



**BANARSIDAS CHANDIWALA INSTITUTE OF INFORMATION
TECHNOLOGY**

**Affiliated To Guru Gobind Singh Indraprastha University SECTOR
16-C ,DWARAKA ,NEW DELHI**



**PRACTICAL FILE
SUBJECT : CLOUD COMPUTING**

SUBMITTED BY:
Abhishek Tyagi
03311104422

SUBMITTED TO:
MR. ALOK MISRA
Asst. Professor

Index

Sr. no	Practical	Pageno
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	66
18	Create a VPC and implement EC2 services on it	70
19	Implement and configure load balancing with all necessary steps	74

20	How to handle a cloud shell explain it	79
21	Create a private cloud on Google Drive and Grant restrict permissions for the user	80

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:

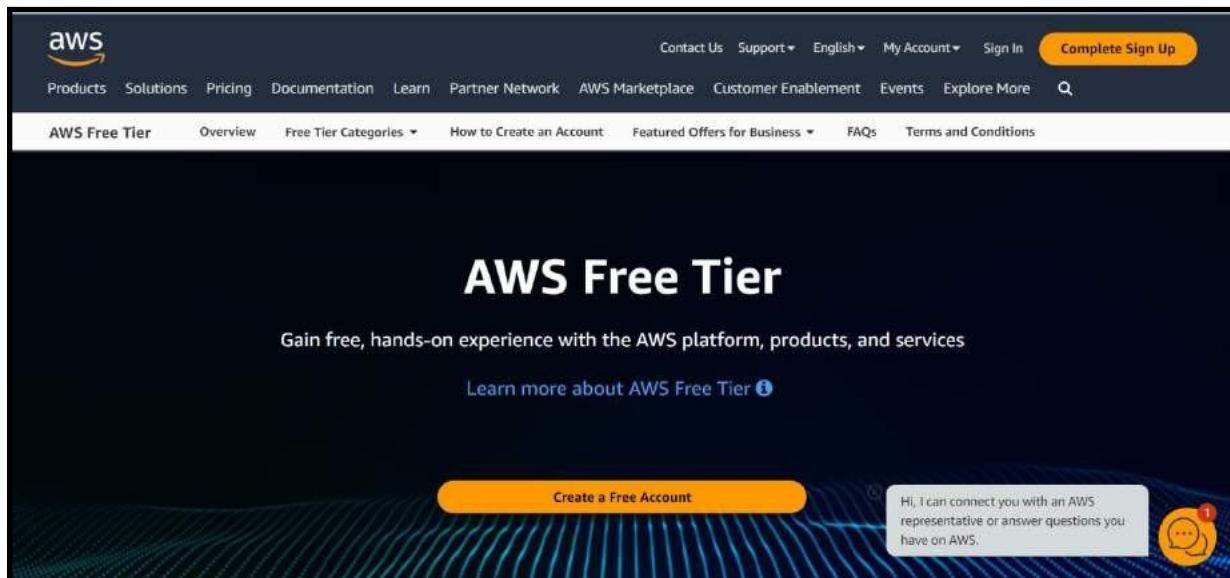
- Access AWS Services: Gain access to a vast array of cloud services, from virtual servers (EC2) to scalable storage (S3) and advanced machine learning (SageMaker).
- Scalability: Easily scale your infrastructure and resources up or down to meet your changing needs, ensuring that you only pay for what 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

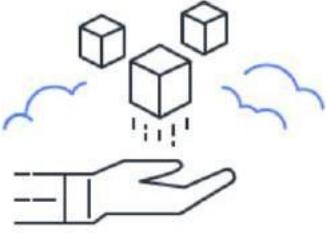
- **Email address:** Provide the mail id which hasn't been registered yet with Amazon AWS.
- **Password:** Type your password.
- **Confirm password:** Authenticate the password.
- **AWS Account name:** Choose a name for your account. You can change this name in your account settings after you sign up



Sign up for AWS

Explore Free Tier products with a new AWS account.

To learn more, visit aws.amazon.com/free.



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).

Sign up for AWS

Confirm your identity

Before you can use your AWS account, you must verify your phone number. When you continue, the AWS automated system will contact you with a verification code.

How should we send you the verification code?

Text message (SMS)
 Voice call

Country or region code

India (+91)

Mobile phone number

A phone number is required.

Security check

Zpb69t

Type the characters as shown above

Send SMS (step 4 of 5)

Step 5 :

Enter your Purpose of Account Registration

Sign up for AWS

Confirm your identity

Primary purpose of account registration

Choose one that best applies to you. If your account is tied to a business, select the one that applies to your business.

Academic

Ownership type

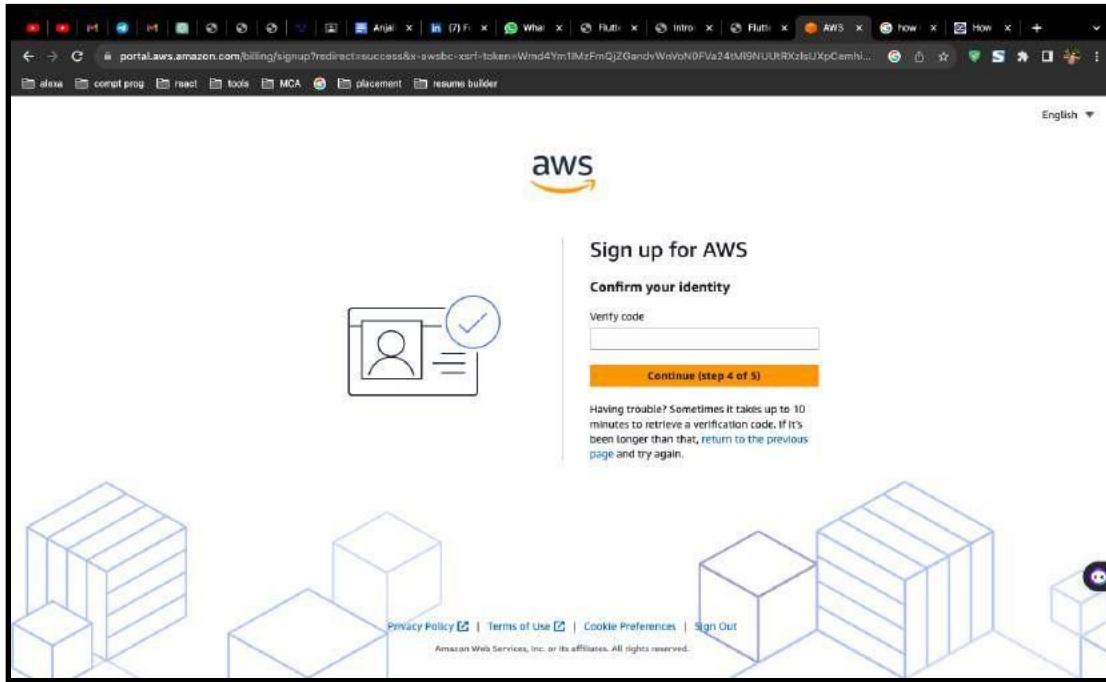
Individual

Continue (step 4 of 5)

Privacy Policy | Terms of Use | Cookie Preferences | Sign Out

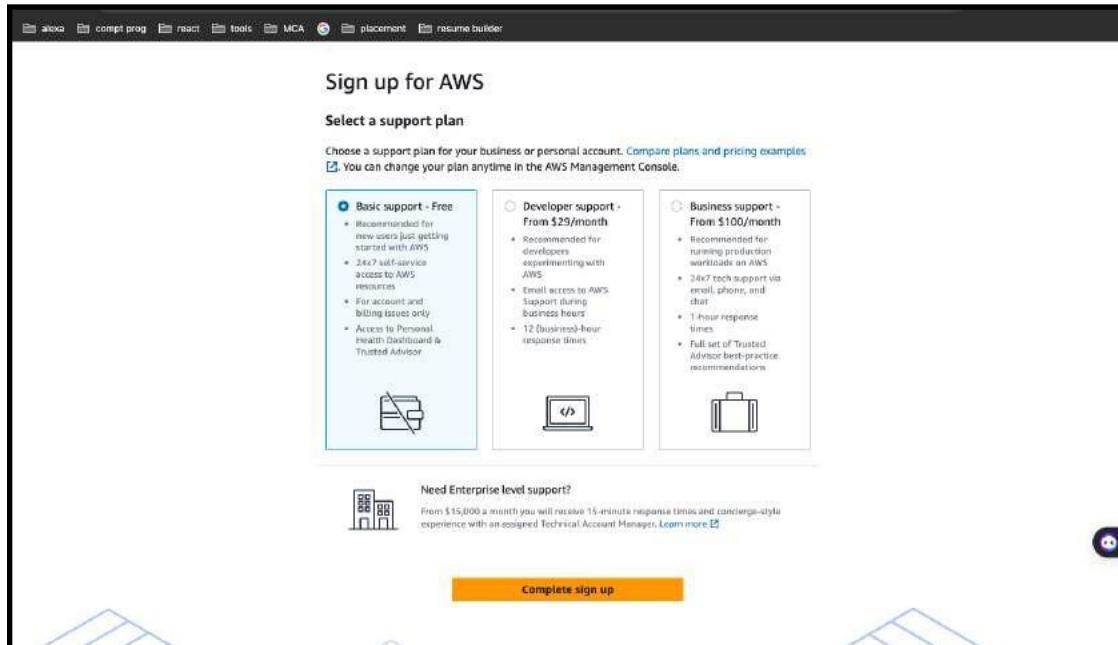
Step 6:

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:

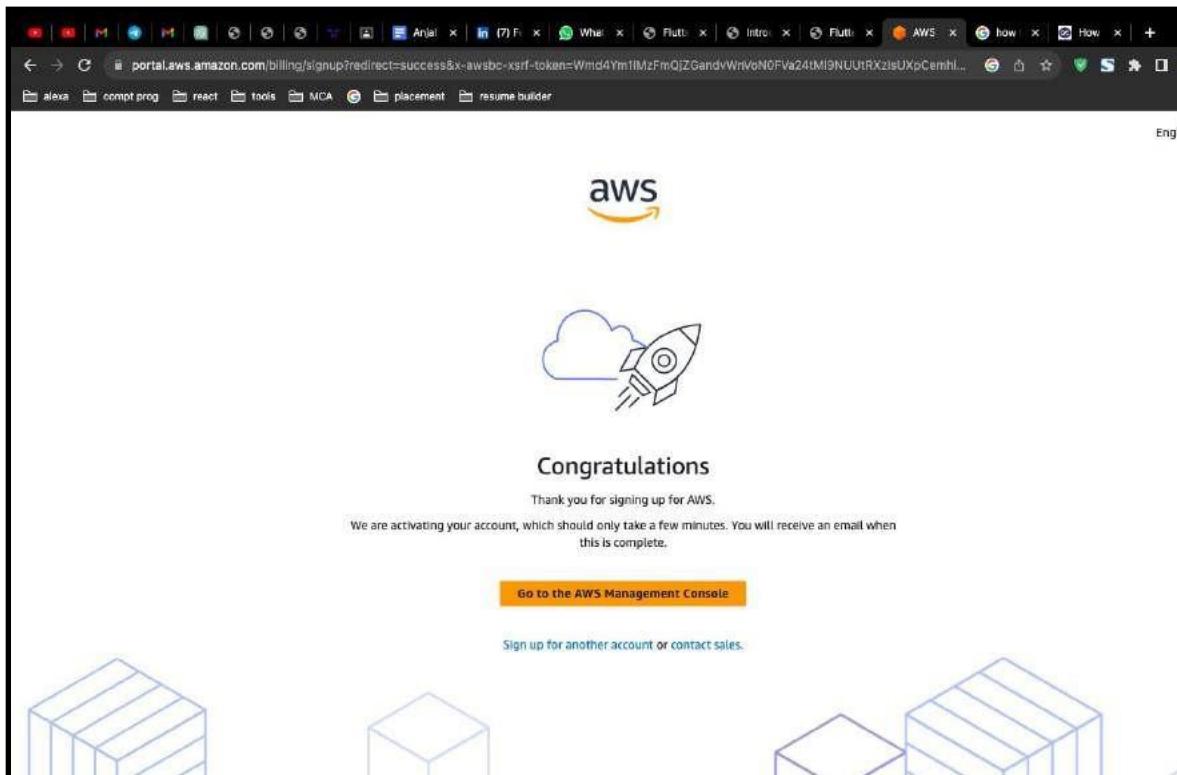
Support plan: AWS support offers a selection of plans to meet your business needs. Select your suitable plan then click continue.



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:

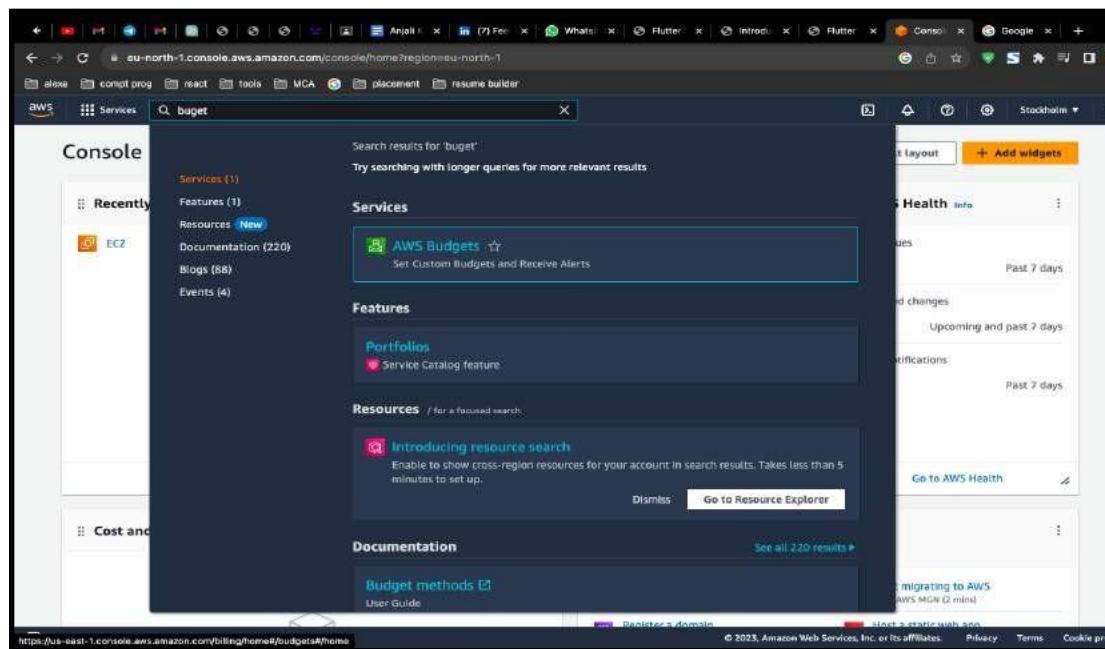
- Cost Monitoring: AWS Budgets provides insights into your AWS cost and usage data, allowing you to monitor your spending in real-time. This helps you keep track of your expenses and ensure they align with your budgetary goals.
- Budget Setting: The service allows you to set specific budgets for different 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. The main page is titled 'AWS Budgets' and features a large heading 'Set custom budgets that alert you when you exceed your budgeted thresholds'. Below this, a section titled 'How it works' contains a flowchart illustrating the budget creation process: 'AWS Budgets - Use free planning and fast set up for tracking' leads to 'Create a budget - Define how much to spend' which then leads to 'Get alerted - Receive email or SMS notifications when you exceed your budget'. 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' link and copyright information.

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. On the left, the navigation bar shows 'Billing' 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 expanded, showing fields for 'Budget name' (containing 'My Zero-Spend Budget') and 'Email recipients' (a placeholder field). The bottom of the screen includes standard AWS footer 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.

