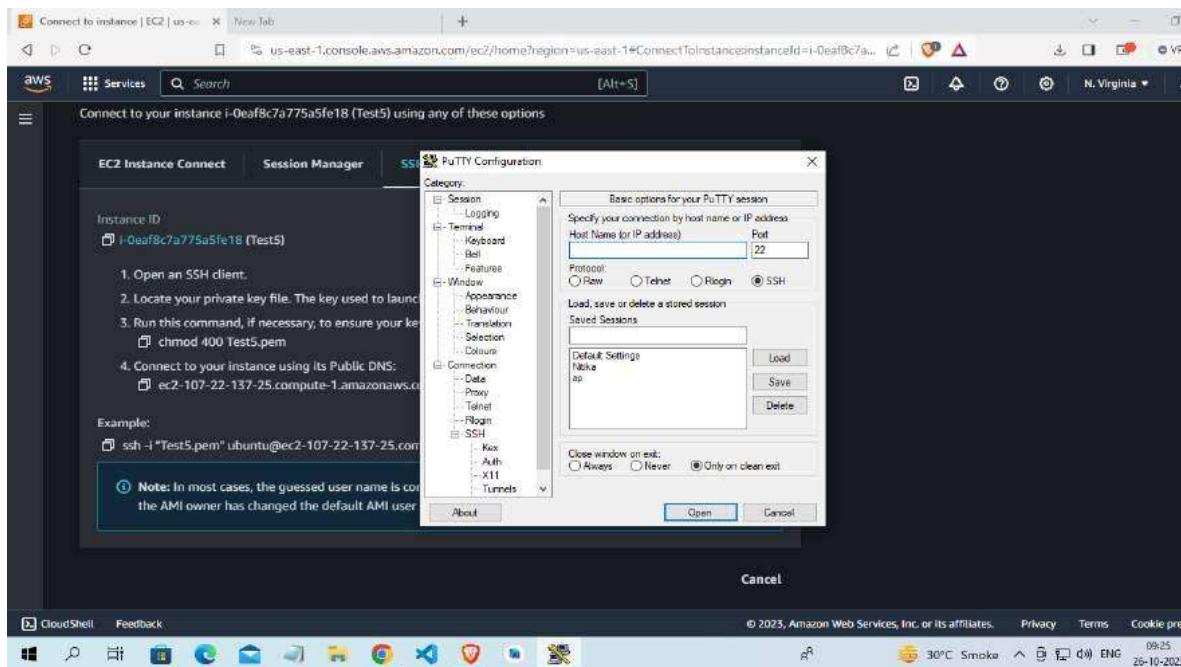


## Step 7:

Download PuTTYgen .

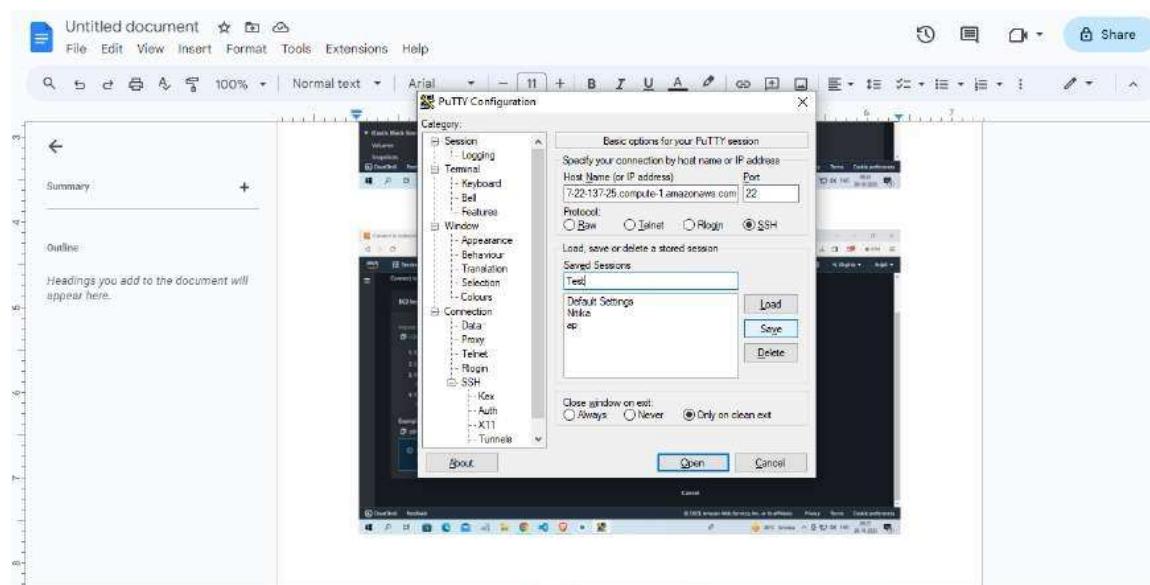
## Step 8:

If you are using Windows, use PuTTY and PuTTYgen to connect. For Linux and Mac users, you can use the terminal.



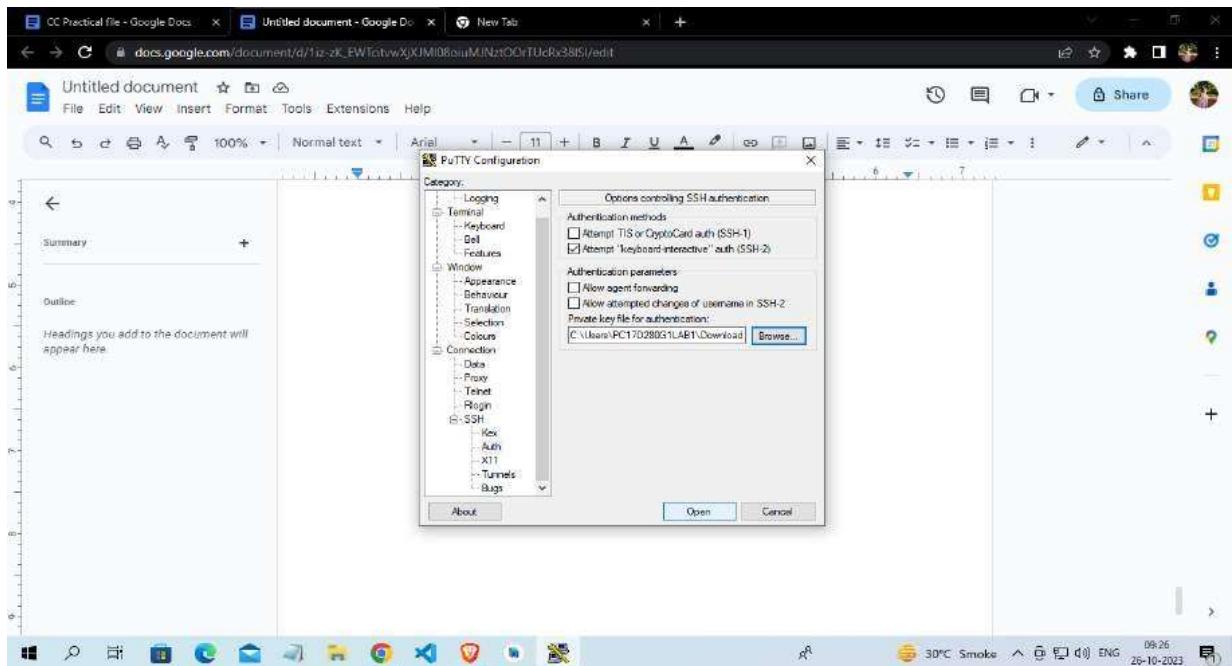
## Step 9:

Go to Connection -> ssh -> Authentication -> Credential



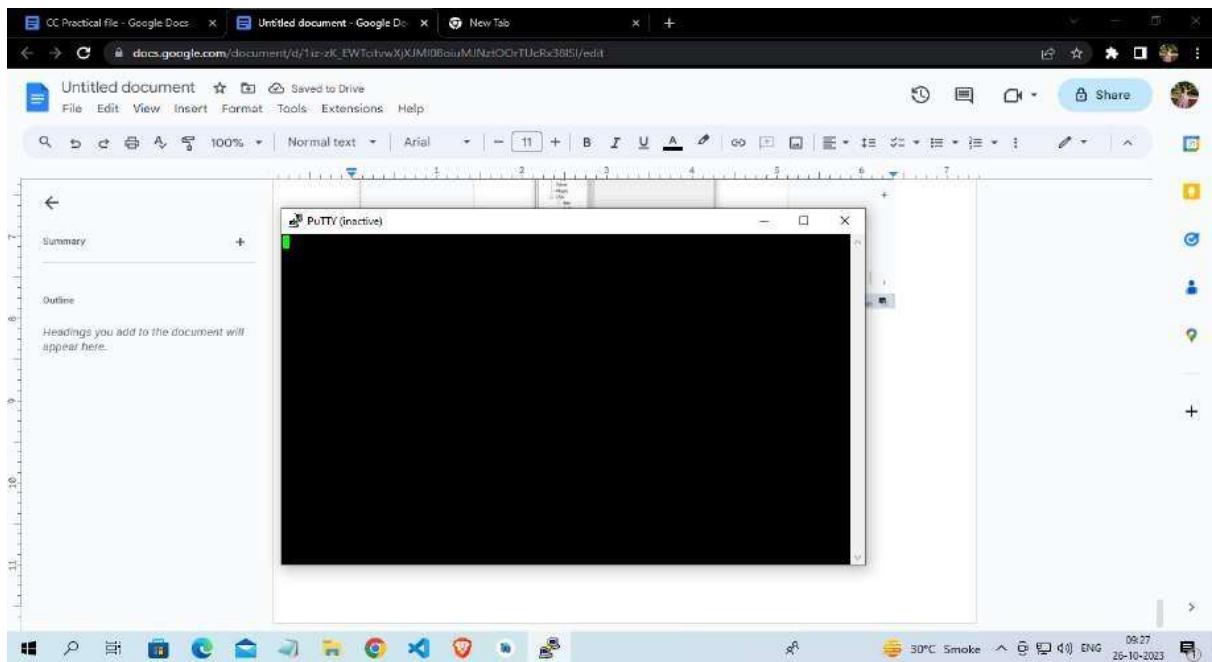
## Step 10:

If using PuTTY on Windows, use PuTTYgen to load your private key (.pem), convert it to a PPK format, and then use PuTTY to connect.