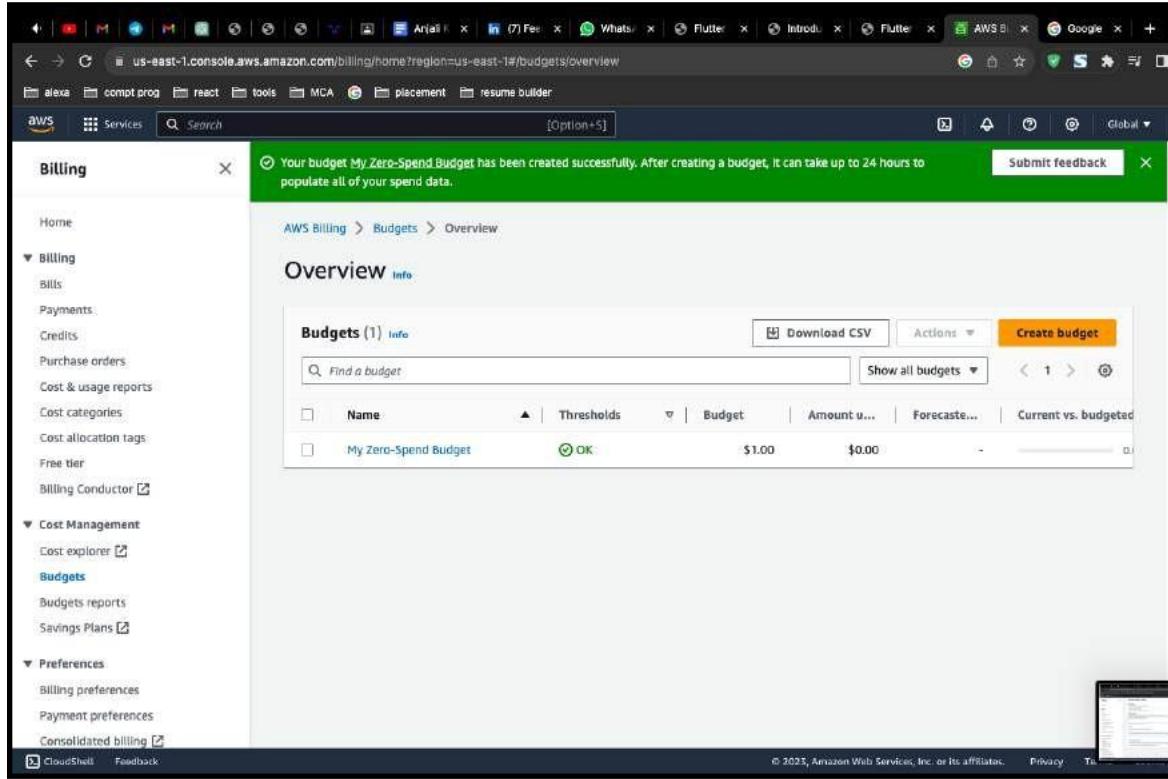
(Optional) Under Notification preferences, for Email recipients, enter the email addresses that you want the alert to notify. Separate multiple email addresses with commas. A notification can be sent to a maximum of 10 email addresses.

(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 interface. On the left, there's a sidebar with 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 area is titled "Zero spend budget - Template". It has fields for "Budget name" (containing "My Zero-Spend Budget") and "Email recipients" (empty). A note below says "You will be notified via email when any spend above \$0.01 is incurred." At the bottom right, there are "Cancel" and "Create budget" buttons. The footer includes links for CloudShell, Feedback, Privacy, Terms, and Cookies.

Step 6 :

Your Aws Budget is Created



The screenshot shows the AWS Billing Overview page. A green success message at the top states: "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 this, the "Overview" section displays a table titled "Budgets (1) Info". The table has columns for Name, Thresholds, Budget, Amount used, Forecasted, and Current vs. budgeted. One row is shown: "My Zero-Spend Budget" with a status of "OK", a budget of "\$1.00", and actual usage of "\$0.00". The left sidebar shows navigation links for Billing, Cost Management, and Preferences.

Budgets (1) <small>Info</small>					
		Thresholds		Budget	Amount u...
	Name			Forecast...	Current vs. budgeted
	My Zero-Spend Budget	OK		\$1.00	\$0.00

Practical 3:

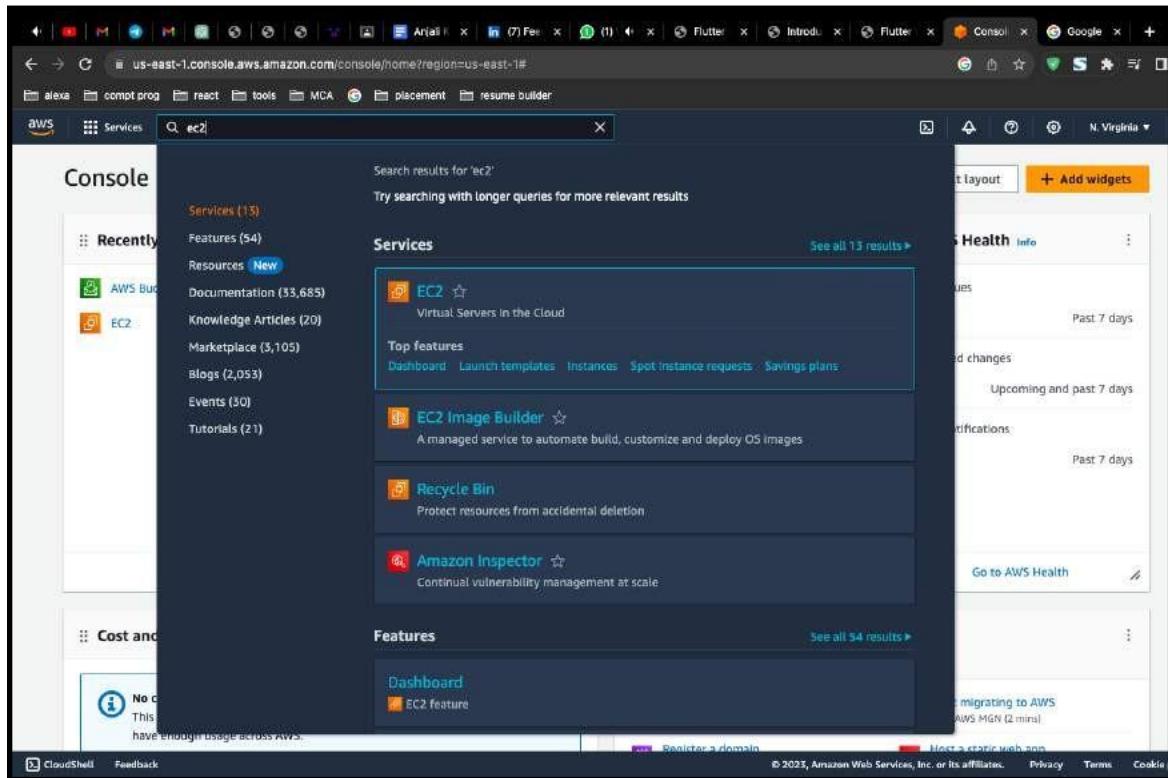
Launch a WINDOW SERVER INSTANCE with t2.micro.instance type and create a security group by using EC2

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.

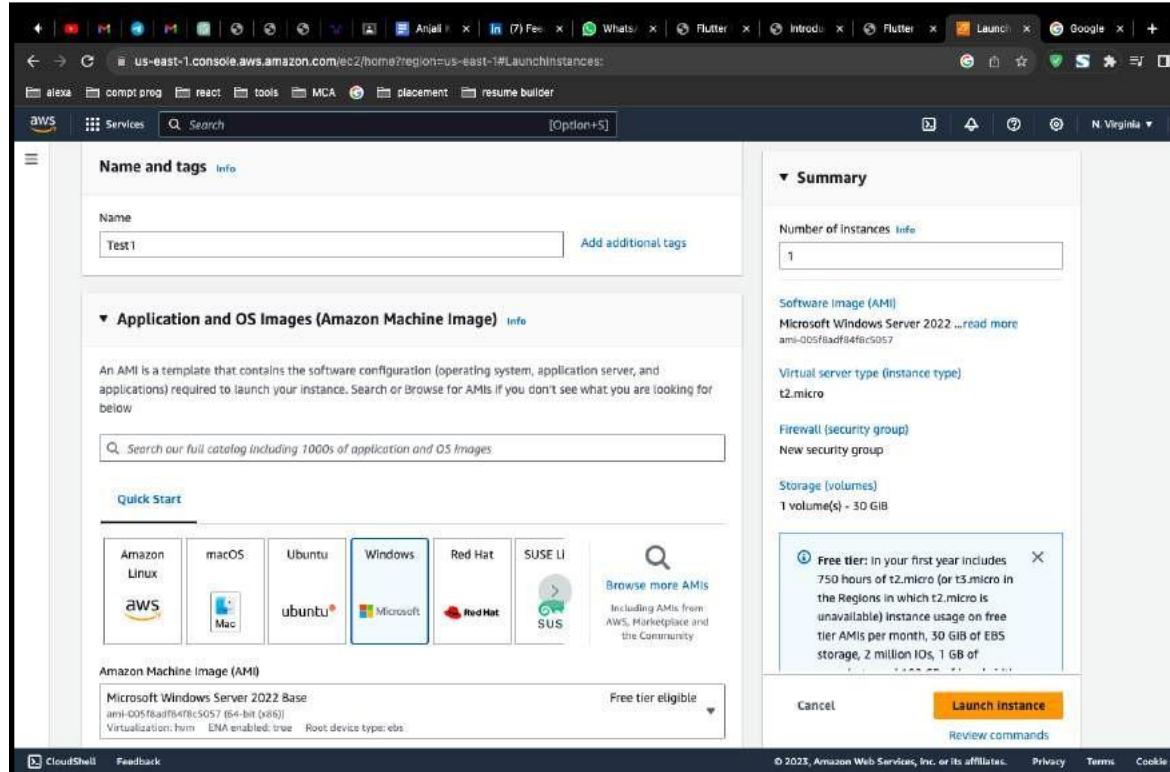


Step 2:

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.

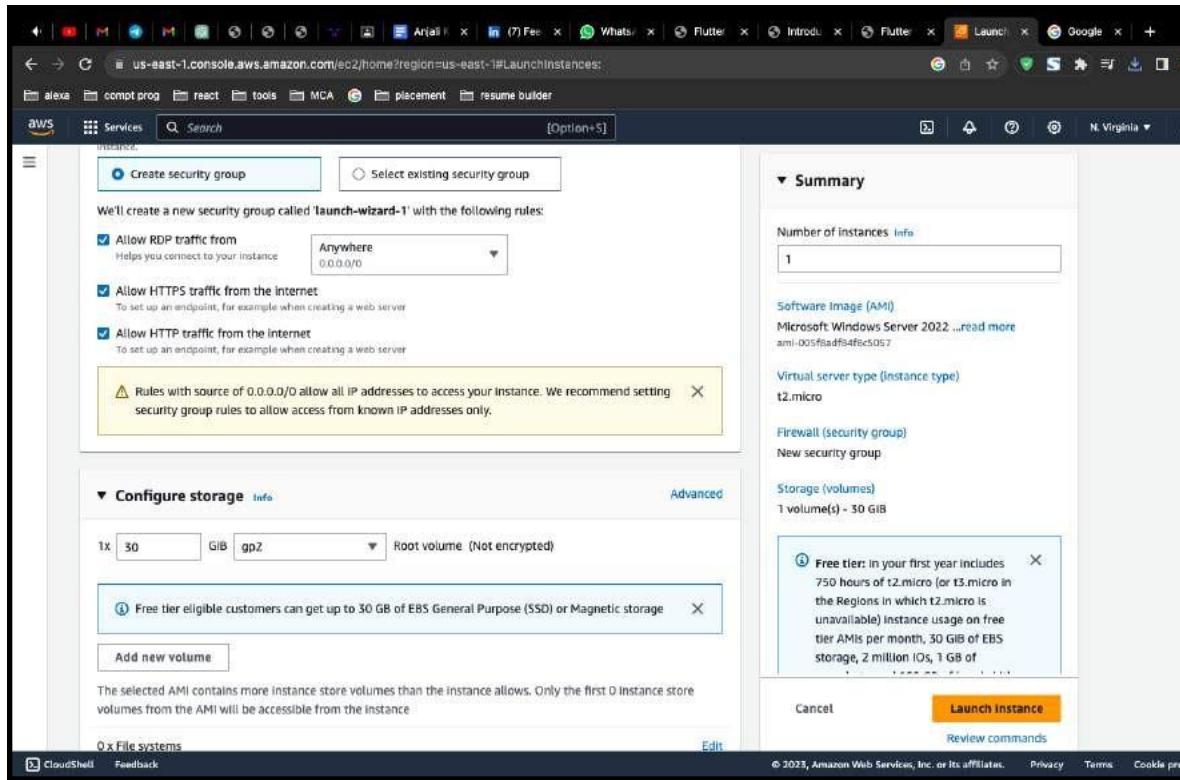


Step 4:

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.

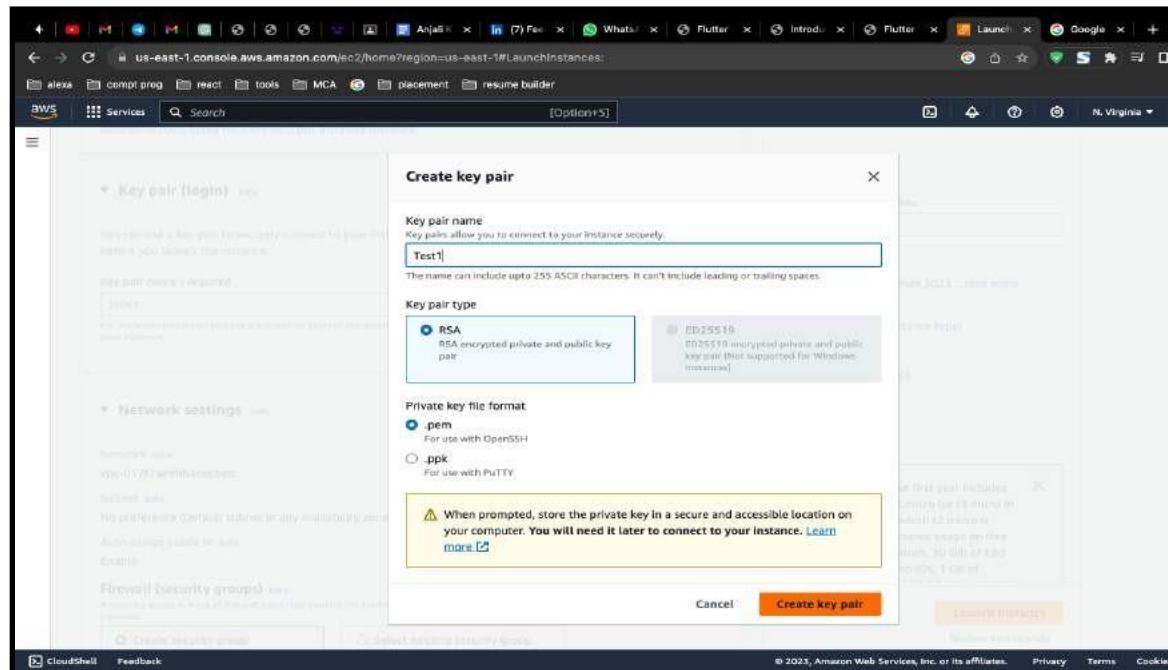
Step 5:

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.



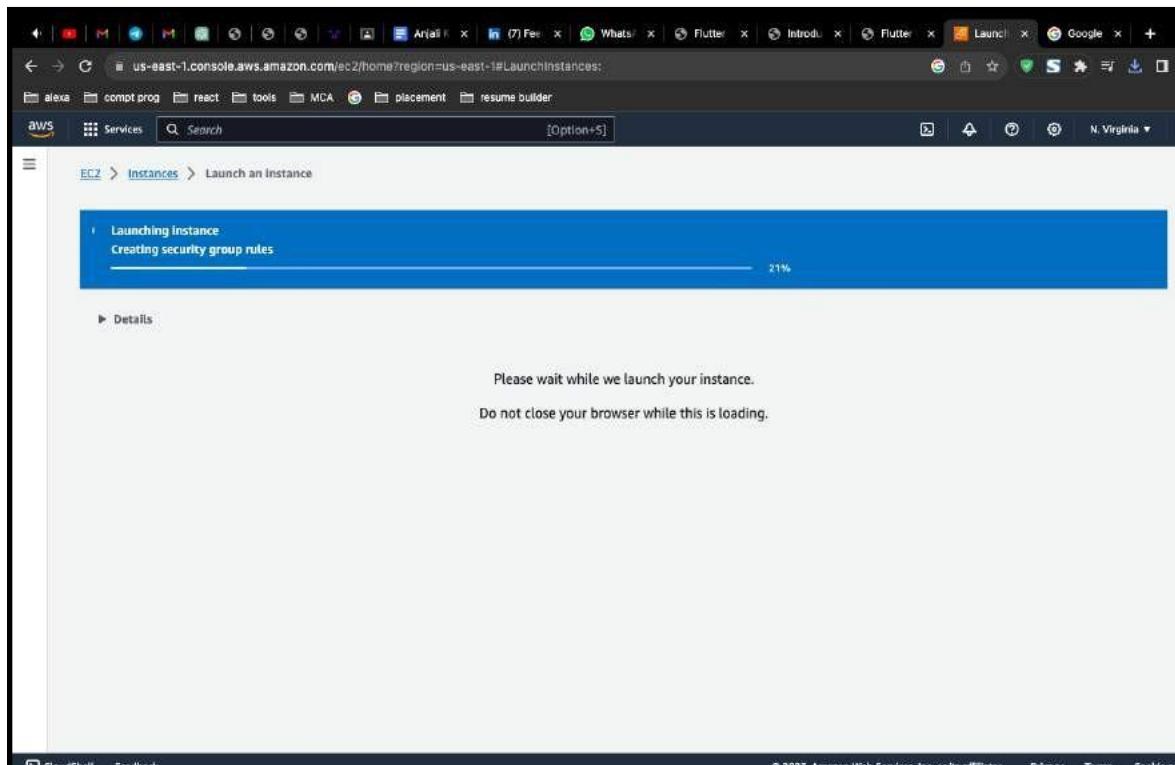
Step 6:

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.