## Step 11:

Linux Instance has been Launched

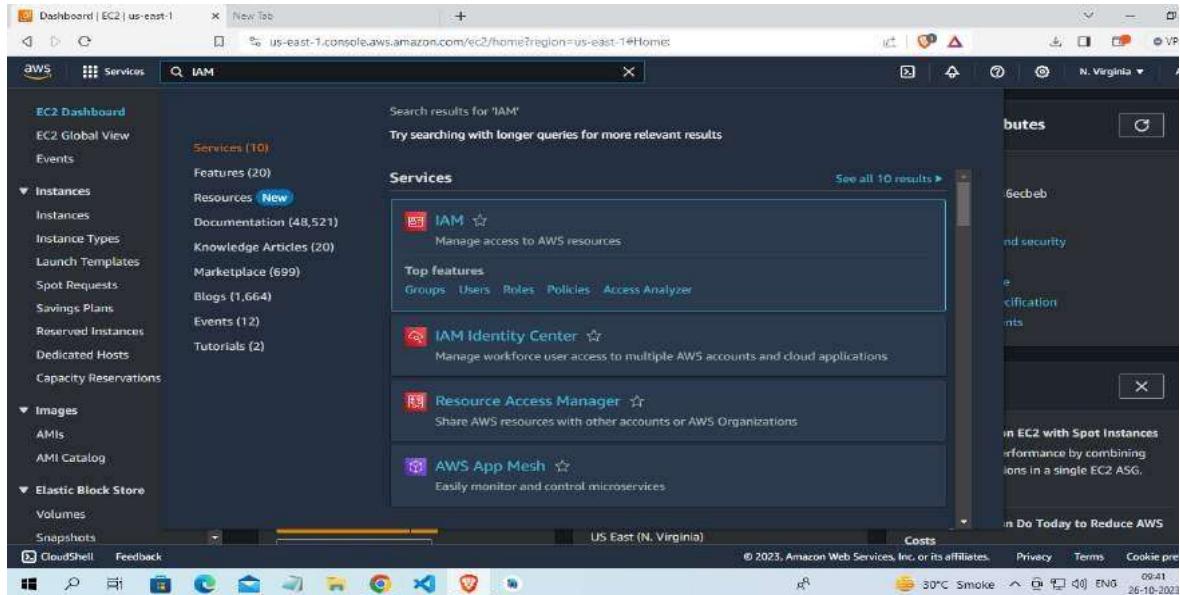


## Practical 12:

### Create IAM user and grant in limited permission to IAM user by AWS route user

#### Step 1 :

Once logged in, navigate to the IAM (Identity and Access Management) Console. You can do this by searching for "IAM" in the AWS Management Console's search bar or by selecting "Security, Identity, & Compliance" and then "IAM" under the "Services" menu.



#### Step 2:

In the IAM dashboard, click on "Users" in the left navigation pane to view the list of existing IAM users.

The screenshot shows the AWS Identity and Access Management (IAM) dashboard. On the left, a navigation menu includes options like Dashboard, User groups, Users, Roles, Policies, Identity providers, Account settings, and more. The main area displays 'Security recommendations' with a red notification badge (1). It lists two items: 'Add MFA for root user' (yellow warning icon) and 'Root user has no active access keys' (green success icon). Below this is a summary of 'IAM resources' with counts: User groups (0), Users (0), Roles (2), Policies (0), and Identity providers (0). To the right, the 'AWS Account' section shows the Account ID (804008392614), Account Alias (Create), and Sign-in URL (https://804008392614.signin.aws.amazon.com/console). A 'Quick Links' section includes a link to 'My security credentials'. The bottom of the screen shows the Windows taskbar with various pinned icons.

### Step 3 :

To create a new IAM user, click the "Add user" button at the top of the dashboard.

The screenshot shows the 'Users' page under the IAM service. The left sidebar has the same navigation as the dashboard. The main area shows a table titled 'Users (0)' with one entry: 'No resources to display'. At the top right of the table, there is a 'Create user' button. The bottom of the screen shows the Windows taskbar.

### Step 4 :

User name: Enter a unique name for the IAM user.

The image consists of three vertically stacked screenshots from the AWS IAM console.

**Screenshot 1: Step 3 - Review and create**

This screenshot shows the "Create user" wizard, Step 3: Review and create. It displays the user name "Anjali Kumari" and a checkbox for "Provide user access to the AWS Management Console - optional". A callout box highlights the "Are you providing console access to a person?" section, which contains two options: "Specify a user in Identity Center - Recommended" (selected) and "I want to create an IAM user". Below this, a note states: "If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)".

**Screenshot 2: IAM Identity Center (successor to AWS Single Sign-On)**

This screenshot shows the IAM Identity Center landing page. It features a large heading "IAM Identity Center (successor to AWS Single Sign-On)" and a sub-headline "Manage workforce access to multiple AWS accounts and cloud applications.". A callout box on the right titled "Enable IAM Identity Center" contains the text: "IAM Identity Center makes it easy to connect an existing directory or use the built-in Identity Center directory to manage user access to AWS accounts and cloud applications." with a "Enable" button.

**Screenshot 3: Getting started**

This screenshot shows the "Getting started" section of the IAM Identity Center. It includes links for "Get started with IAM Identity Center" and "IAM Identity Center prerequisites".

## Step 5 :

After the user is successfully created, you'll see a confirmation page. This page provides important information, such as the user's access key and secret access key (if you selected "Programmatic access"). Make sure to download and securely store the access keys because they will only be displayed once.

**IAM Identity Center requires AWS Organizations**

We detected that your AWS account does not currently use this service. After you create an organization, you cannot join this account to another organization until you delete its current organization.

AWS Organizations provides the following benefits:

1. Enables single payer and centralized cost tracking
2. Lets you create and invite other AWS accounts
3. Allows you to apply policy-based controls
4. Helps you simplify organization-wide management of AWS services

Would you like us to create an AWS organization for you now? We will also enable IAM Identity Center as part of this process.

Getting started

Create AWS organization

**IAM Identity Center**

**Dashboard**

The features that comprised AWS Single Sign-On (AWS SSO) are available through the IAM Identity Center console. They offer a better way to connect to a workforce directory, and to manage users' access across AWS accounts and integrated applications. Learn more

**Dashboard**

IAM Identity Center enables you to manage workforce user access to multiple AWS accounts and applications. Learn more

**Recommended setup steps**

Step 1  
Choose your identity source  
The identity source is where you administer users and groups, and is the service that authenticates your users.

Step 2  
Manage access to multiple AWS accounts  
Give users and groups access to specific AWS accounts in your organization.

**Settings summary**

Go to settings

Identity source: Identity Center directory  
Region: US East (N. Virginia) | us-east-1  
AWS access portal URL: https://d-90678913c2.awsapps.com/start

## Step 6:

Give limited permission rights by navigating to permissions

## **Practical 13:**

### **Create a bucket by using S3 AWS service**

**Objective :** The primary objective of Amazon S3 (Simple Storage Service) buckets in AWS is to provide a highly scalable, durable, and secure storage solution for a wide range of use cases. S3 buckets serve as containers for storing and managing data, and their key objectives include:

- Scalable Storage: S3 buckets can store an almost unlimited amount of data, making it suitable for organizations of all sizes. You can start with a small amount of storage and scale as needed without any disruption.
- Durability: Data stored in S3 buckets is designed to be highly durable. AWS replicates data across multiple Availability Zones, providing 99.99999999% (11 nines) durability. This means data is protected against hardware failures, and even if an entire Availability Zone goes down, your data is still safe.
- Data Availability: S3 provides high data availability. Your data is accessible over the internet, and you can access it from anywhere with an internet connection. This makes it a suitable solution for content delivery and web hosting.
- Data Security: S3 buckets offer various security features, including access control through bucket policies, IAM roles, and Access Control Lists (ACLs). You can also enable server-side encryption to protect data at rest.

## Step 1:

Once logged in, navigate to the S3 dashboard. You can do this by searching for "S3" in the AWS Management Console's search bar or by selecting "Storage" and then "S3" under the "Services" menu.

The screenshot shows the AWS Management Console search interface. The search bar at the top contains the query 's3'. Below the search bar, the left sidebar is titled 'Amazon S3' and lists various services and features. The 'Features' section is expanded, showing categories like 'Imports from S3', 'Batch Operations', 'Buckets', and 'Access points'. A large orange 'Create bucket' button is prominently displayed on the right side of the search results page. The bottom of the screen shows the Windows taskbar with several pinned icons.

## Step 2:

Click the "Create bucket" button.

The screenshot shows the 'Buckets' page in the AWS S3 Management Console. The left sidebar is identical to the previous search results page. The main content area displays an 'Account snapshot' summary and a table for managing buckets. The table has columns for 'Name', 'AWS Region', 'Access', and 'Creation date'. A large orange 'Create bucket' button is located at the bottom right of the table area. The bottom of the screen shows the Windows taskbar.

### Step 3:

In the "Bucket name" field, enter a unique and globally-unique name for your bucket. Bucket names must be unique across all of AWS.

The screenshot shows the 'Create bucket' page in the AWS S3 console. The 'Bucket name' field is filled with 'test0203'. The 'AWS Region' dropdown is set to 'US East (N. Virginia) us-east-1'. In the 'Object Ownership' section, the radio button for 'ACLs disabled (recommended)' is selected, while 'ACLs enabled' is unselected. Other configuration options like 'Copy settings from existing bucket - optional' are also visible.

### Step 4:

Review your configuration and click the "Create bucket" button to create your S3 bucket.