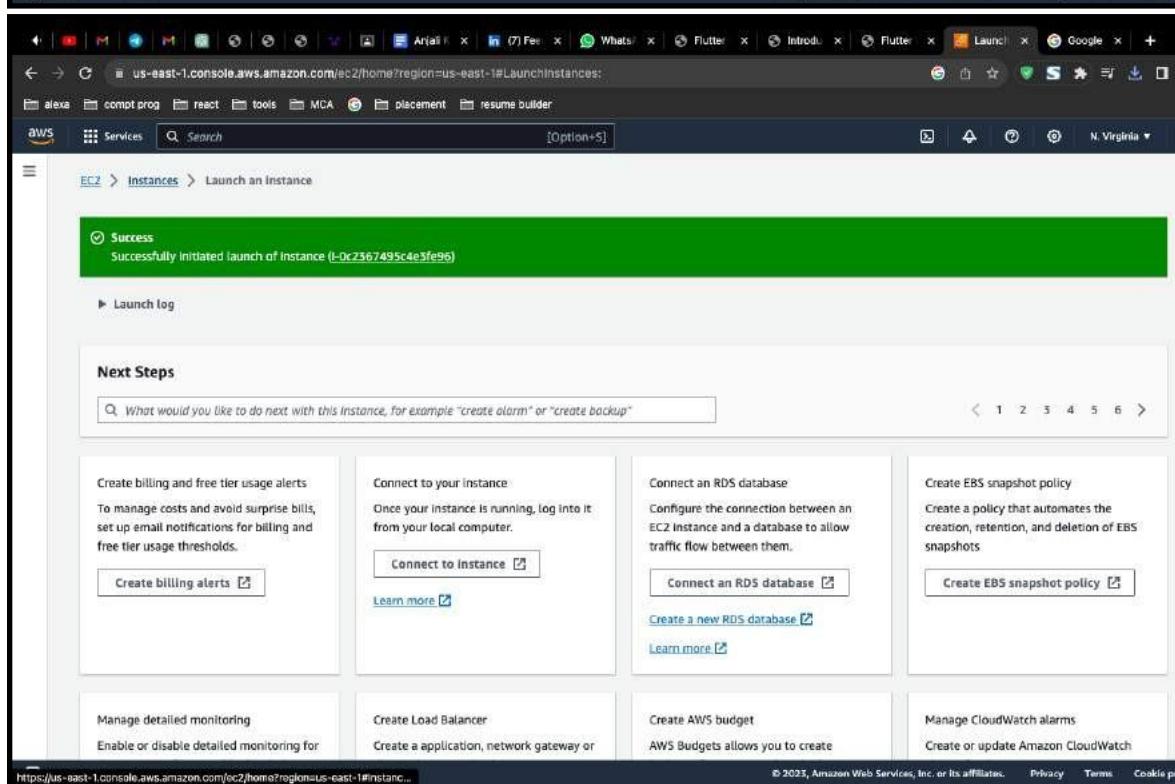
Step 6:

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



The screenshot shows a progress bar indicating the status of launching an instance. The bar is blue and has the text "Launching Instance" and "Creating security group rules" above it. A progress indicator shows "21%" completed. Below the progress bar, there is a link labeled "Details".

Please wait while we launch your instance.
Do not close your browser while this is loading.



The screenshot shows a green success message box with the text "Success" and "Successfully Initiated launch of instance (i-0c2367495c4e5fe96)". Below the message, there is a link labeled "Launch log".

Next Steps

What would you like to do next with this instance, for example "create alarm" or "create backup"?

< 1 2 3 4 5 6 >

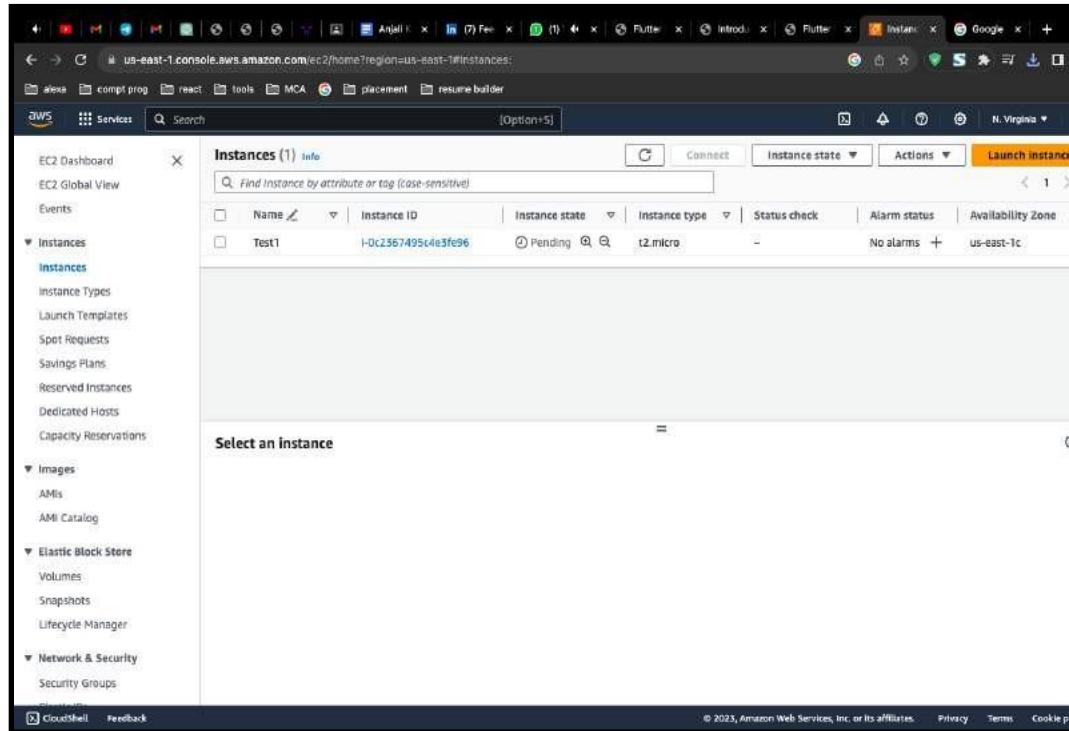
Create billing and free tier usage alerts To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. Create billing alerts	Connect to your instance Once your instance is running, log into it from your local computer. Connect to Instance Learn more	Connect an RDS database Configure the connection between an EC2 instance and a database to allow traffic flow between them. Connect an RDS database Create 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 sidebar with various navigation 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-0c2567495e4e3fe06, Instance state: Pending, Instance type: t2.micro, Status check: -, Alarm status: No alarms, Availability Zone: us-east-1c). There are also 'Connect', 'Actions', and 'Launch instance' buttons. At the bottom, there's a message 'Select an instance'.

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:

Public DNS ec2-54-197-166-238.compute-1.amazonaws.com	User name Administrator
Password Get password	

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 (selected), Images, and Elastic Block Store. The main area displays a table titled "Instances (1) Info". The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. One row is shown for "Test1" with the instance ID "i-0c2367495c4a3fe96", which is currently "Running". The status check shows "2/2 checks passed" and there are no alarms. The instance is located in the "us-east-1c" availability zone. At the top right of the main area, there are buttons for "Connect", "Instance state", "Actions", and "Launch instances". Below the table, there's a section labeled "Select an instance".

Practical 5:

Terminate the launched window server instance from AWS EC2.

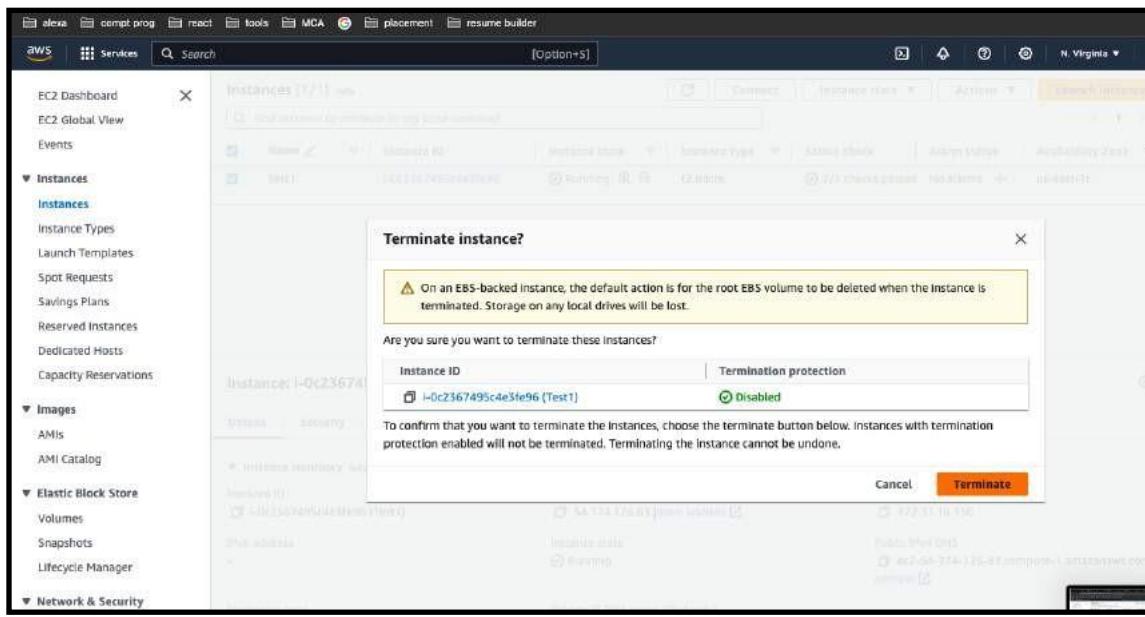
Step 1:

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 has a search bar and a table titled 'Instances (1/1)'. The table shows one instance: 'Test1' (Instance ID: i-0c2367495c4e3fe96, Instance state: Running, Instance type: t2.micro). Below the table, there are buttons for Stop instance, Start instance, Reboot instance, Hibernate instance, and Terminate instance. The 'Terminate instance' button is highlighted with a blue border. At the bottom of the page, there's a summary section for the selected instance (i-0c2367495c4e3fe96) with details like Public IPv4 address (54.174.126.83), Instance state (Running), and Private IP/DNS (172.31.16.150, ec2-54-174-126-83.compute-1.amazonaws.com).

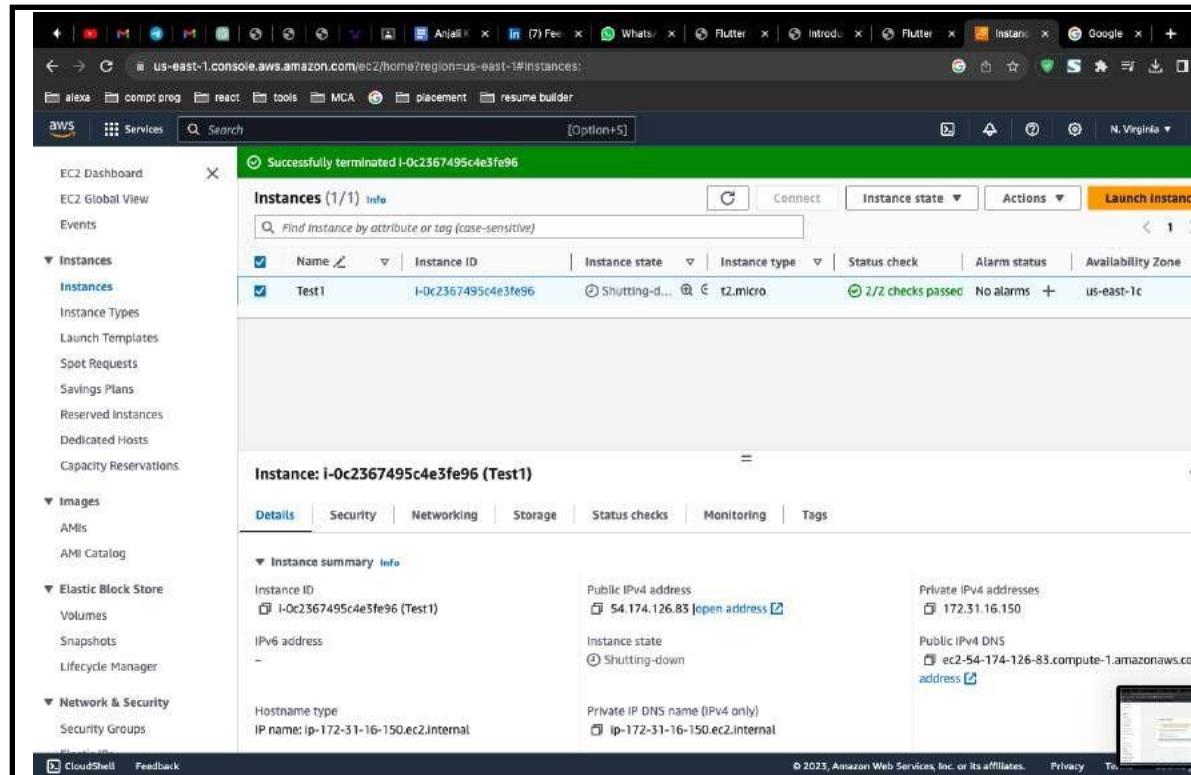
Step 2:

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.

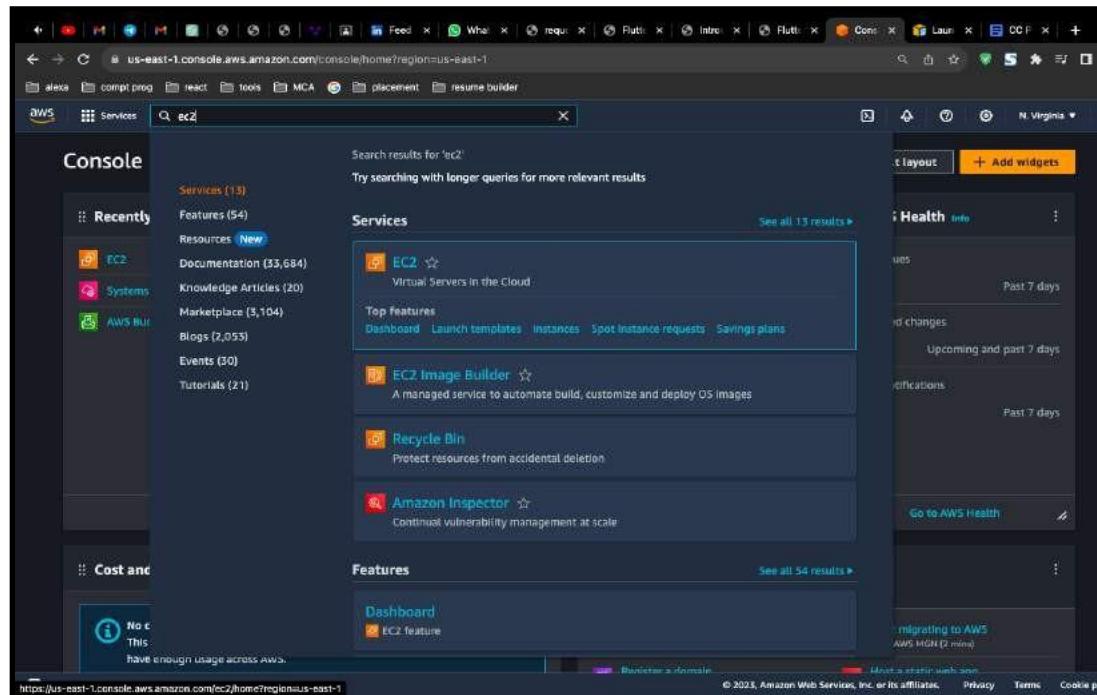


Practical 6:

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

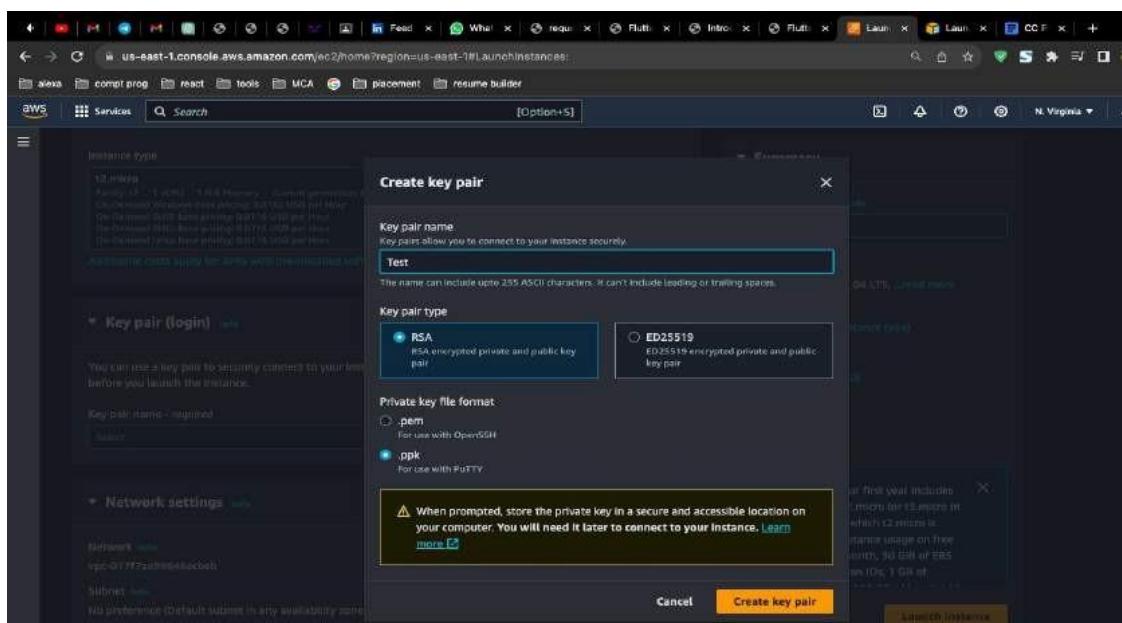


Step 2 :

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.).