The screenshot shows the 'Buckets' page in the AWS S3 console. A green banner at the top indicates that the bucket 'test0203' was successfully created. The 'Account snapshot' section displays the single bucket. The table below lists the bucket details: Name (test0203), AWS Region (US East (N. Virginia) us-east-1), Access (not explicitly shown), and Creation date (October 28, 2023, 21:45:13 (UTC+05:30)).

Name	AWS Region	Access	Creation date
test0203	US East (N. Virginia) us-east-1		October 28, 2023, 21:45:13 (UTC+05:30)

## Practical 14:

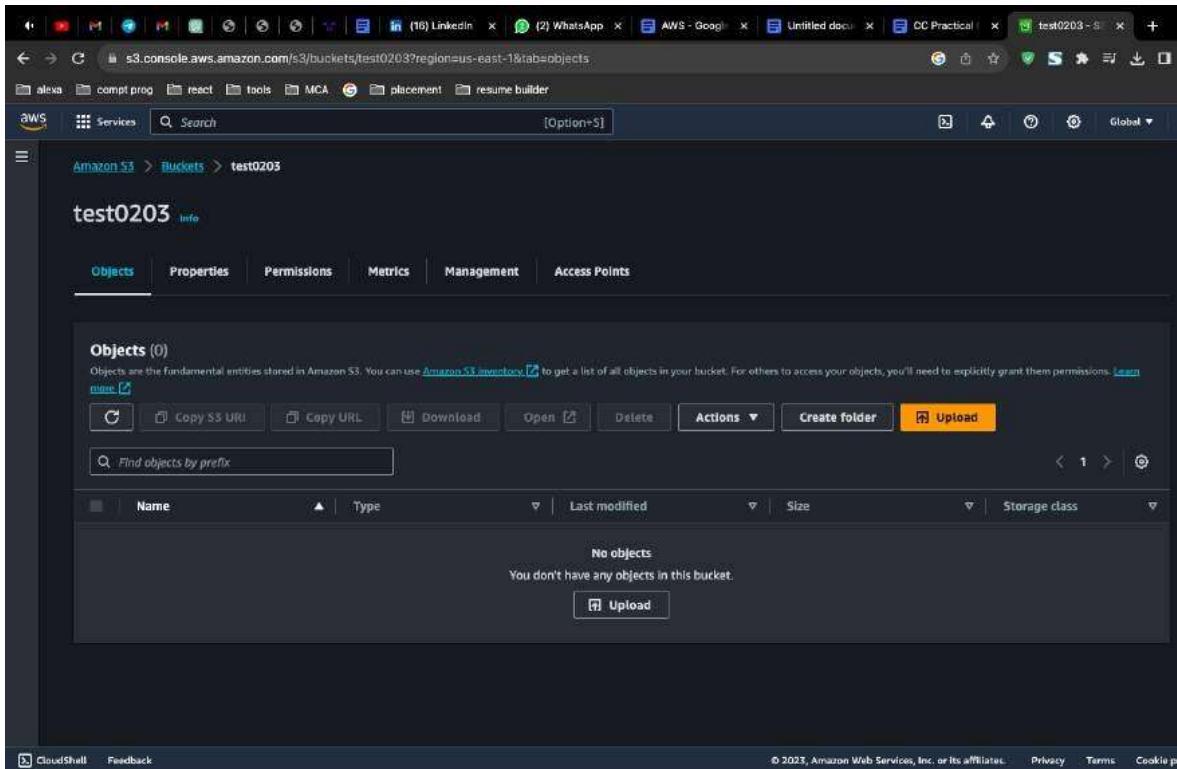
### Upload an object on bucket created by using S3 AWS service

**Objective :** The objective of the practical task "Upload an object on bucket created by using S3 AWS service"

is to demonstrate how to upload files to an Amazon S3 bucket. This can be useful for various purposes, such as hosting static websites, sharing files, or distributing public content.

#### Step 1:

Select the object (file) you want to make publicly accessible by checking the checkbox next to its name.



**Step 2:** Click the "Actions" button, and from the dropdown menu, select "Make public."

Note: If you want all objects in the bucket to be public, you can modify the bucket's access control settings to allow public access.

The screenshot shows the AWS S3 'Upload' interface. At the top, there's a message: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more.' Below this is a large input field with the placeholder 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' A table titled 'Files and folders (1 Total, 10.1 MB)' lists one file: 'IMG\_1008.JPG' (image/jpeg, 10.1 MB). There are 'Remove', 'Add files', and 'Add folder' buttons above the table. The 'Destination' section shows 'Destination' set to 's3://test0203'. Under 'Destination details', it says 'Bucket settings that impact new objects stored in the specified destination.' The 'Permissions' section allows granting public access and access to other AWS accounts. The bottom right has 'Cancel' and 'Upload' buttons.

This screenshot is identical to the one above, showing the AWS S3 'Upload' interface. The main difference is that the 'Upload' button at the bottom right is now highlighted in orange, indicating it is the active button to proceed with the upload.

## Step 3: File Uploaded in Bucket

The screenshot shows the AWS S3 console interface. At the top, a progress bar indicates 'Uploading' with '1%' completed. Below it, a message says 'Total remaining: 1 File: 10.0 MB (99.61%)' and 'Estimated time remaining: 4 minutes'. Transfer rate is listed as '44.8 KB/s'. The main section is titled 'Upload: status' with a note: 'The information below will no longer be available after you navigate away from this page.' A 'Summary' table shows the destination 's3://test0203' with 'Succeeded' (0 files, 144.0 KB (1.39%)) and 'Failed' (0 files, 0 B (0%)). Below this is a 'Files and folders' table with one item: 'Files and folders (1 Total, 10.1 MB)'. The table has columns: Name, Folder, Type, Size, Status, and Error. The single entry is 'IMG\_1008.JPG' with a size of '10.1 MB' and a status of 'Succeeded'. The bottom of the screen includes standard browser navigation and AWS links.

This screenshot shows the same AWS S3 console interface after the upload has completed successfully. The top message now reads 'Upload succeeded' with a link to 'View details below.'. The 'Upload: status' section still displays the summary table with the same data, and the 'Files and folders' table shows the uploaded file 'IMG\_1008.JPG' with a green 'Succeeded' status. The overall interface remains consistent with the first screenshot, including the header and footer elements.

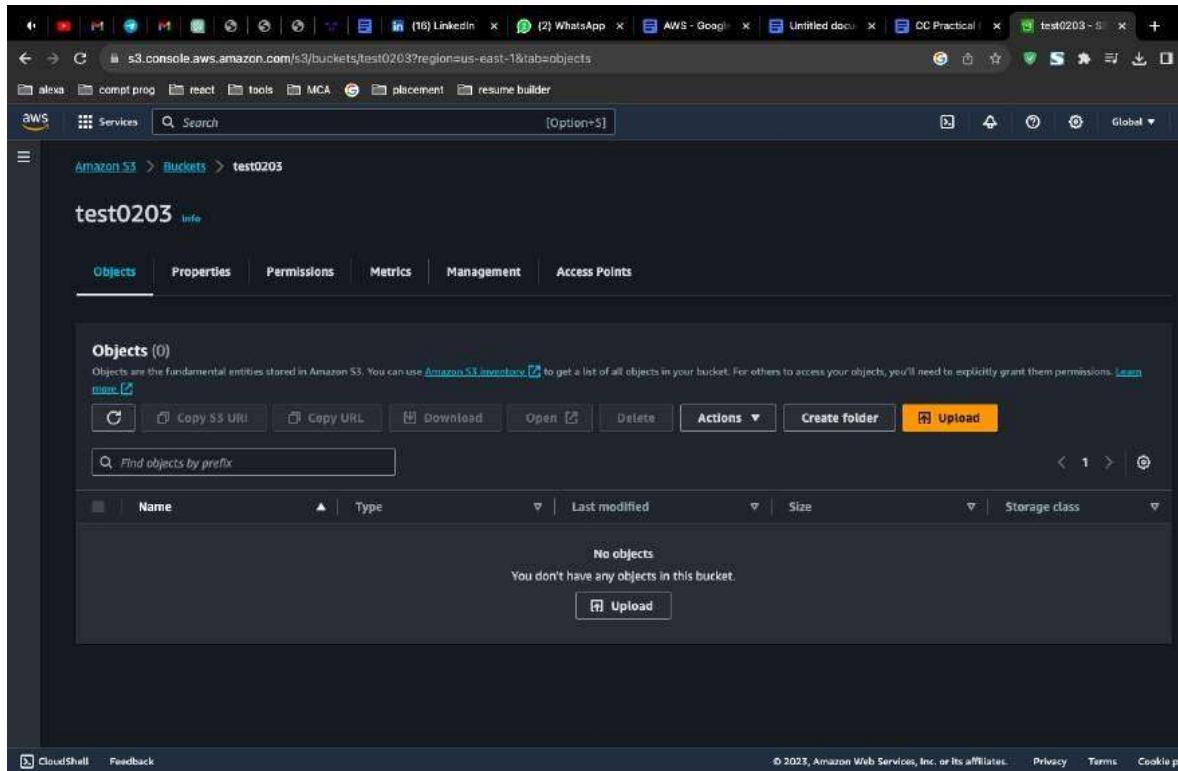
## Practical 15:

### Upload objects and enable it by using URL to get public access in the created bucket.

**Objective :** The objective of the practical task "Upload objects and enable public access via URL in a created Amazon S3 bucket" is to demonstrate how to upload files to an Amazon S3 bucket and configure the bucket settings to make those objects publicly accessible via a URL. This can be useful for various purposes, such as hosting static websites, sharing files, or distributing public content.

#### Step 1:

Select the object (file) you want to make publicly accessible by checking the checkbox next to its name.



#### Step 2:

Click the "Actions" button, and from the dropdown menu, select "Make public."

**Note:** If you want all objects in the bucket to be public, you can modify the bucket's access control settings to allow public access.