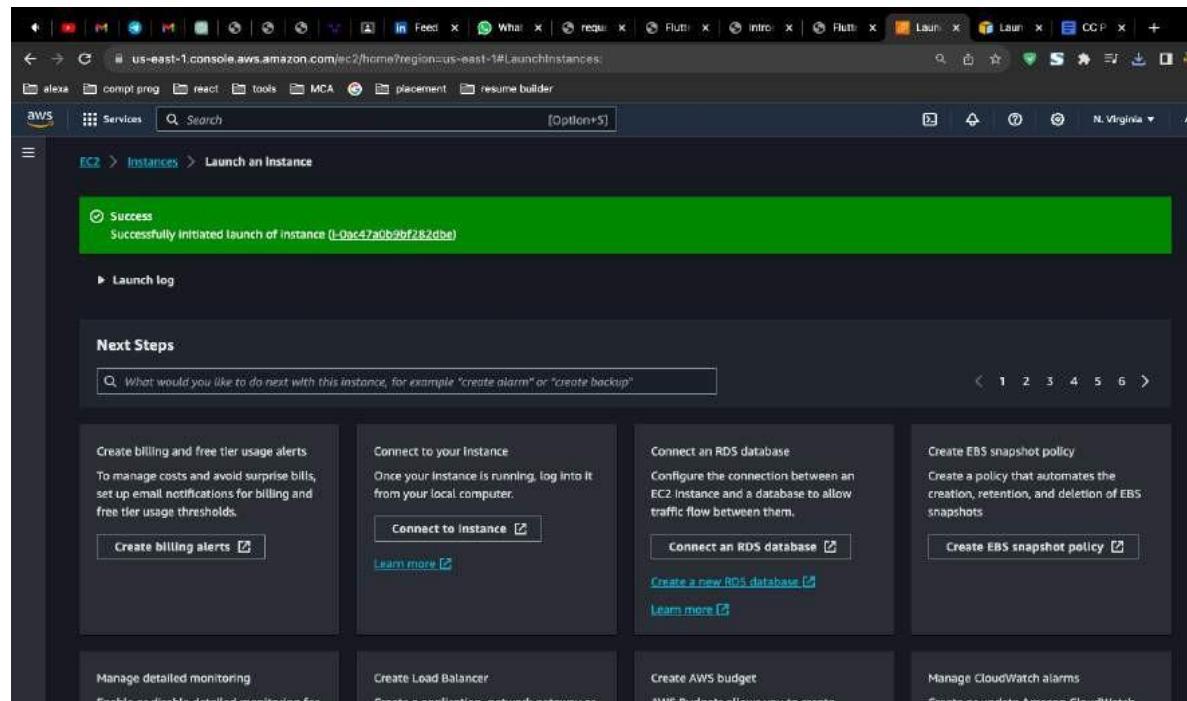
Step 3:

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.

The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main area displays a table titled "Instances (1) info". The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. One row is present, labeled "Test1" with the instance ID "i-0ca27a0b9ef282dbe". The instance state is "Running", the instance type is "t2.micro", the status check is "2/2 checks passed", there are "No alarms", and it is in the "us-east-1d" availability zone.

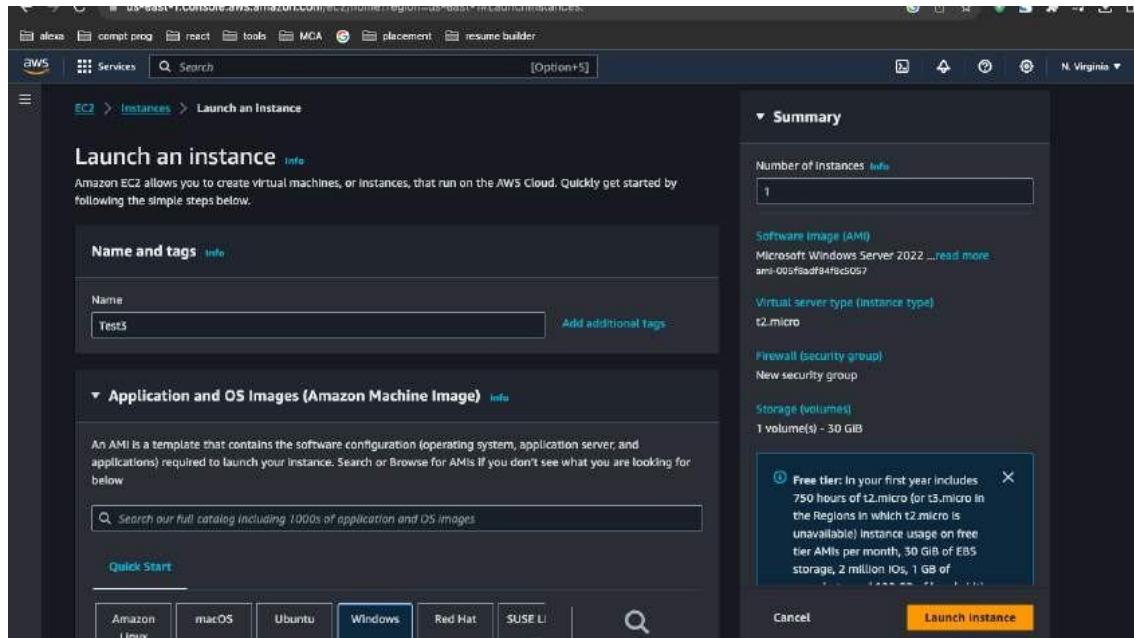
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Test1	i-0ca27a0b9ef282dbe	Running	t2.micro	2/2 checks passed	No alarms	us-east-1d

Practical 7:

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

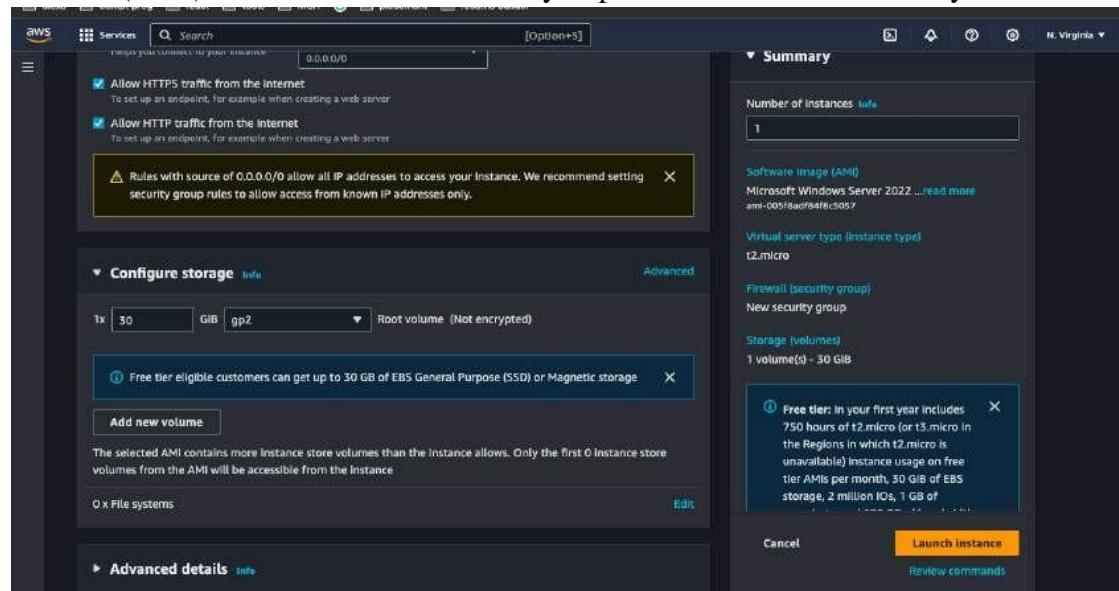
Step 1:

Create a Aws Ec2 Window Instance



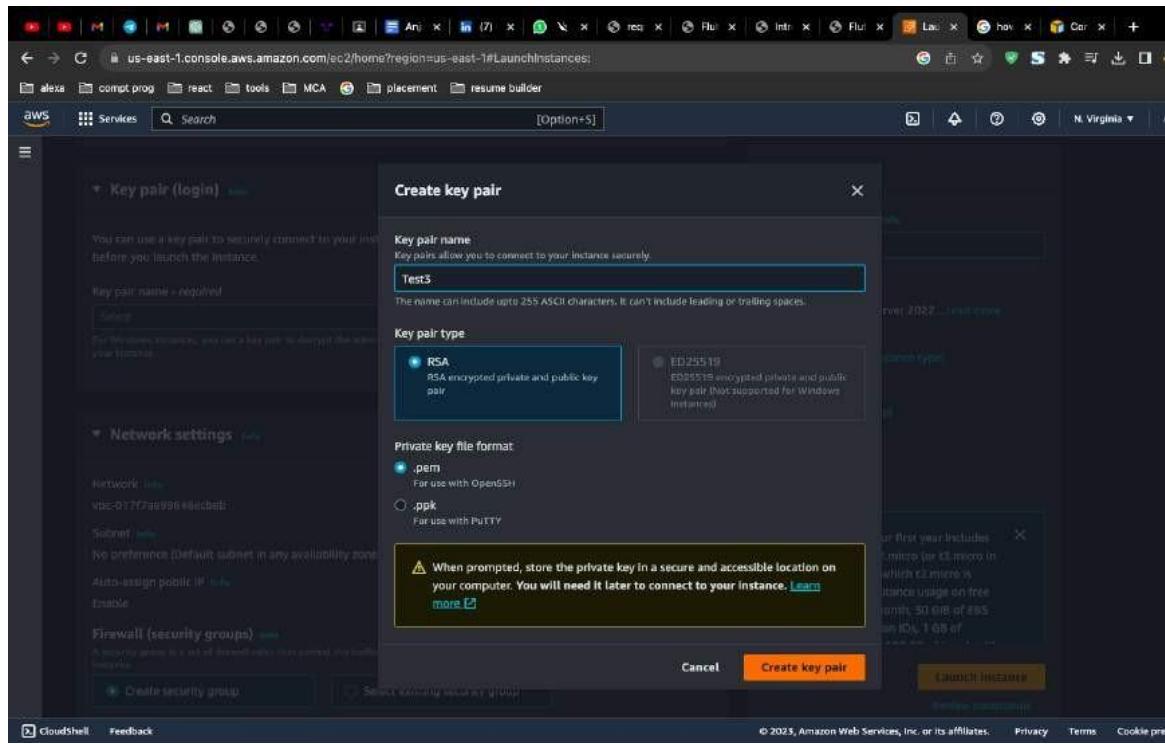
Step 2:

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.



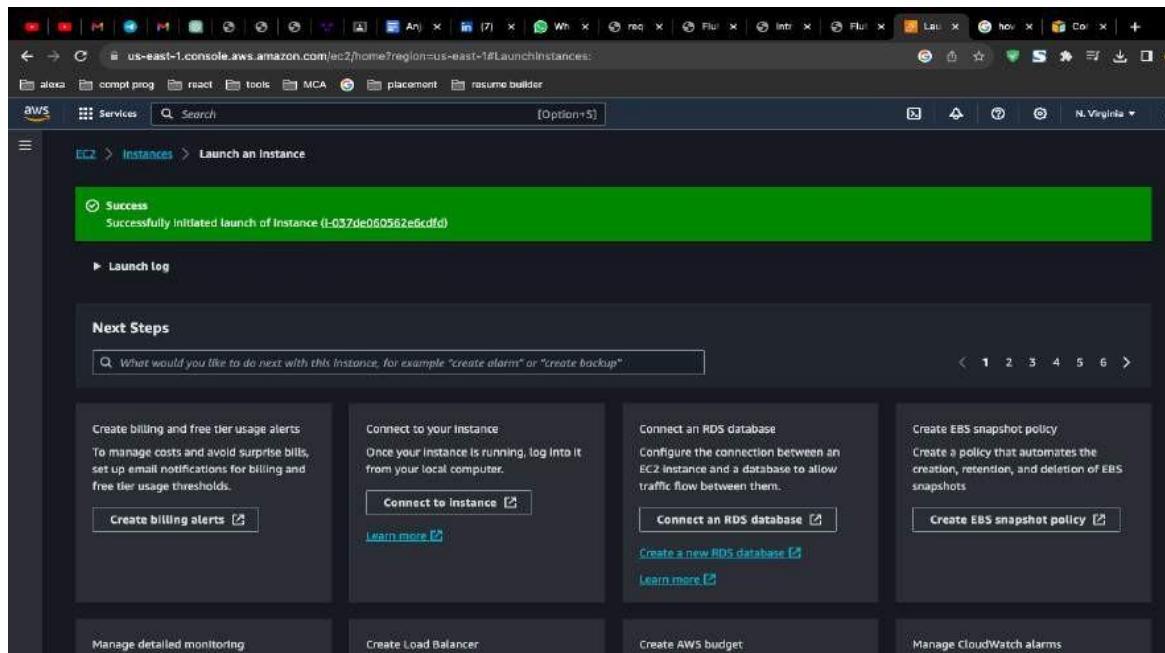
Step 3:

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 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, Network & Security, and Security Groups. The main content area has a header 'Instances (1) info' with a search bar and buttons for Connect, Instance state, Actions, and Launch Instances. Below the header is a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. A single row is shown for 'Test3' with instance ID 'i-037de060562e6cdff', state 'Pending', type 't2.micro', and availability zone 'us-east-1c'. At the bottom, there's a 'Select an instance' dropdown and footer links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Step 5 :

Check Instance Summary

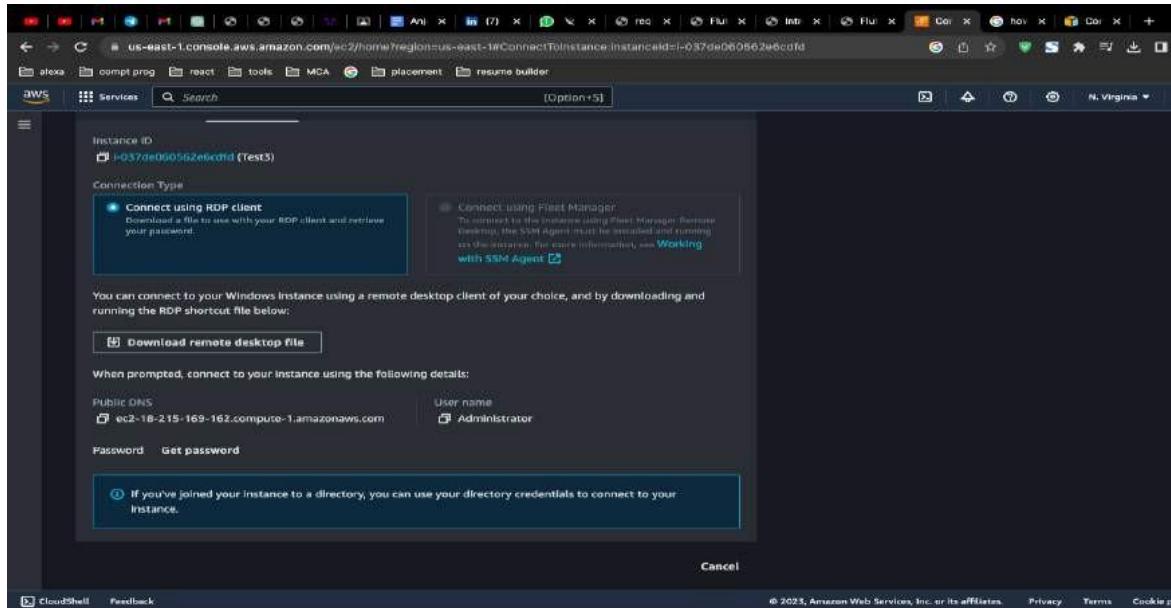
The screenshot shows the AWS EC2 Instance summary page for instance 'Test3'. The left sidebar is identical to the previous screenshot. The main content area shows detailed information for 'Test3'. The 'Details' tab is selected. Key details include:

- Instance ID: i-037de060562e6cdff (Test3)
- Public IPv4 address: 18.215.169.162 [open address]
- Instance state: Running
- Private IP DNS name (IPv4 only): ip-172-51-51-179.ec2.internal
- Instance type: t2.micro
- VPC ID: vpc-012f7ae99646ecbeb
- Subnet ID: subnet-0944dd8d0c7f2ac2

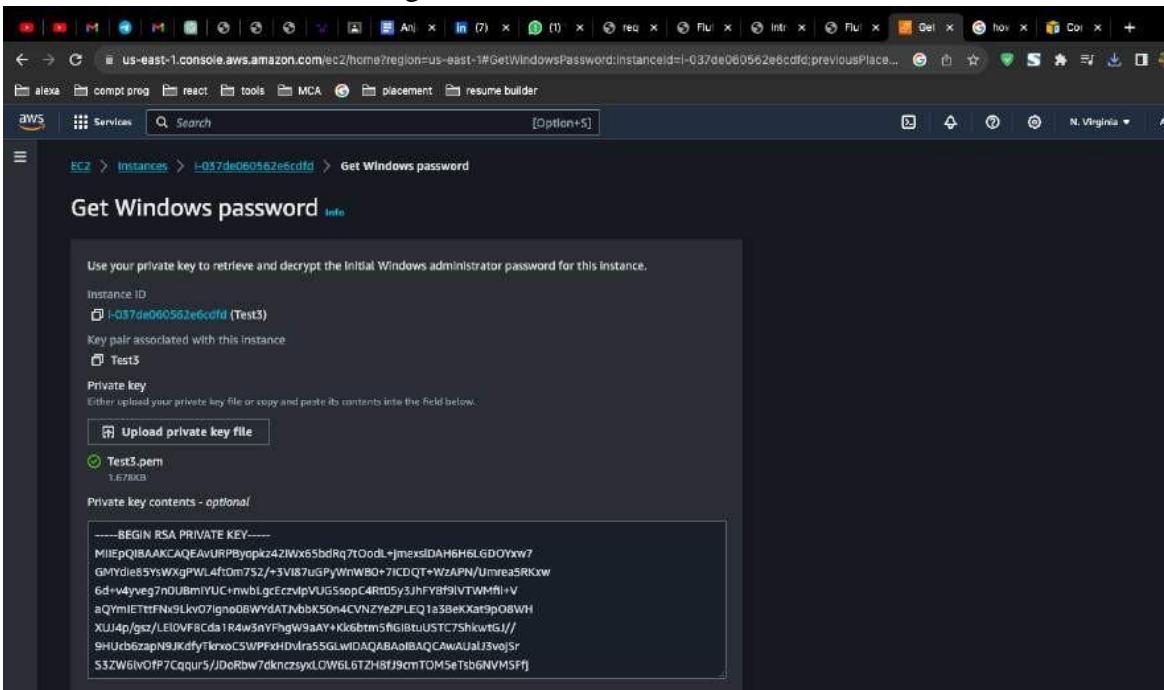
Other tabs available are Security, Networking, Storage, Status checks, Monitoring, and Tags. The footer includes links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

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.



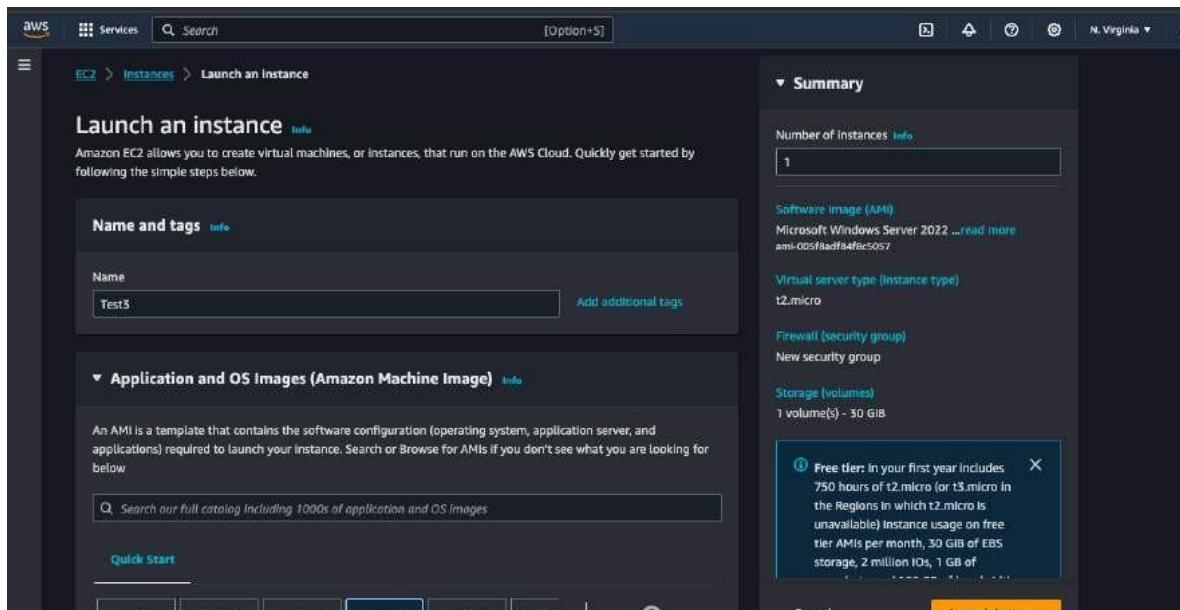
```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpQIBAAKCAQEAvURPBopkz42Iwx8SbdRq7QodL+jmexsIDAH6H6LGDOYxxw/  
GMYdIe85yvXgPWL4fOm752/+3V6t7ugPyWrWBO+7ICDQT+WzAPN/Umrca5RKxw/  
6d+v4yveg7n0U8mUYC+nwblgcEcvlpVUGssopC4RtD5y3jhFY8f9lVTWMfii+V  
aQYmIEttFn9LkvO7ignoBWydATnbk50h4CVNZyeZPLEQ1a38ekXat9p08WVH  
XU4p/gz/LEl0WF8Cda1R4w5nYFngW9AA+Vkk6btm5fGIbtUSTC7shkwGJ//  
9HZcb6zapN9JKdfyTkrxoCSWPFxHDvra55GLwiDAQABAoiBAQCAwAUalJ3vojSr  
S3ZW6lvOP7Cqqr5/jDoRbw7dknczsyxLOW6L6TzH8fJ9cmTOMSeTsb6NVMSFj
```

Practical 8:

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

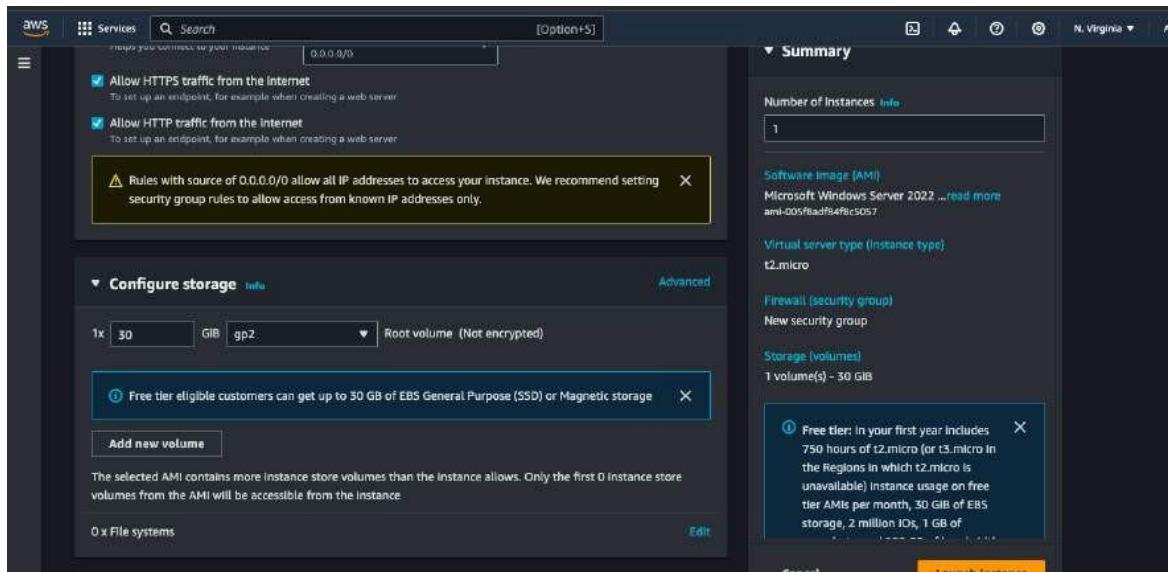
Step 1:

Create a Instance



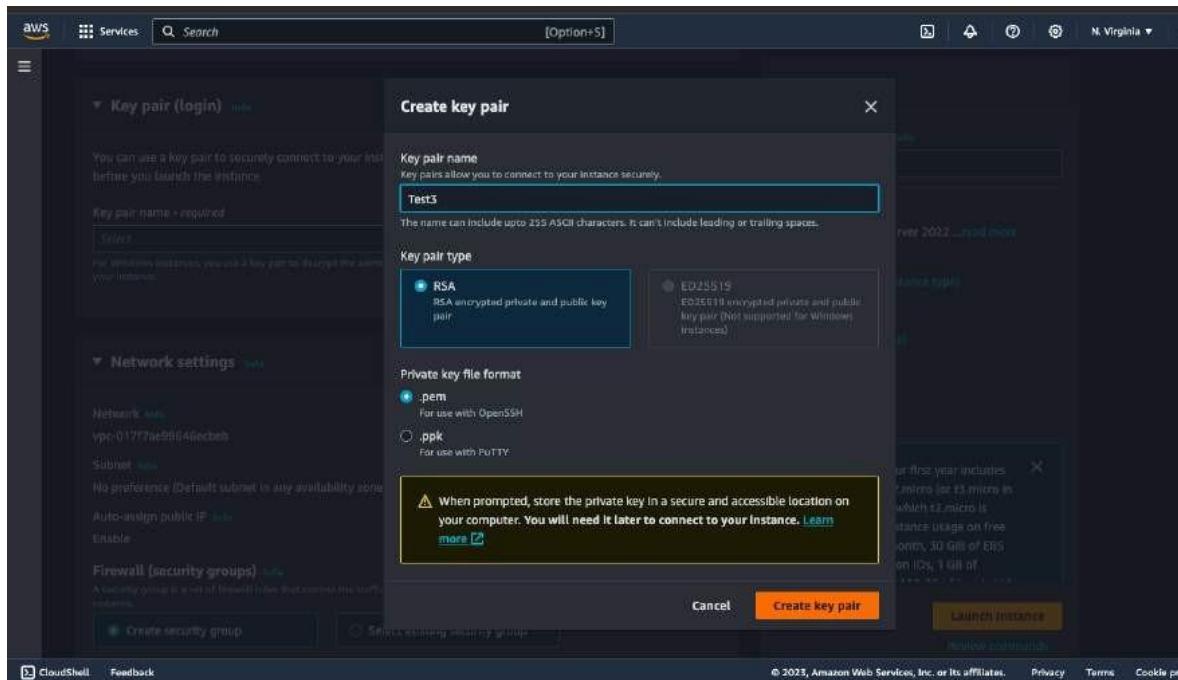
Step 2 :

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.



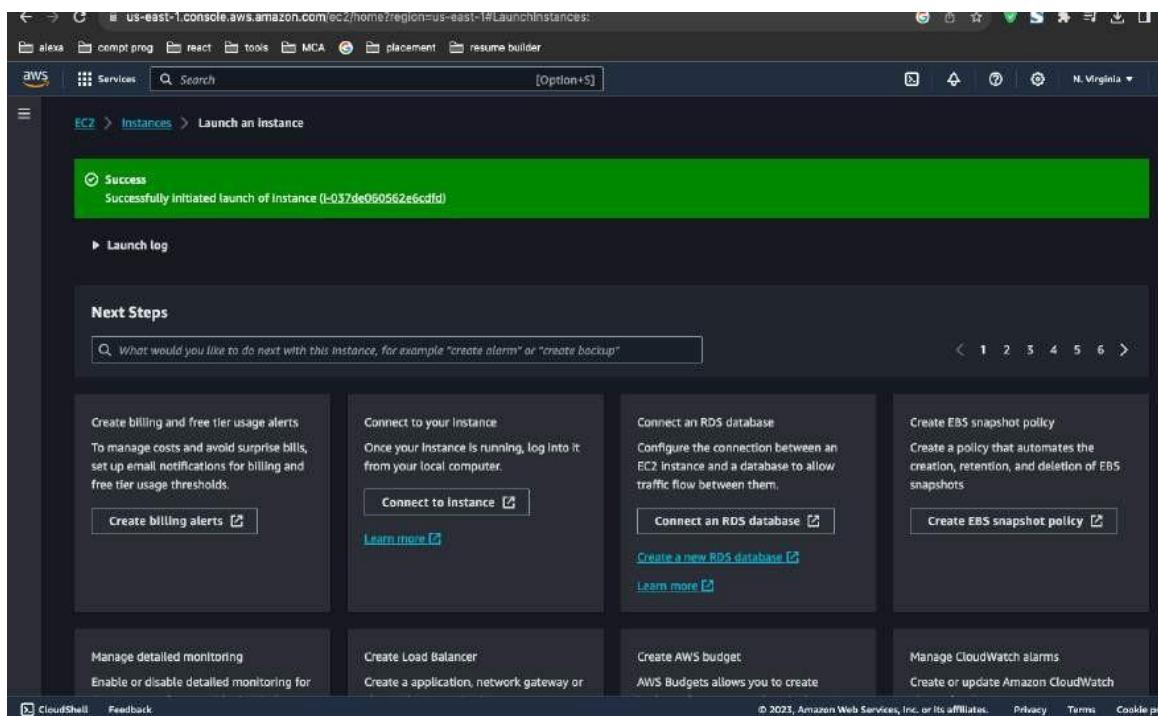
Step 3:

Create a key pair for instance.



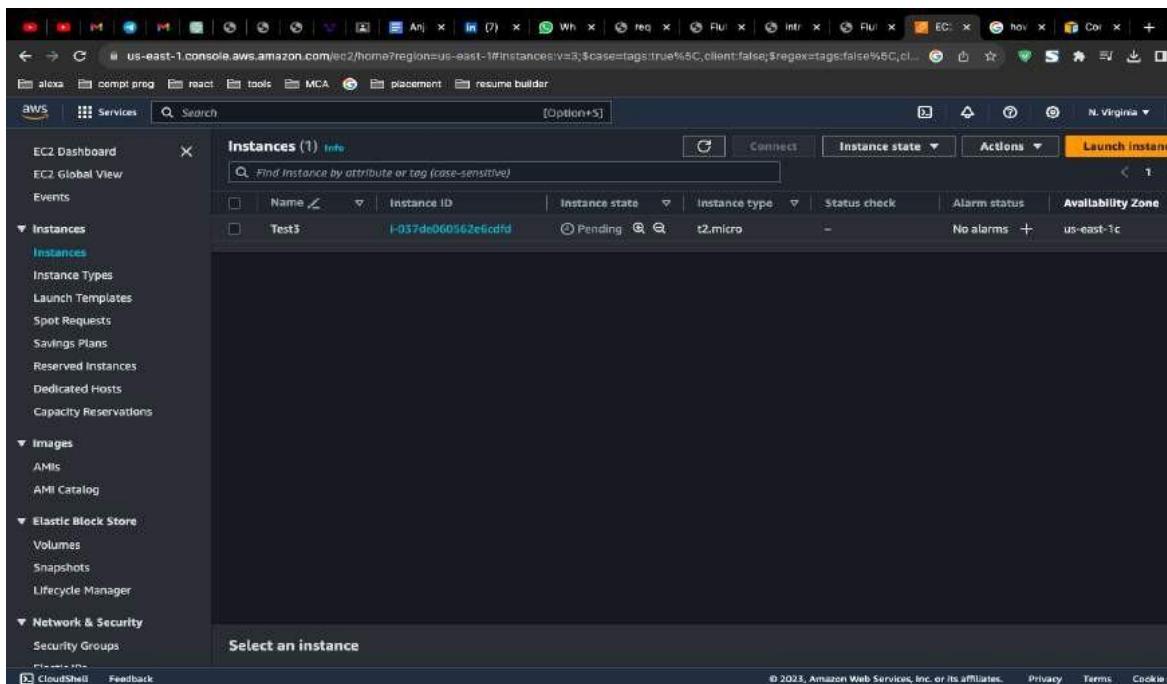
Step 4:

Launch the instance



Step 5 :

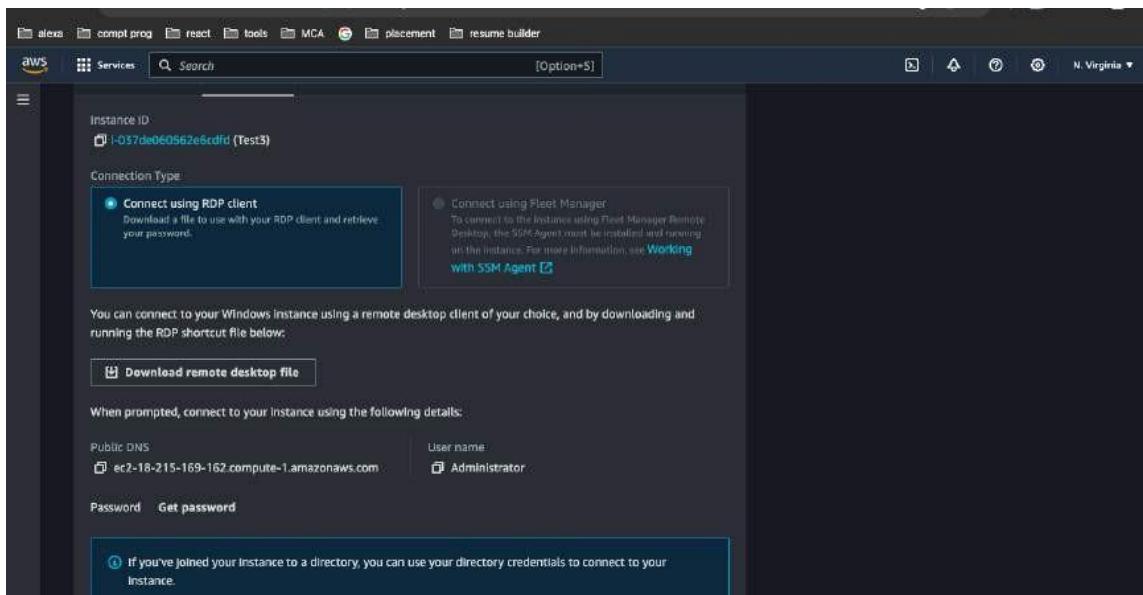
Launch the instance



The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections 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 is titled "Instances (1) info" and shows a table with one row. The row contains the name "Test3", instance ID "i-037de060562e6cdff", state "Pending", type "t2.micro", status check "-", alarm status "No alarms", and availability zone "us-east-1c". There are buttons for "Connect", "Instance state", "Actions", and "Launch instance". At the bottom, there's a "Select an instance" dropdown and some footer links.