The screenshot shows the AWS S3 console's 'Upload' interface. A single file, 'IMG\_1008.JPG', is selected for upload. The destination is set to 'test0203'. The 'Destination details' section is expanded, showing bucket settings. The 'Permissions' section is also visible. The 'Upload' button is highlighted at the bottom right.

Files and folders (1 Total, 10.1 MB)

Name	Type	Size
IMG_1008.JPG	image/jpeg	10.1 MB

Destination

Destination  
s3://test0203

▶ Destination details

Bucket settings that impact new objects stored in the specified destination.

▶ Permissions

Grant public access and access to other AWS accounts.

Cancel      Upload

### Step 3:

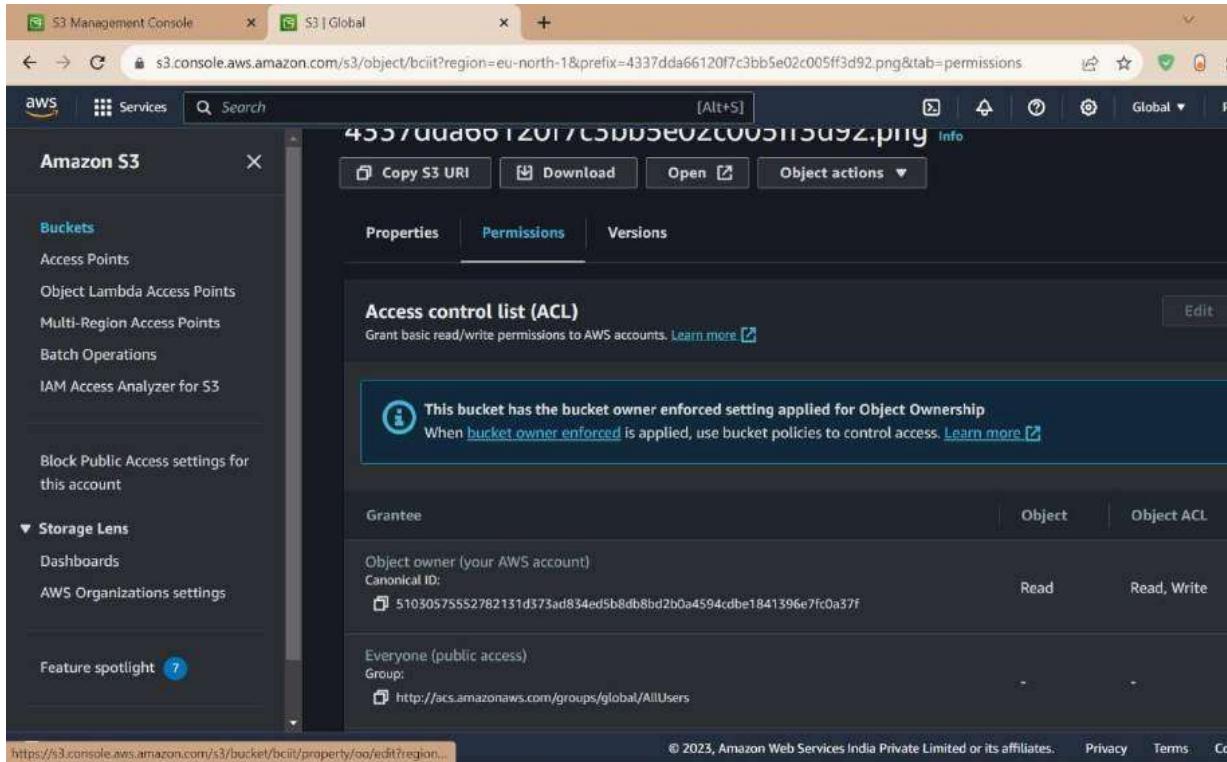
File Uploaded in Bucket

The screenshot shows the AWS S3 console interface. At the top, there is a progress bar indicating "Uploading" with a status message: "Total remaining: 1 file; 10.0 MB(98.61%)". Below this, the "Upload: status" section displays a summary table. The table has three columns: Destination, Succeeded, and Failed. The Destination column shows "s3://test0203". The Succeeded column shows "0 files, 144.0 KB (1.39%)". The Failed column shows "0 files, 0 B (0%)". Below the summary table, there are tabs for "Files and folders" and "Configuration", with "Files and folders" being selected. A table titled "Files and folders (1 Total, 10.1 MB)" lists one item: "IMG\_1008.JPG" which is "Image/jpeg", "10.1 MB", and "Succeeded".

The screenshot shows the AWS S3 console interface after the upload has completed successfully. A green banner at the top left says "Upload succeeded" with a link to "View details below.". Below this, the "Upload: status" section displays a summary table. The table has three columns: Destination, Succeeded, and Failed. The Destination column shows "s3://test0203". The Succeeded column shows "1 file, 10.1 MB (100.00%)". The Failed column shows "0 files, 0 B (0%)". Below the summary table, there are tabs for "Files and folders" and "Configuration", with "Files and folders" being selected. A table titled "Files and folders (1 Total, 10.1 MB)" lists one item: "IMG\_1008.JPG" which is "Image/jpeg", "10.1 MB", and "Succeeded".

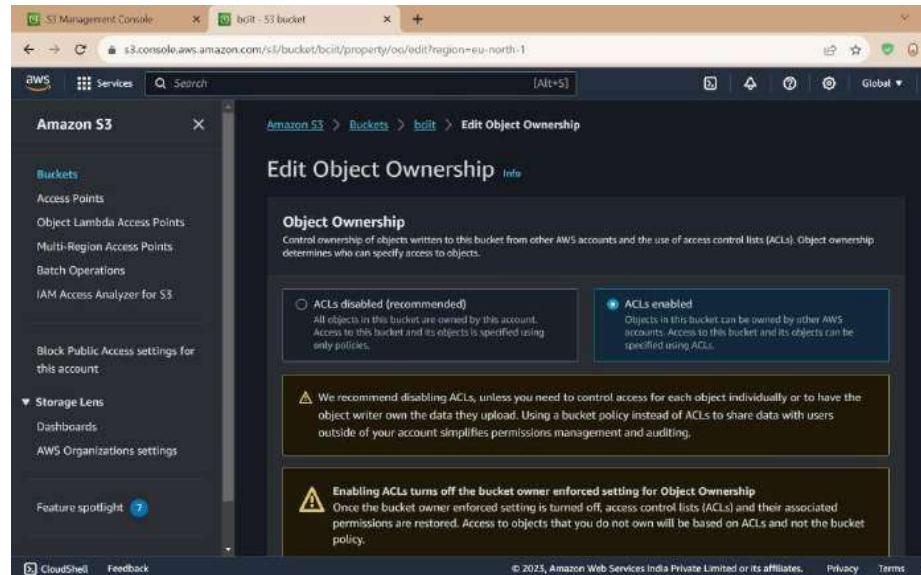
## **Step 5 :**

Go to permissions and click on bucket owner enforced



## **Step 6:**

click on ACL2 enabled



## Step 7 :

give access to read and write

Grantee	Objects	Object ACL
Object owner (your AWS account)	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read <input type="checkbox"/> Write
Canonical ID: 5103057552782131d37 3ad834ed5b8db8bd2b0a4594c dbe1841396e7fc0a37f		
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	<input checked="" type="checkbox"/> <span style="color: yellow;">⚠</span> Read	<input checked="" type="checkbox"/> <span style="color: yellow;">⚠</span> Read <input type="checkbox"/> Write
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	<input checked="" type="checkbox"/> <span style="color: yellow;">⚠</span> Read	<input checked="" type="checkbox"/> <span style="color: yellow;">⚠</span> Read <input type="checkbox"/> Write

## Step 8:

Navigate to S3 terminal and paste the object URL

```
[root@ip-172-31-36-199 ec2-user]# wget https://bciiit.s3.eu-north-1.amazonaws.com/4337dda66120f7c3bb5e02c005ff3d92.png
--2023-10-16 05:05:42-- https://bciiit.s3.eu-north-1.amazonaws.com/4337dda66120f7c3bb5e02c005ff3d92.png
Resolving bciiit.s3.eu-north-1.amazonaws.com (bciiit.s3.eu-north-1.amazonaws.com)... 52.95.171.72, 52.95.171.40
Connecting to bciiit.s3.eu-north-1.amazonaws.com (bciiit.s3.eu-north-1.amazonaws.com)|52.95.171.72|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13025 (13K) [image/png]
Saving to: '4337dda66120f7c3bb5e02c005ff3d92.png'

4337dda66120f7c3bb5e02c005ff3d92.png 100%[=====] 12.72K --.-KB/s   in
2023-10-16 05:05:43 (96.4 MB/s) - '4337dda66120f7c3bb5e02c005ff3d92.png' saved [13025/13025]

[root@ip-172-31-36-199 ec2-user]# ls
4337dda66120f7c3bb5e02c005ff3d92.png
[root@ip-172-31-36-199 ec2-user]# 
```

## Practical 16:

### Delete the created object and its bucket.

**Objective :** The objective of deleting objects in an Amazon S3 (Simple Storage Service) bucket in AWS can vary depending on the specific use case and needs of the user. Here are some common objectives for deleting objects from an S3 bucket:

- Data Cleanup: Removing outdated or unnecessary objects to free up storage space and reduce storage costs. Over time, old versions of files or expired data can accumulate, and deleting them helps maintain an efficient storage environment.
- Security: Deleting sensitive or confidential data that is no longer needed to reduce the risk of unauthorized access or data breaches. This is especially important for compliance with data privacy regulations.
- Version Control: Managing versioned objects by removing older versions of files that are no longer relevant. This ensures that only the most up-to-date versions are retained.
- Archiving: Deleting objects that have been archived to more cost-effective storage classes, such as Amazon Glacier, when they are no longer needed in the original S3 bucket.
- Temporary Files: Removing temporary files or objects that were only needed for a specific task or process. This helps in keeping the bucket organized and reducing clutter.

#### Step 1:

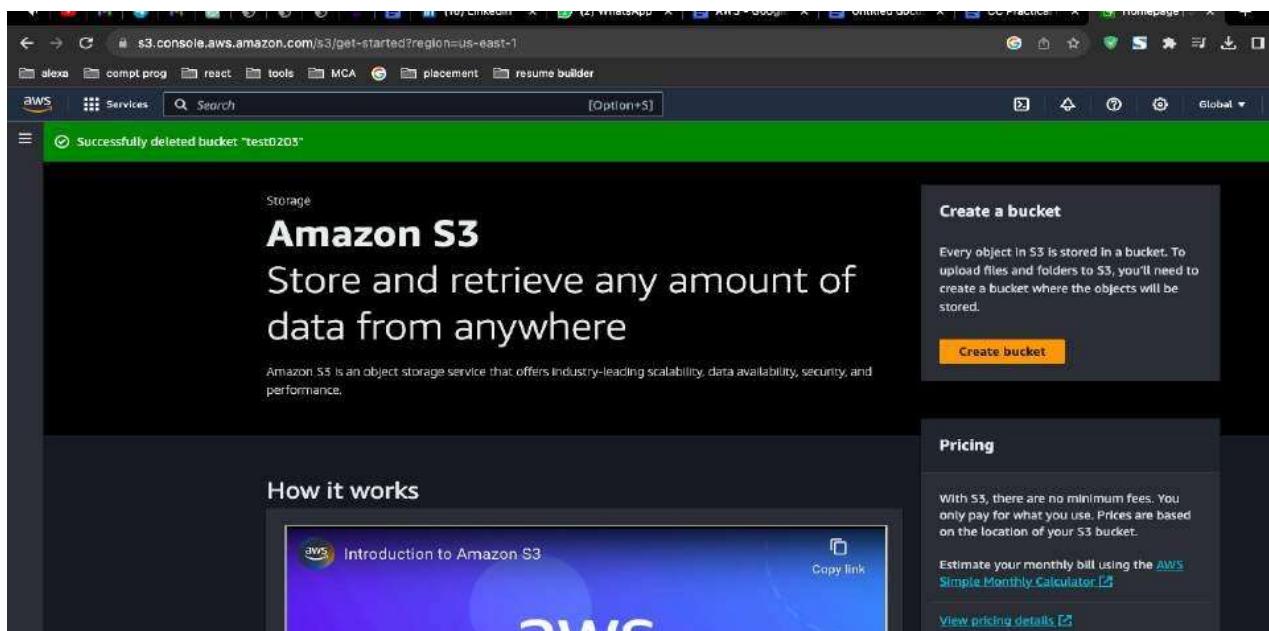
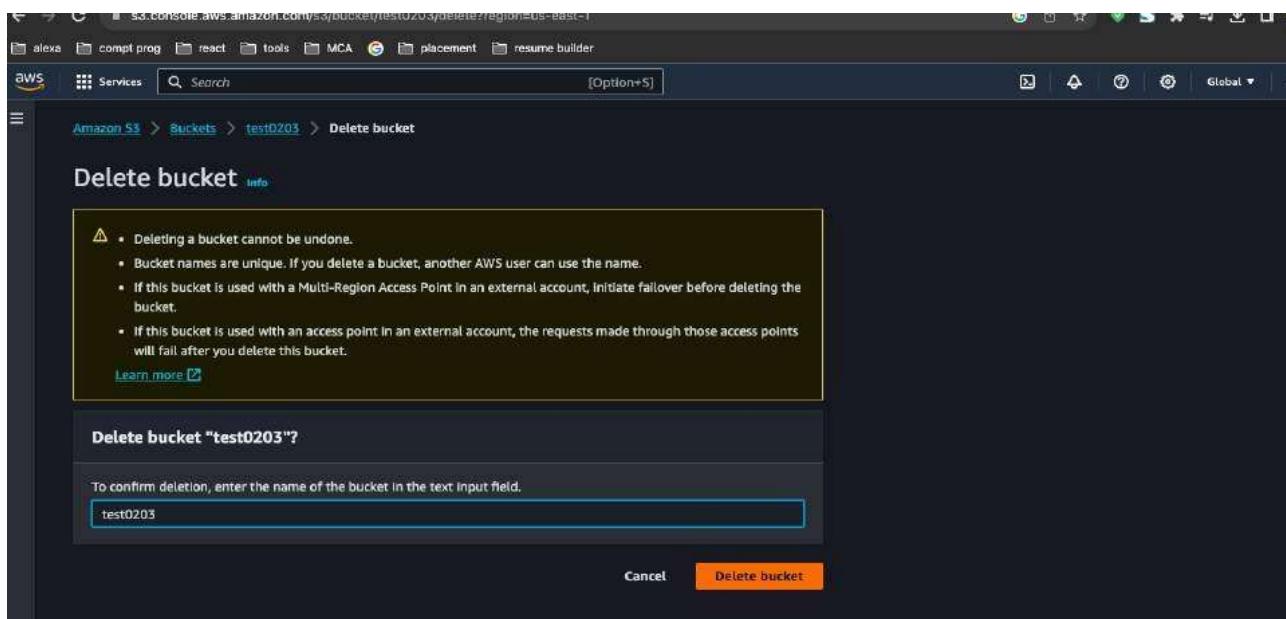
In the S3 dashboard, click on the name of the bucket from which you want to delete objects.

The screenshot shows the AWS S3 Buckets page. At the top, there's a header with the AWS logo and a search bar. Below the header, the URL is s3.console.aws.amazon.com/s3/buckets?region=us-east-1&region=us-east-1. The main content area has a title 'Amazon S3 > Buckets'. Underneath, there's an 'Account snapshot' section with a link to 'View Storage Lens dashboard'. A table lists 'Buckets (1) Info', with a note that buckets are containers for data stored in S3. The table has columns for Name, AWS Region, Access, and Creation date. One row is shown for 'test0203' in 'US East (N. Virginia) us-east-1', with 'Bucket and objects not public' under Access and 'October 28, 2023, 21:45:13 (UTC+05:30)' under Creation date. Action buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket' are visible above the table. At the bottom, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2023, Amazon Web Services, Inc. or its affiliates.' and 'Privacy Terms Cookie policy'.

## Step 2:

In the bucket, navigate to the objects you want to delete. You can do this by clicking on the folders and subfolders, if applicable, to locate the objects.

- Select the objects you wish to delete by checking the checkboxes next to their names.
- Once the objects are selected, you can choose one of the following methods to delete them:
  - Click the "Actions" button, then select "Delete" to delete the selected objects.
  - Alternatively, you can simply press the "Delete" key on your keyboard after selecting the objects.
  - Confirm the deletion by clicking "Delete" in the confirmation dialog.



## Practical 18: Create VPC and implement EC2 services on it.

The screenshot shows the AWS VPC Dashboard. On the left sidebar, under 'Virtual Private Cloud', 'Your VPCs' is selected. The main area displays 'Resources' with tabs for 'Start VPC Wizard' and 'Launch EC2 Instances'. Below this, a message says 'You are using the following Amazon VPC resources in the US East (N. Virginia) region.' A detailed list follows:

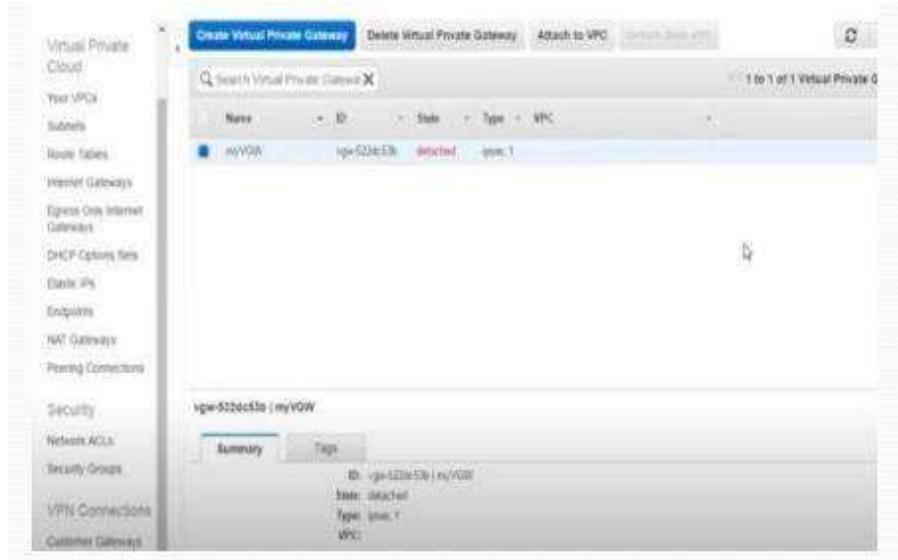
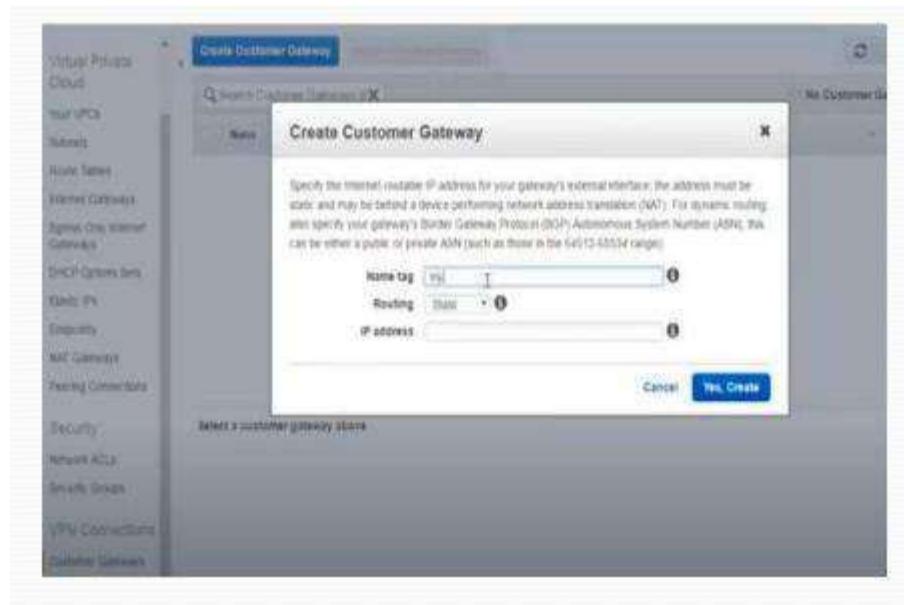
Count	Resource Type
1	VPCs
0	Egress Only Internet Gateways
1	Route Tables
1	Internet Gateways
0	Egress Only Internet Gateways
0	DHCP Options Sets
0	Classic IPs
0	Encryption
0	NAT Gateways
0	Peering Connections
0	Security
0	Network ACLs
0	Security Groups

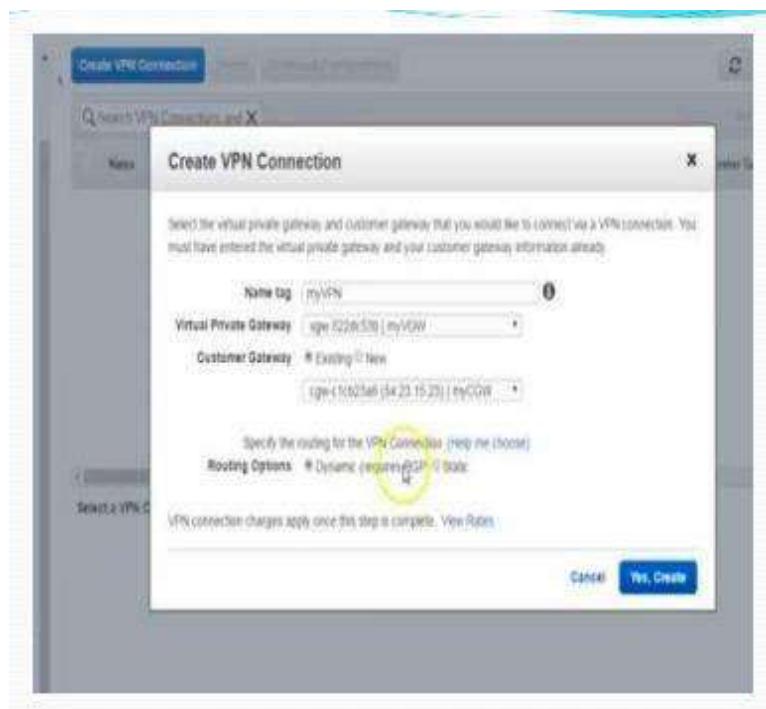
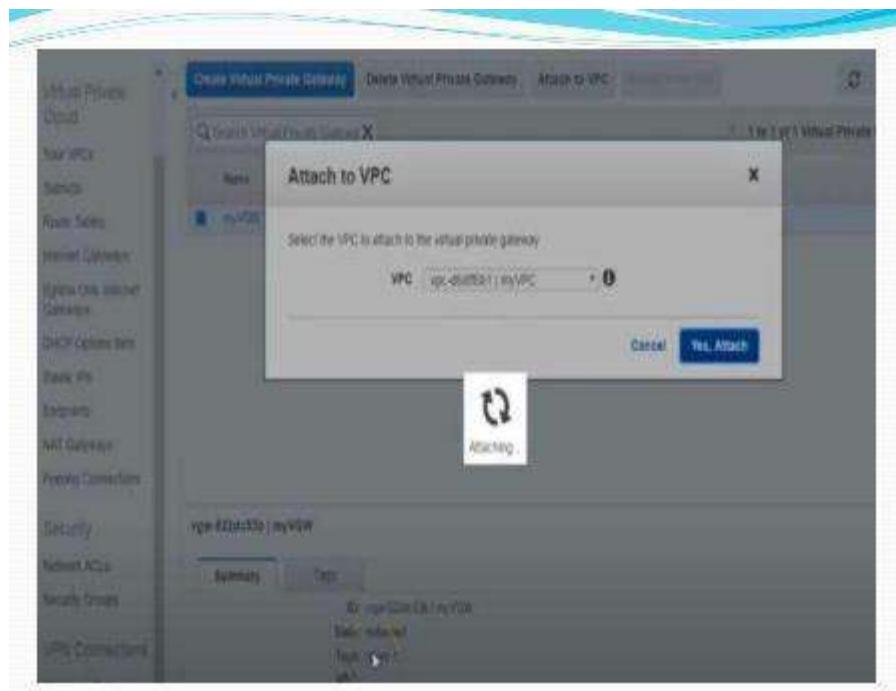
On the right, the 'Service Health' section shows two items: 'Amazon VPC - US East (N. Virginia)' and 'Amazon EC2 - US East (N. Virginia)', both listed as 'Service is operating normally'. Below this is an 'Additional Information' section with links to 'VPC Documentation', 'All VPC Resources', 'Feedback', and 'Report an issue'.

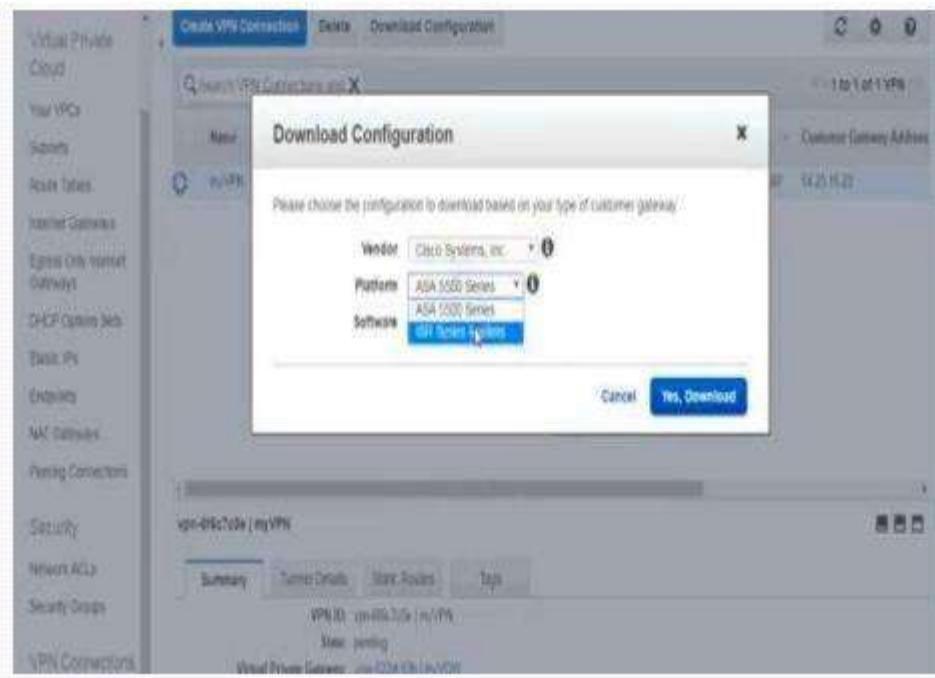
The screenshot shows the AWS VPC Dashboard with 'Create VPC' selected in the top navigation bar. A search bar at the top contains 'myVPC'. The main table lists one VPC entry:

Name	VPC ID	Status	IPv4 CIDR	IPv6 CIDR	DHCP options set	Route table	Network ACL
myVPC	vpc-d50f01	Available	10.0.0.0/16		dhcpcid4	rtb-467a6f1	na-10e647e

A yellow circle highlights the 'myVPC' row in the table. At the bottom of the page, a progress bar indicates the task is 100% complete.







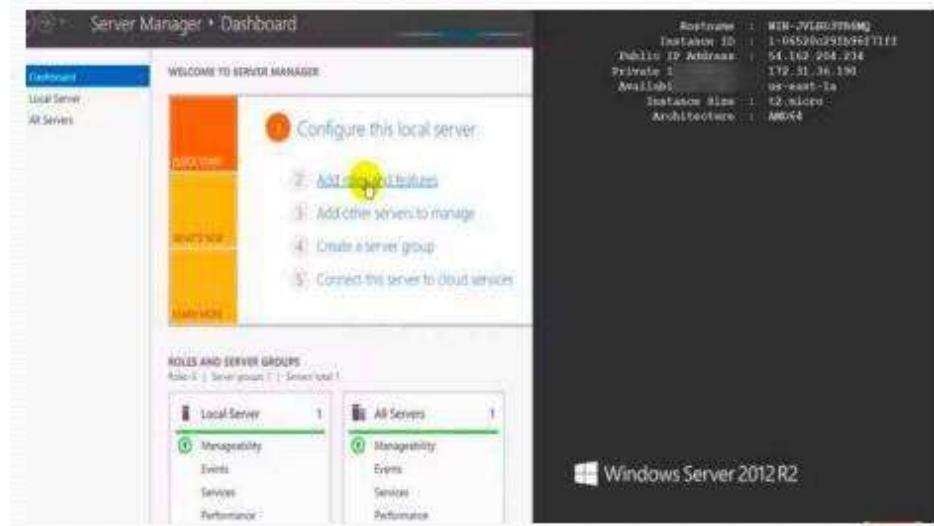
```

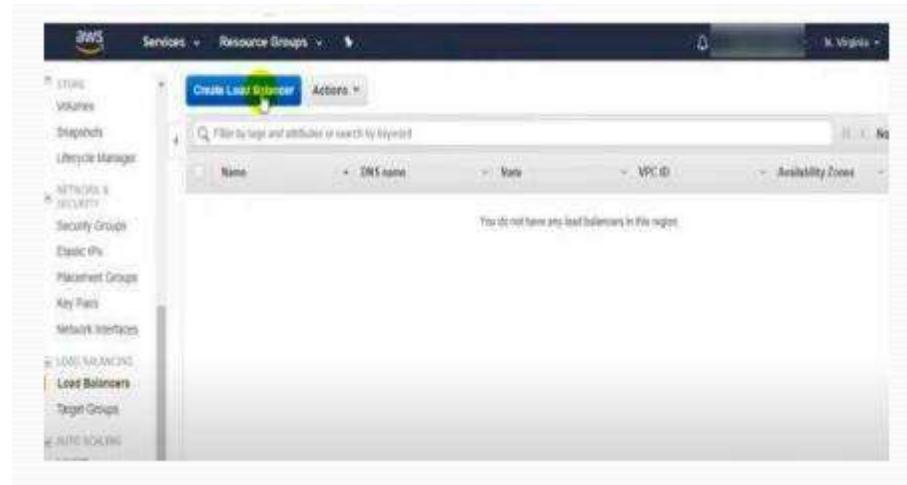
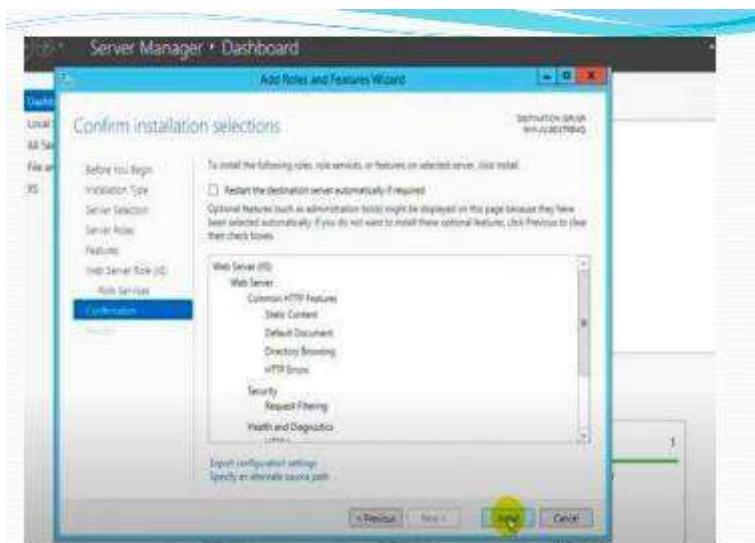
1 ! Amazon Web Services
2 ! Virtual Private Cloud
3 !
4 ! AWS utilizes unique identifiers to manipulate the configuration of
5 ! a VPN Connection. Each VPN Connection is assigned an identifier and is
6 ! associated with two other identifiers, namely the
7 ! Customer Gateway Identifier and Virtual Private Gateway Identifier.
8 !
9 ! Your VPN Connection ID : vpn-6f6c7c0e
10 ! Your Virtual Private Gateway ID : vgw-522dc53b
11 ! Your Customer Gateway ID : cgw-c1cb13a1
12 !
13 !
14 ! This configuration consists of two tunnels. Both tunnels must be
15 ! configured on your Customer Gateway. Only a single tunnel will be up at a
16 ! time to the VGW.
17 !
18 ! You may need to populate these values throughout the config based on your setup:
19 ! <outside_interface> - External interface of the ASA
20 ! <outside_access_in> - Inbound ACL on the external interface
21 ! <amazon_vpn_map> - Outside crypto map
22 ! <vpn_subnet> and <vpn_subnet_mask> - VPC address range
23 ! <local_subnet> and <local_subnet_mask> - Local subnet address range
24 ! <sia_monitor_address> - Target address that is part of acl-amzn to run SIA monitoring
25 !
26 !
27 ! IPsec Tunnels
28 !
29 ! #1: Internet Key Exchange (IKE) Configuration

```

## Practical 19:

### Implement & Configure load balancing with all necessary steps.





**Step 1: Configure Load Balancer**

### Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet listener that receives HTTP traffic on port 80.

Name	<input type="text" value="ApplicationLB"/>
Scheme	<input checked="" type="radio"/> Internet-facing <input type="radio"/> Internal
IP address type	<input type="text" value="IPv4"/>

**Listeners**

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
------------------------	--------------------

**Step 3: Configure Security Groups**

A security group is a set of firewall rules that controls the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

Assign a security group:  Create a new security group  
 Select an existing security group

Security Group ID	Name	Description
sg-37e490a	default	default VPC security group.
sg-0153b6848c70cd	ECS-1	Launch wizard-1 created 2019-11-07T17:20:21.327+00:00

**Step 4: Configure Routing**

Name:

Target type:  Instance  
 IP  
 Lambda function

Protocol:

Port:

**Health checks**

Protocol:   
Path:

Advanced health check settings

**Buttons:** Cancel | Previous | Next

The screenshot shows the 'Step 5: Register Targets' page of the AWS CloudFront configuration wizard. The top navigation bar includes links for '1. Configure Load Balancer', '2. Configure Security Settings', '3. Configure Security Groups', '4. Configure Routing', '5. Register Targets' (which is underlined in blue), and '6. Review'. The main content area is titled 'Step 5: Register Targets' with the sub-instruction 'Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts making requests to the targets as soon as the registration process completes all passes the initial health checks.' Below this, a section titled 'Registered targets' contains a table with two entries:

Instance	Name	Port	Status	Security groups	Zone
i-061401a8359fc	ELB140140	80	Green (running)	ELB001	us-east-1a
i-06206298a071f	ELB140140	80	Green (running)	ELB001	us-east-1a

Below the table, there's a section titled 'Instances' with the instruction 'To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered, specify a different port.' At the bottom right are 'Cancel' and 'Previous' buttons.

The screenshot shows the 'Step 6: Review' page of the AWS CloudFront configuration wizard. The top navigation bar includes links for '1. Configure Load Balancer', '2. Configure Security Settings', '3. Configure Security Groups', '4. Configure Routing', '5. Register Targets', and '6. Review' (which is underlined in blue). The main content area is titled 'Step 6: Review' and includes sections for 'Tags' and configuration details:

- Security groups:** Security groups: sg-011051645c70cde (highlighted with a yellow circle)
- Routing:** Target group: New target group  
Target group name: T01  
Port: 80  
Target type: instance  
Protocol: HTTP  
Health check protocol: HTTP  
Path: /  
Health check port: traffic port  
Healthy threshold: 5  
Unhealthy threshold: 2

At the bottom right are 'Cancel' and 'Next Step' buttons.

AWS Services Resource Groups N. Virginia

Volumes Snapshots Lifecycle Manager NETWORK & SECURITY Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces LOAD BALANCING Load Balancers Target Groups AUTO SCALING Launch Configurations Auto Scaling Groups

Create target group Actions

Name Port Protocol Target type Load Balancer VPC ID Monitor

TG1 80 HTTP Instance Application ip-7a99030

Registered targets

Instance ID	Name	Port	Availability Zone	Status	Description
i-0b1647a6c5a8f5c	ELB SERVER2	80	us-east-1a	initial	Target registration is in progress.
i-0802029690718	ELB SERVER1	80	us-east-1a	initial	Target registration is in progress.

Availability Zones

Availability Zone	Target count	Healthy?
us-east-1a	1	No (Availability Zone contains no healthy targets)
us-east-1b	1	No (Availability Zone contains no healthy targets)



## **Practical 20:**

### **How to handle a cloud shell. Explain it**

#### **Steps to use AWS Cloud shell:**

Using AWS CloudShell is a convenient way to access the AWS Command Line Interface (CLI) and various AWS services directly from your web browser,

Here are the steps to use AWS CloudShell:

#### **Login to the AWS Management Console:**

Ensure you have an AWS account and are logged into the AWS Management Console.

#### **Access AWS CloudShell:**

Once you're logged in, you can access AWS CloudShell from the AWS Management Console. You can find it in the top-right corner of the AWS Management Console, labeled as "AWS CloudShell."

#### **Initialize the Environment:**

The first time you access CloudShell, it may take a moment to initialize your environment. Once it's ready, you'll be presented with a command-line interface.

#### **Use the AWS CLI and AWS SDKs:**

CloudShell comes pre-configured with the AWS CLI and various AWS SDKs.

1. You can use these tools to interact with AWS services. For example, you can run AWS CLI commands, Python scripts, or
2. use any of the supported SDKs to manage your AWS resources.  
Customize Your Environment (optional): You can customize your CloudShell environment by installing additional packages
3. or configuring your shell as per your preferences. You can use package managers like pip, npm, or brew to install

**Save Your Work:** AWS CloudShell provides you with home directory storage that is persistent, even across sessions. This means you can save your scripts, configuration files, and other data within your home directory.

**Exit CloudShell:** When you're done with your session, you can type exit to exit CloudShell. Your home directory data will persist for the next time you log in.

## Practical 21:

### Create a private cloud on google drive and grant permission for the user.

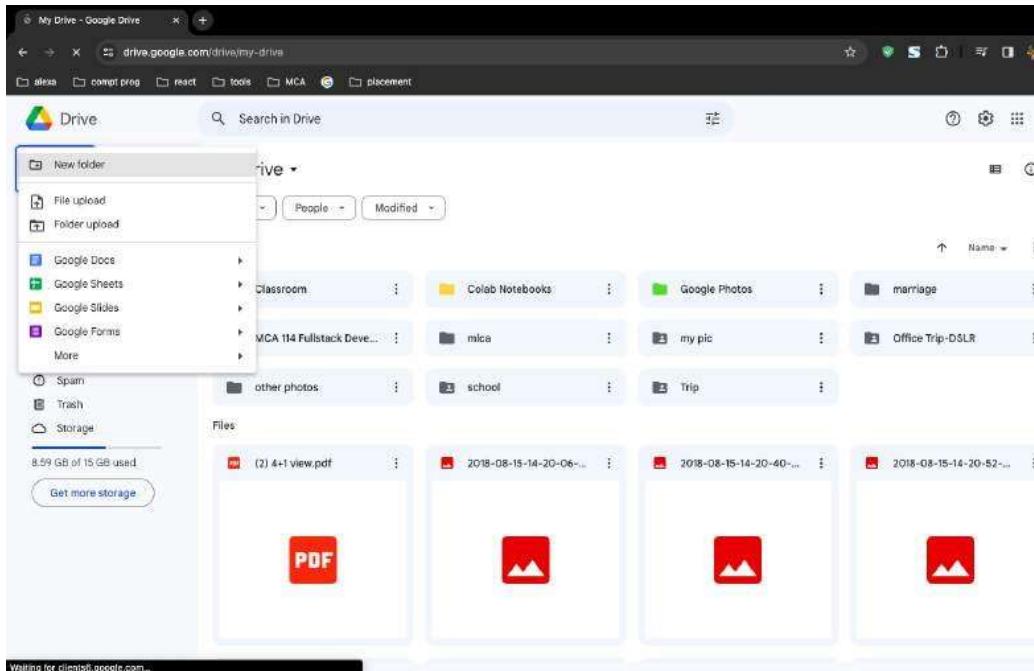
#### Steps to Create a Private Cloud on Google Drive:

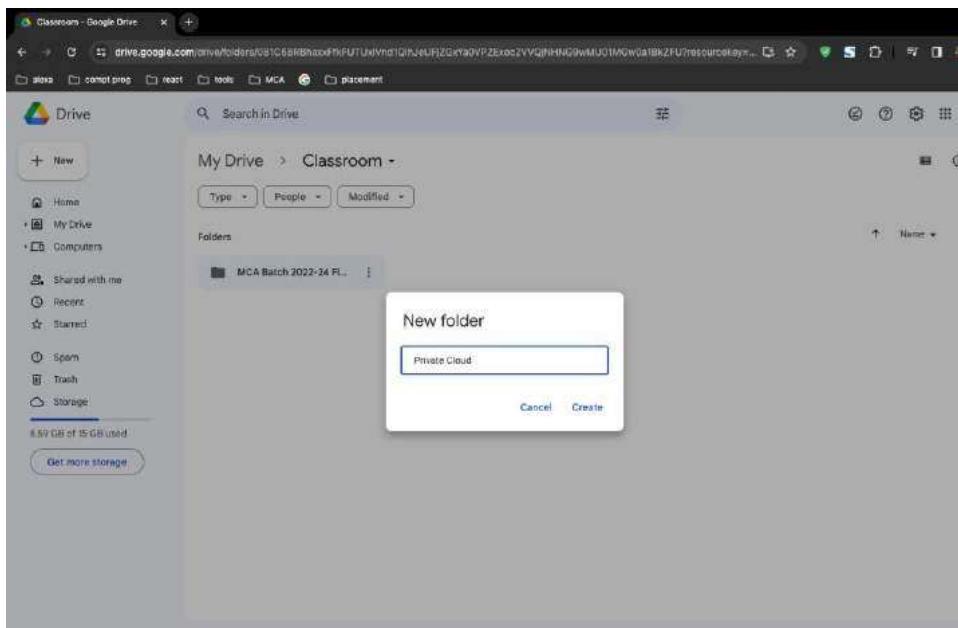
##### 1. Sign in to Google Drive:

Open your web browser and go to Google Drive.  
Sign in with your Google account or create one if you don't have it.

##### 2. Create a New Folder:

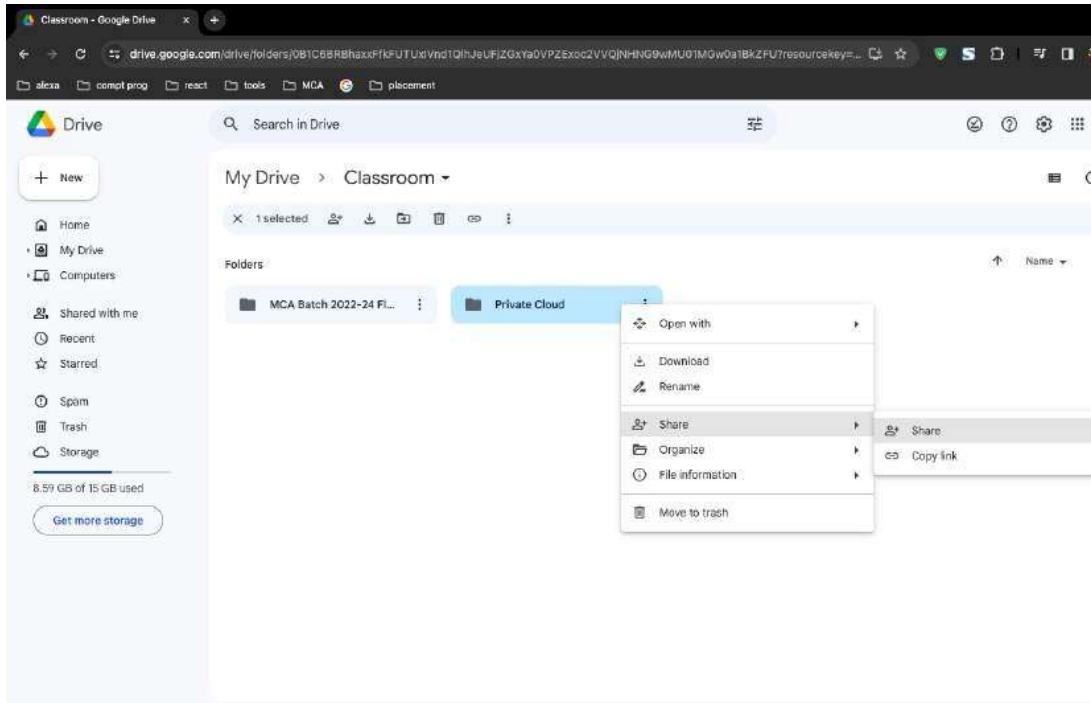
Click on the "+ New" button on the left side.  
Choose "Folder" to create a new folder.  
Name the folder appropriately, e.g., "Private Cloud."





### 3. Share the Folder:

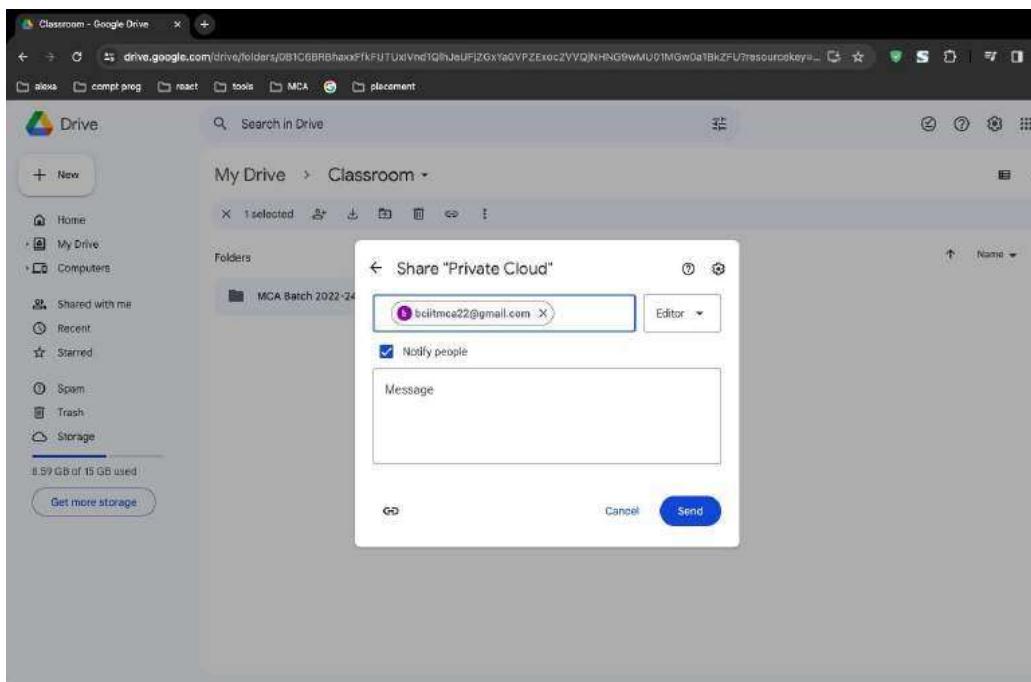
Right-click on the folder you just created.  
Select "Share."



#### 4. Add Users:

In the sharing dialog, enter the email addresses of the users you want to grant access to.

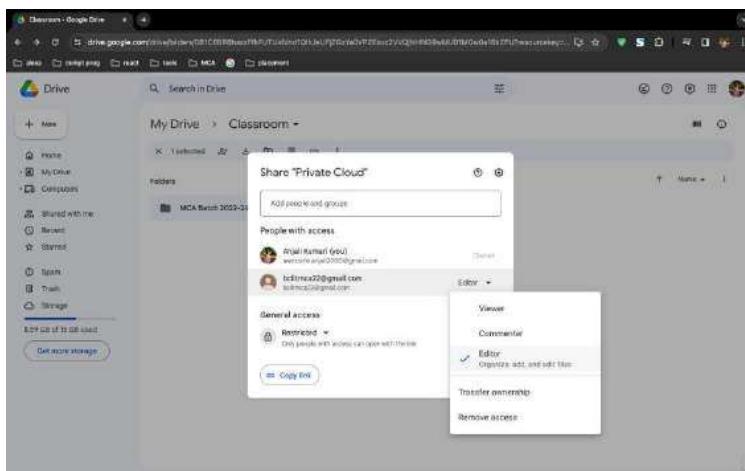
Choose the appropriate access level (e.g., Viewer, Commenter, Editor) based on the level of access you want to provide.



#### 5. Configure Advanced Settings (Optional):

Click on "Advanced" in the sharing dialog.

Adjust settings like link sharing, preventing editors from changing access, etc.



## **6. Send Invitations:**

Click on "Send" to send invitations to the specified email addresses.

Users will receive an email notification and can access the shared folder through their Google Drive.

## **7. User Access Management:**

As the owner, you can manage access at any time by right-clicking on the folder, selecting "Share," and modifying permissions.