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."



The screenshot shows the "Connect using RDP client" page for the instance "i-037de060562e6cdff (Test3)". It has two main sections: "Connection Type" and "Download remote desktop file". The "Connection Type" section is expanded, showing "Connect using RDP client" (selected) and "Connect using Fleet Manager". Below these are instructions for using the RDP client and a note about using Fleet Manager. A "Download remote desktop file" button is present. The "Download remote desktop file" section includes fields for "Public DNS" (set to "ec2-18-215-169-162.compute-1.amazonaws.com") and "User name" (set to "Administrator"). There are "Password" and "Get password" buttons. A note at the bottom states: "If you've joined your instance to a directory, you can use your directory credentials to connect to your instance." The page also includes standard AWS footer links.

Step 7:

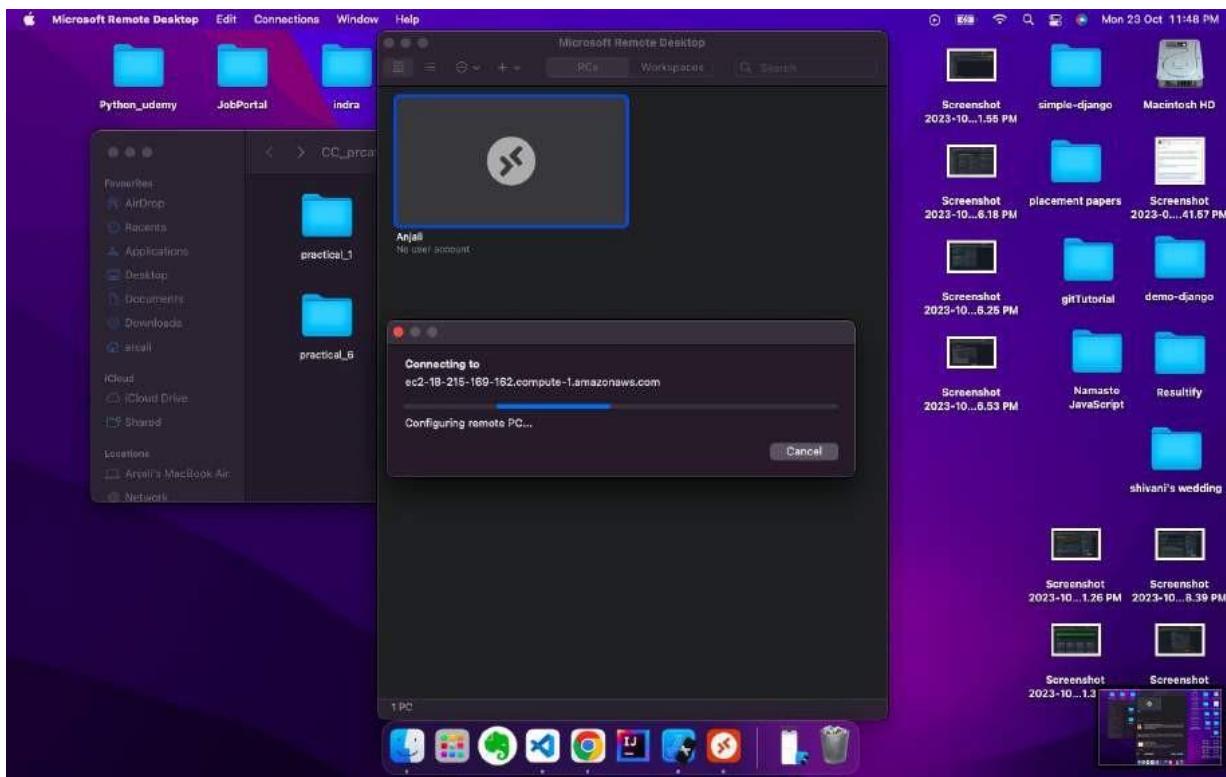
After configuring the connection, click "Connect" or "Connect" in your RDP client. You will be prompted to enter the administrator's username and password. If you haven't changed the password, you can retrieve it from the AWS Management Console.

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)** and a key pair associated with it: **Test3**. A section for "Private key" allows uploading a private key file, with **Test3.pem** selected. Below this is a "Private key contents - optional" field containing the RSA PRIVATE KEY content:

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpQIBAAKCAQEAvURPByopkz42lWxG5bdRq7tOodL+jmexsIDAH6H6LGDOYxw/  
GMYdie85YsWXgPMi4Rfcm7S2/-3V8t7GPyVnWB0+7CDQT+VtzAPN/Umrre5RKxw  
6d+vyeg7n0UBmiYLIC-nwbLgeEczvlpVUG5opC4Rt05y3jhFY8f9lVTWMfl+V  
aQmRIETRFNxSLkv77gho08WYgATjvbk50h4CVNZYez2PLEQta3EkXat9pOBWH  
XUJ4ap/gsz/LElOVF8Cda1Raw3nYFhpW9aAY+k6ebtm5f6lBtuUSTC7shkwG3//  
9HUb6zaphSJKeTyTknoCSWPFXHDvra55GLWiDAQAoIbAQCAwAUAjJsv0jSr  
53ZWBlvOff7Cqqurs/JDcRbw/dkncsyxL0W6LBZH8fjqcnTOMSebsb6NVMSfF
```

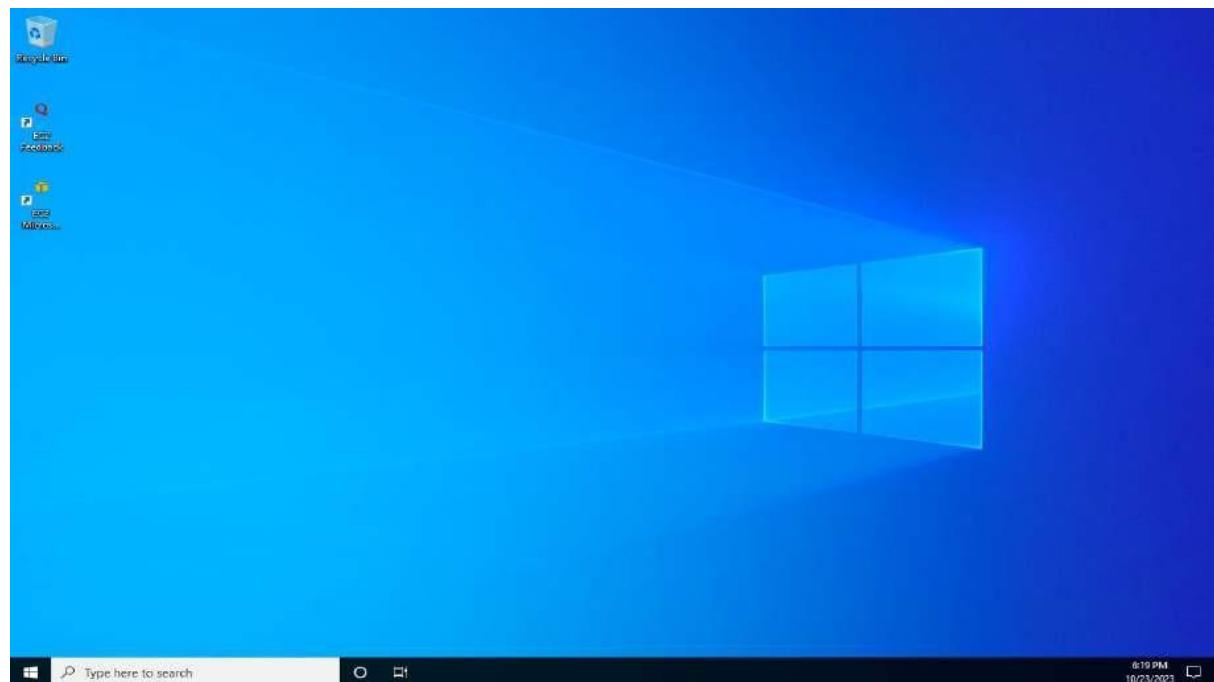
At the bottom are "Cancel" and "Decrypt password" buttons.





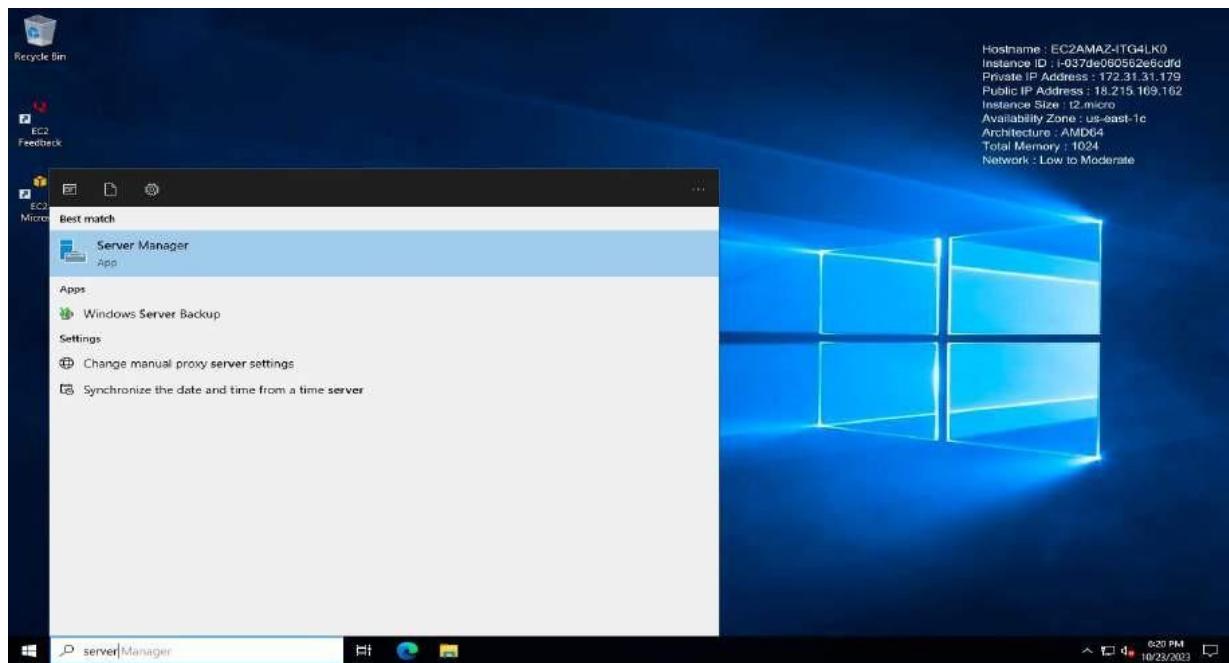
Step 8:

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:



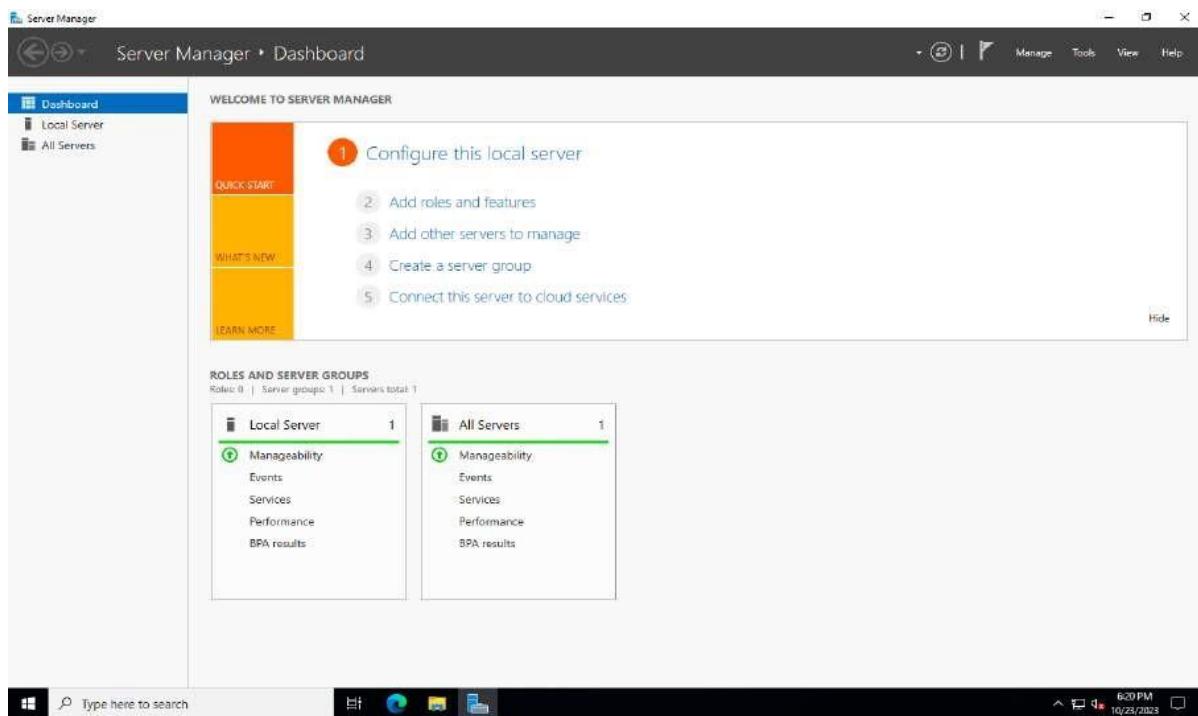
Step 9 :

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



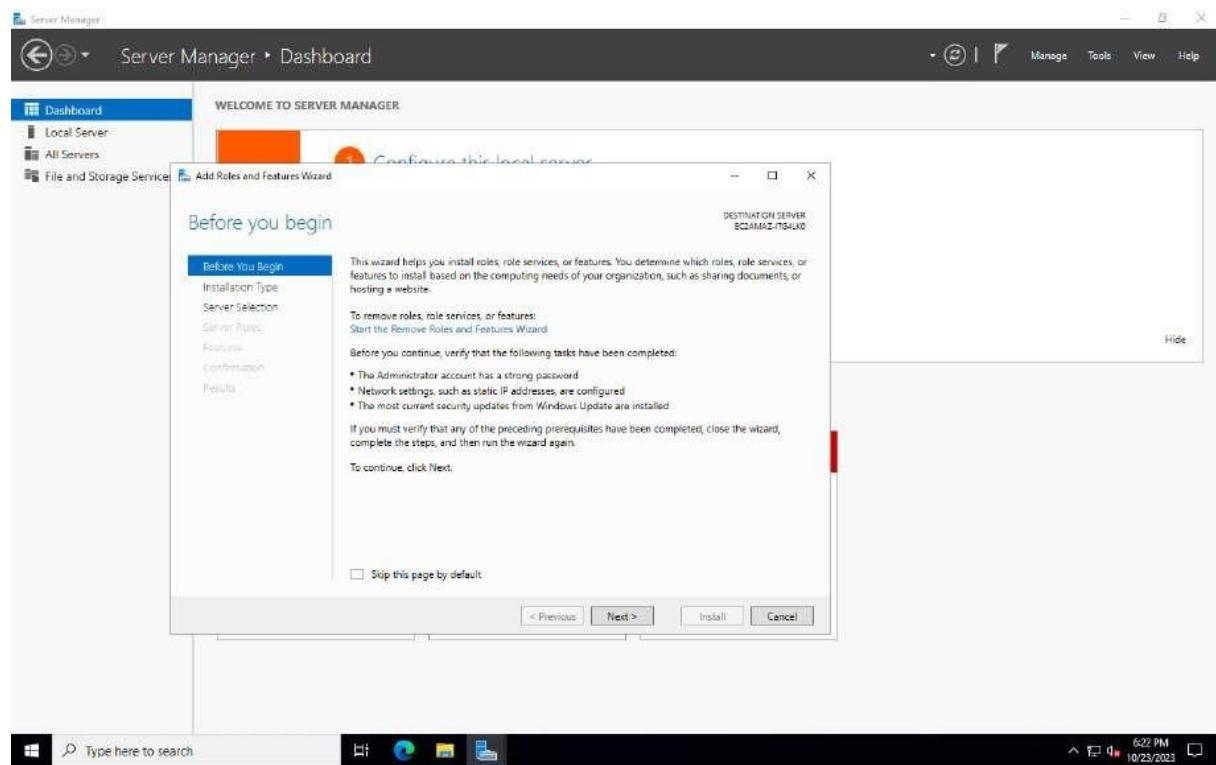
Step 10 :

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:

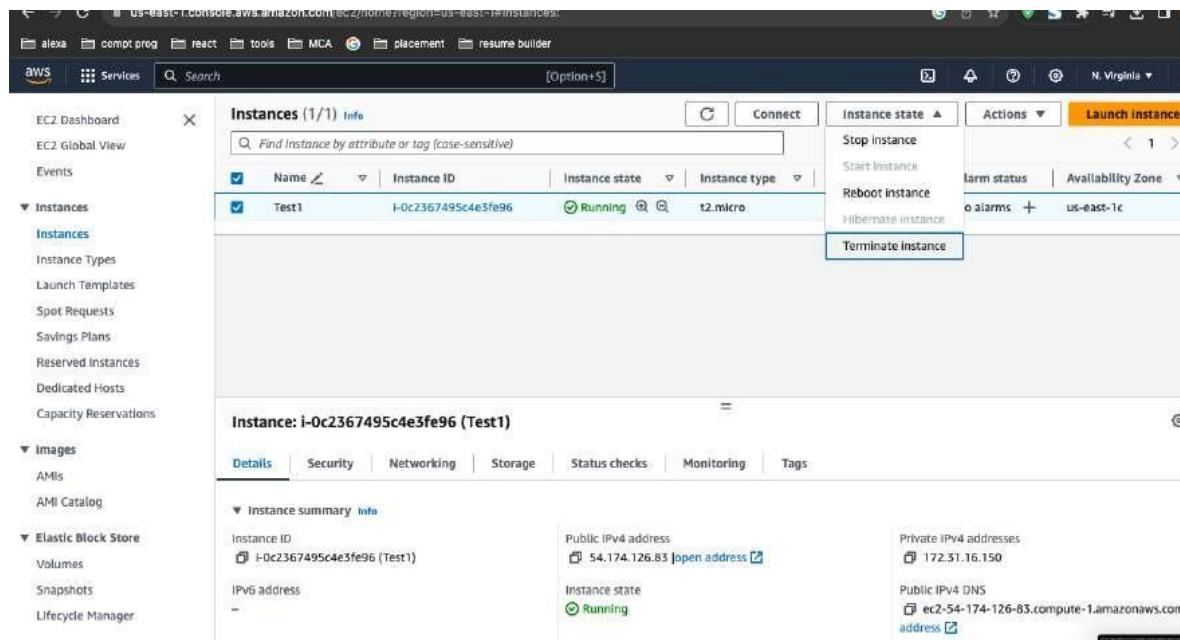
Terminate the launched Linux server instance from AWS EC2.

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:

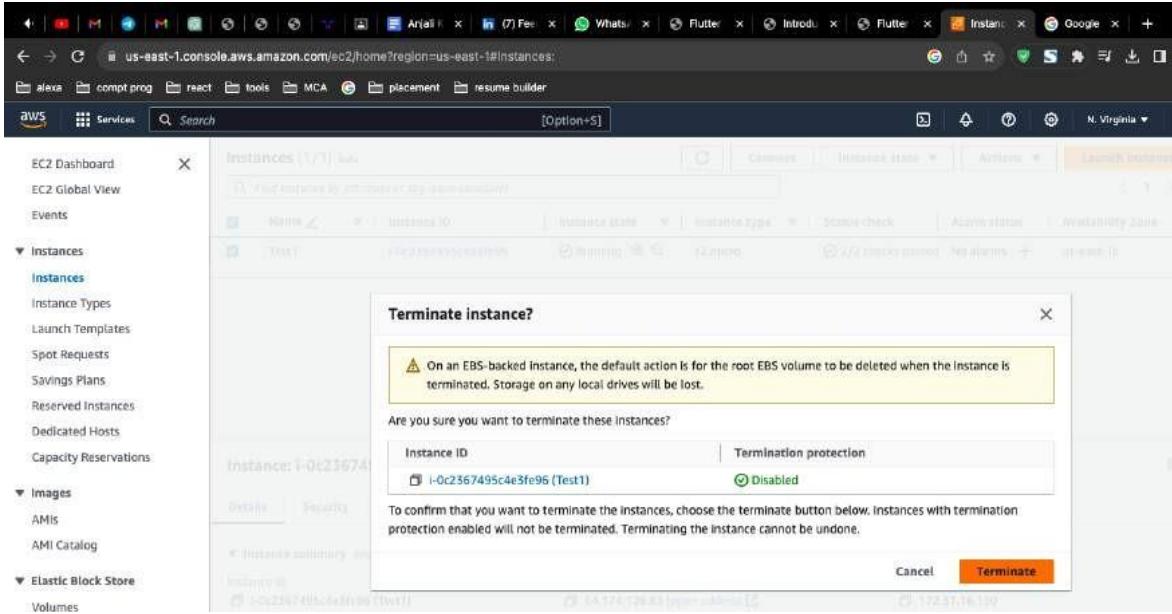
Click the checkbox next to the Linux server instance you want to terminate. It will become selected.



The screenshot shows the AWS EC2 Instances page. On the left, the navigation pane is open, showing 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 (selected), AMIs, and AMI Catalog. The main content area displays a table titled 'Instances (1/1)'. A single instance, 'Test1' (Instance ID: i-0c2367495c4e3fe96), is listed and selected, indicated by a checked checkbox in the 'Name' column. The instance status is 'Running'. To the right of the table is an 'Actions' dropdown menu with options: Stop Instance, Start Instance, Reboot Instance, Hibernate instance, and Terminate Instance (which is highlighted with a blue border). Below the table, a detailed view for 'Instance: i-0c2367495c4e3fe96 (Test1)' is shown, including 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)), Public IPv4 address (54.174.126.83), Private IPv4 addresses (172.31.16.150), Instance state (Running), Public IPv4 DNS (ec2-54-174-126-83.compute-1.amazonaws.com), and a note about the address being static.

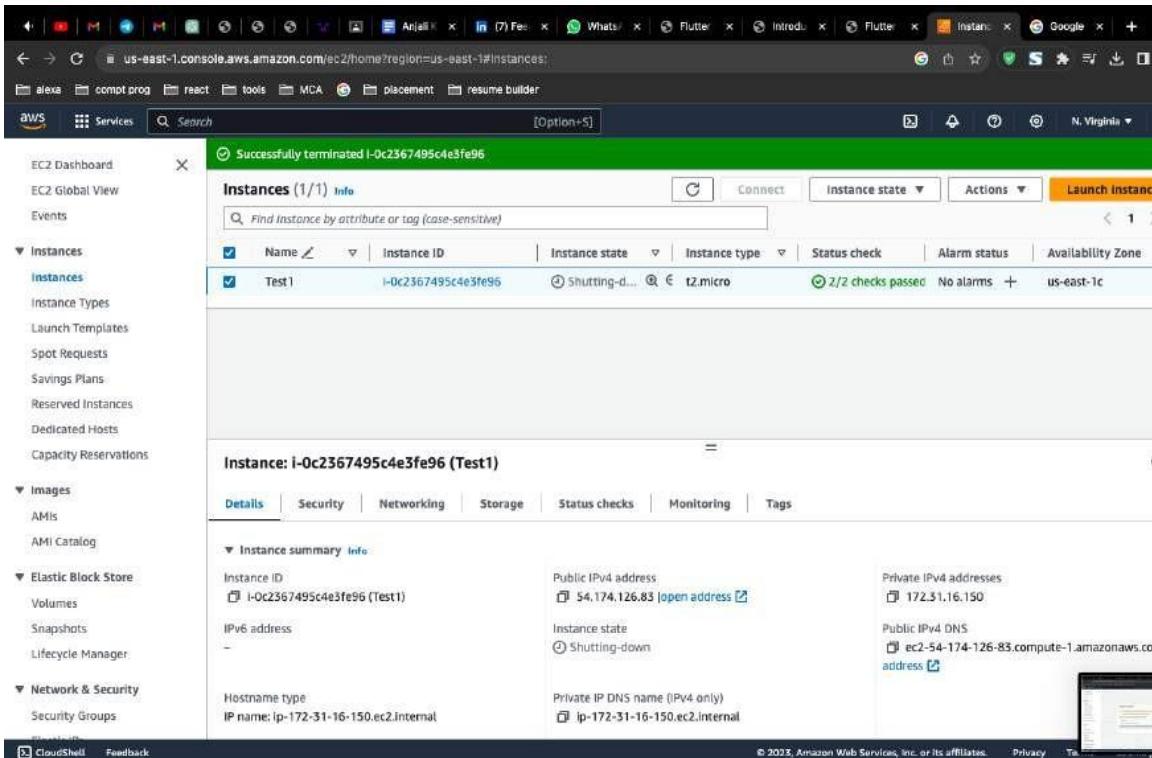
Step 3 :

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.

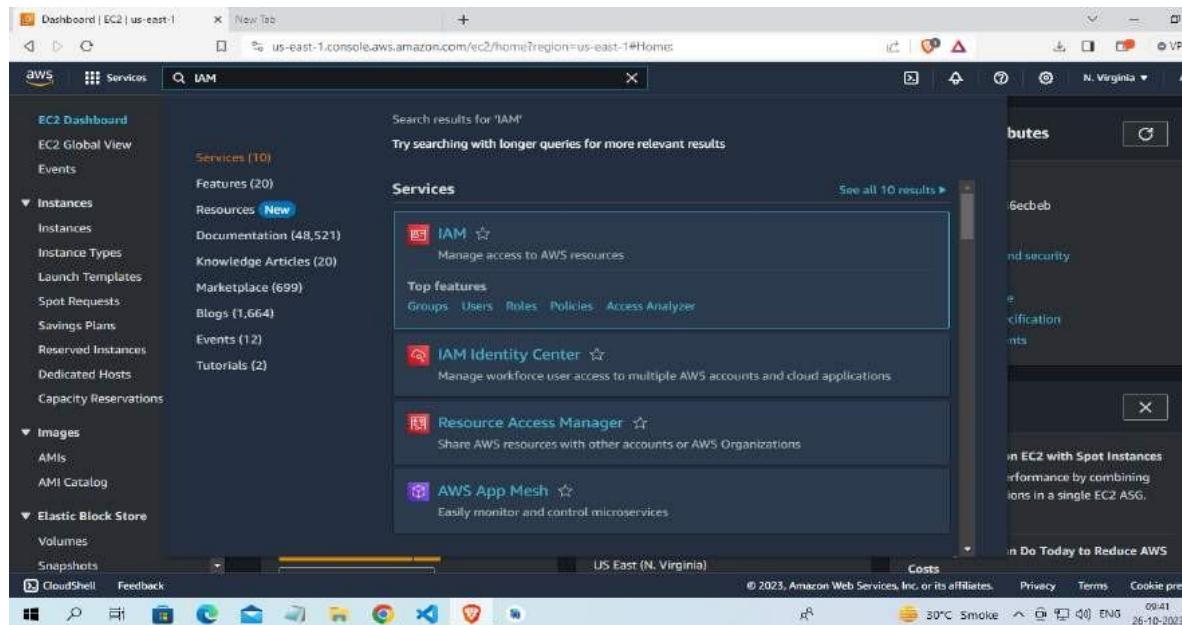


Practical 10:

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 IAM Dashboard. On the left, a navigation sidebar lists various IAM management options like Access management, Users, Roles, Policies, and Identity providers. The main area displays 'Security recommendations' with a red notification badge (1). It shows two items: 'Add MFA for root user' (warning icon) and 'Root user has no active access keys' (green checkmark 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 provides account details: 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 a Windows taskbar with various icons and system status.

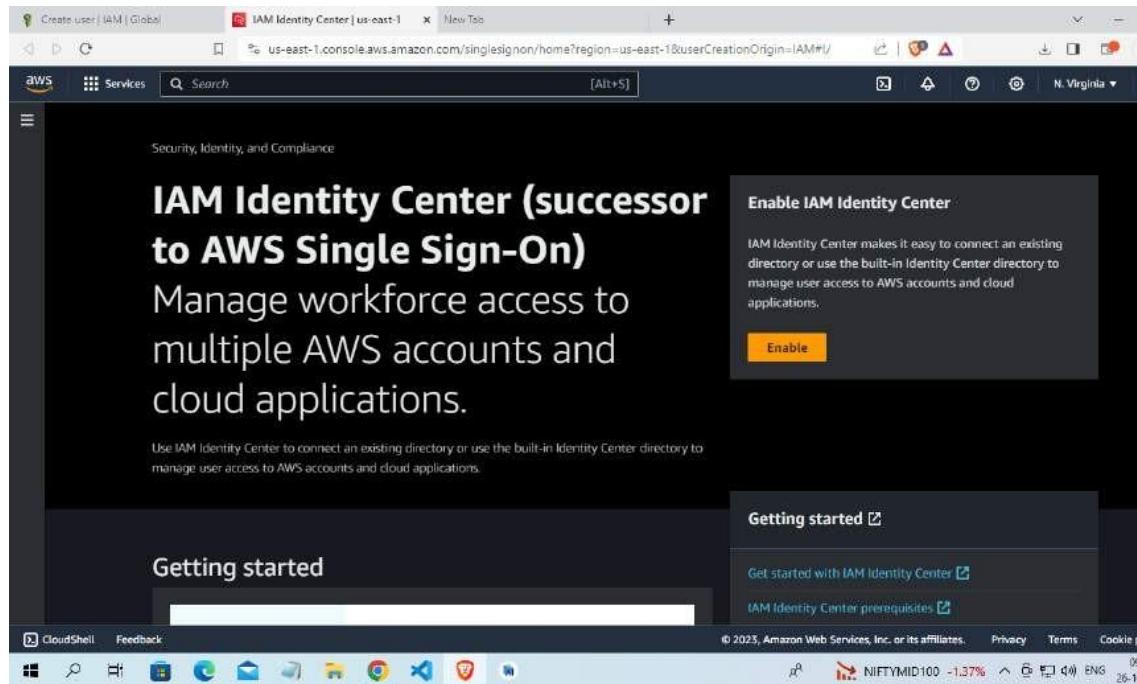
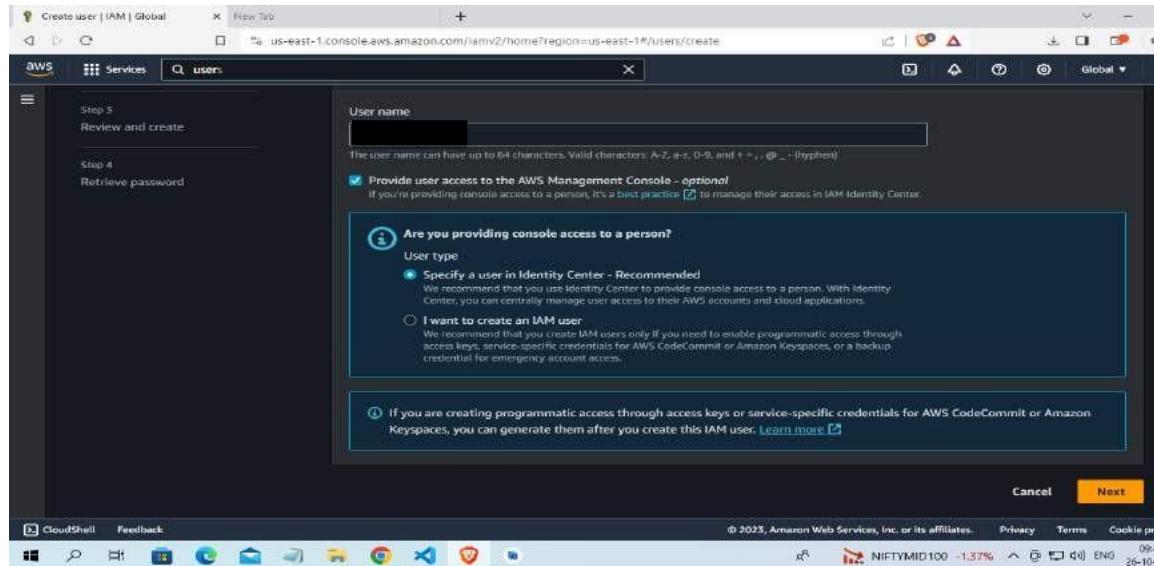
Step 3 :

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 is identical to the previous dashboard view. 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. The table body displays the message 'No resources to display'. The bottom of the screen shows a Windows taskbar with various icons and system status.

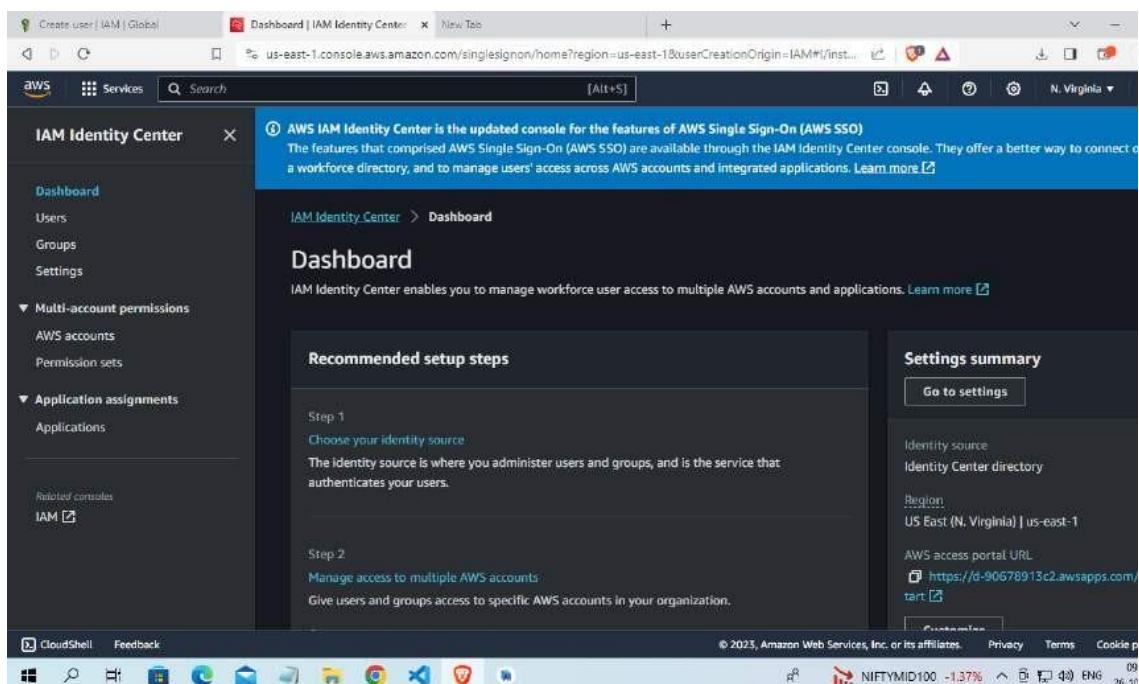
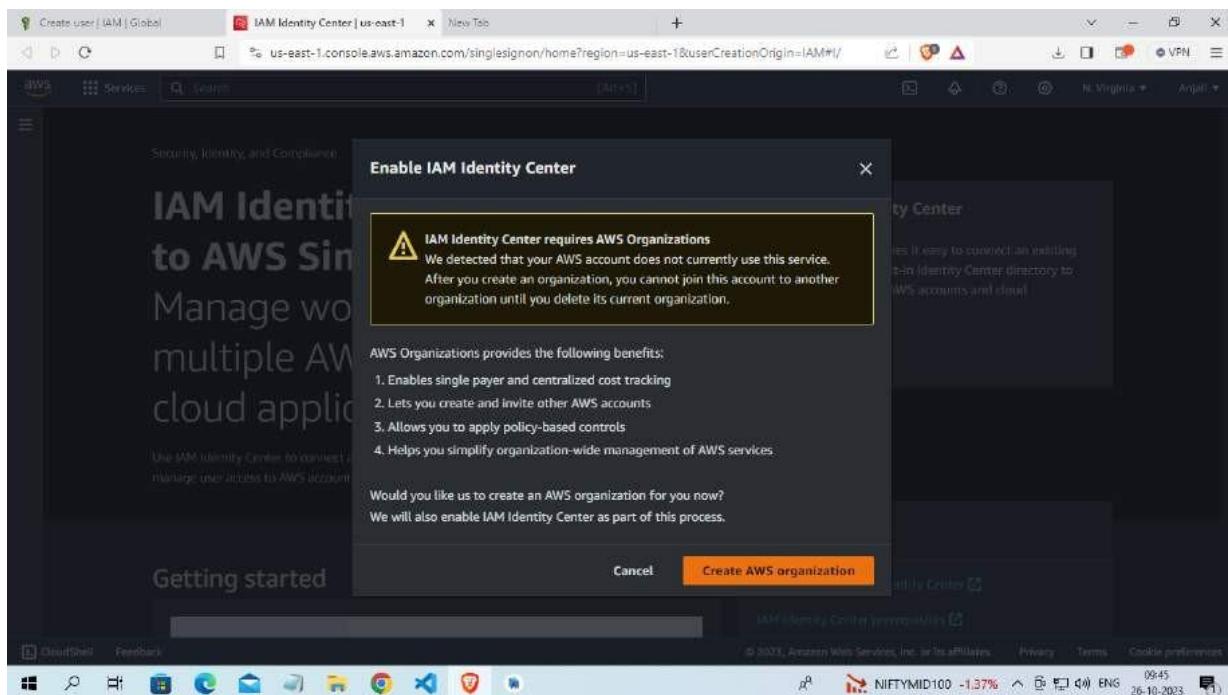
Step 4 :

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:

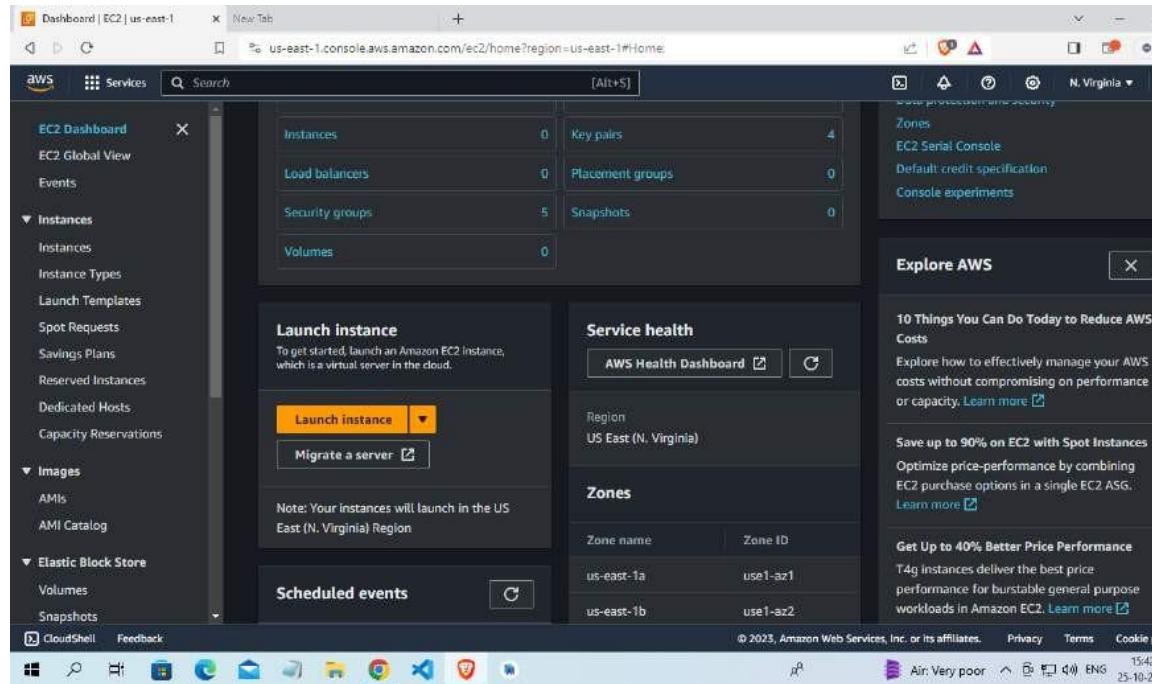
How to connect with a launched instance of Linux (putty & putty gen software).

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."

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Name and tags'. The 'Name' field contains 'Test5'. The 'Summary' panel on the right shows the configuration: 1 instance, Software Image (AMI) set to 'Amazon Linux 2023 AMI 2023.2.2...', Virtual server type (instance type) set to 't2.micro', and a message indicating a 'Free tier: In your first year'. A yellow 'Launch instance' button is visible.

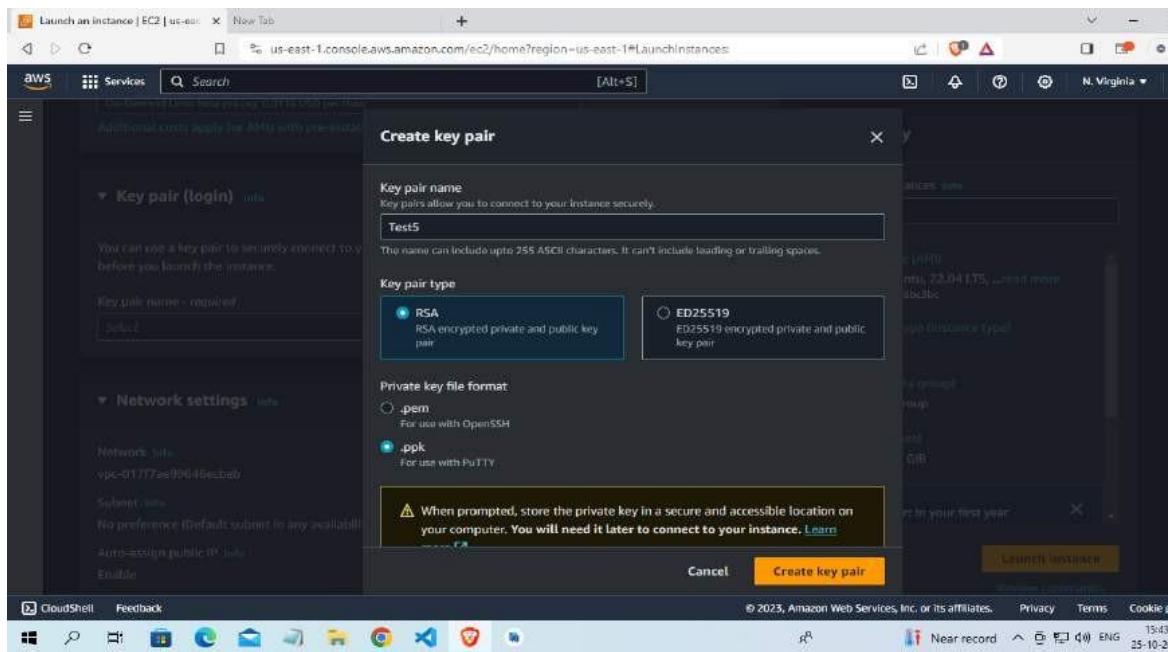
The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Application and OS Images (Amazon Machine Image)'. The 'Amazon Linux' option is selected. The 'Summary' panel on the right shows the configuration: 1 instance, Software Image (AMI) set to 'Canonical, Ubuntu, 22.04 LTS ...', Virtual server type (instance type) set to 't2.micro', and a message indicating a 'Free tier: In your first year'. A yellow 'Launch instance' button is visible.

Step 3:

Select Key Pair or Create a New Key Pair:

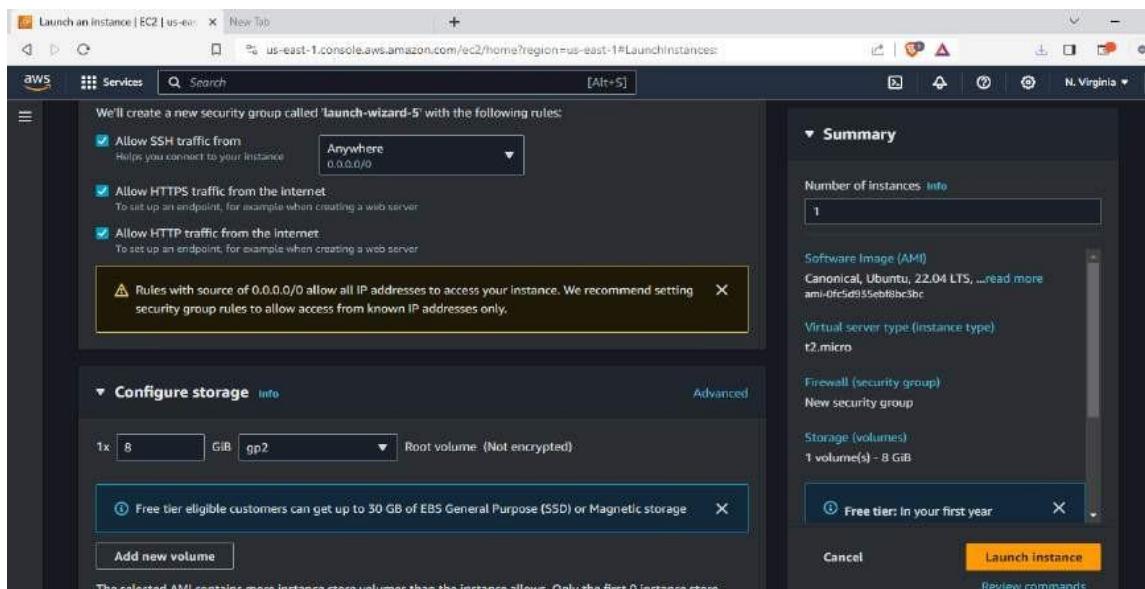
You'll need a key pair to securely connect to your instance. If you already have one, select it. If not, create a new key pair.

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



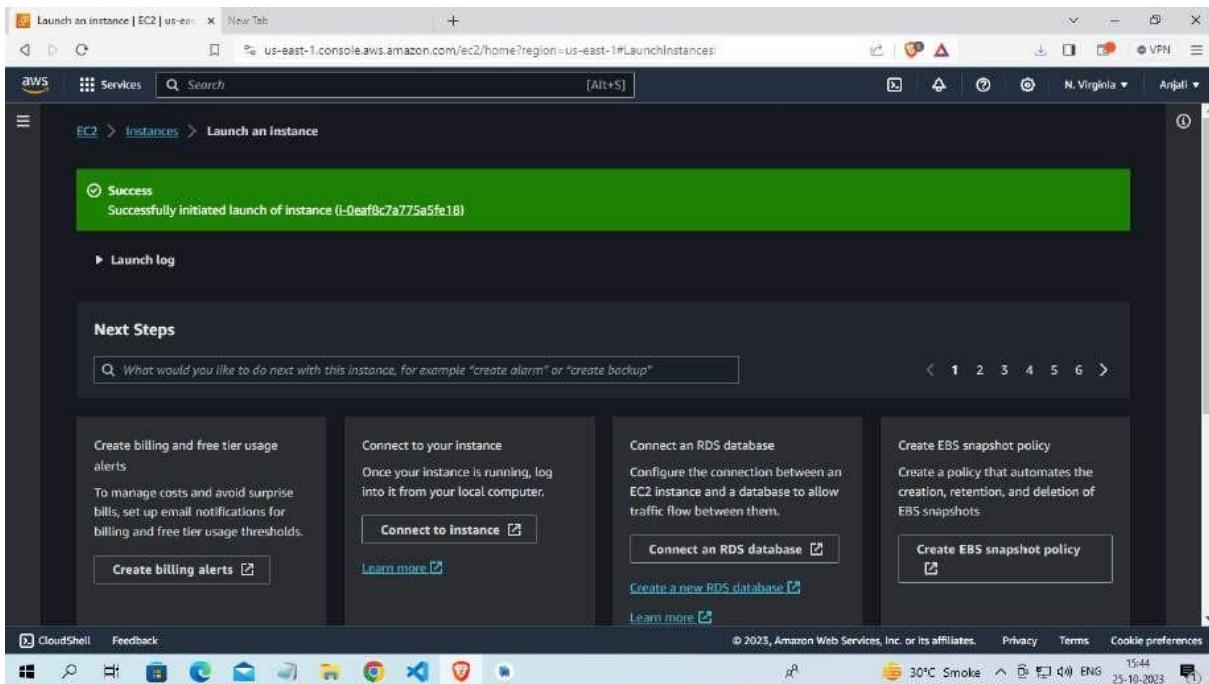
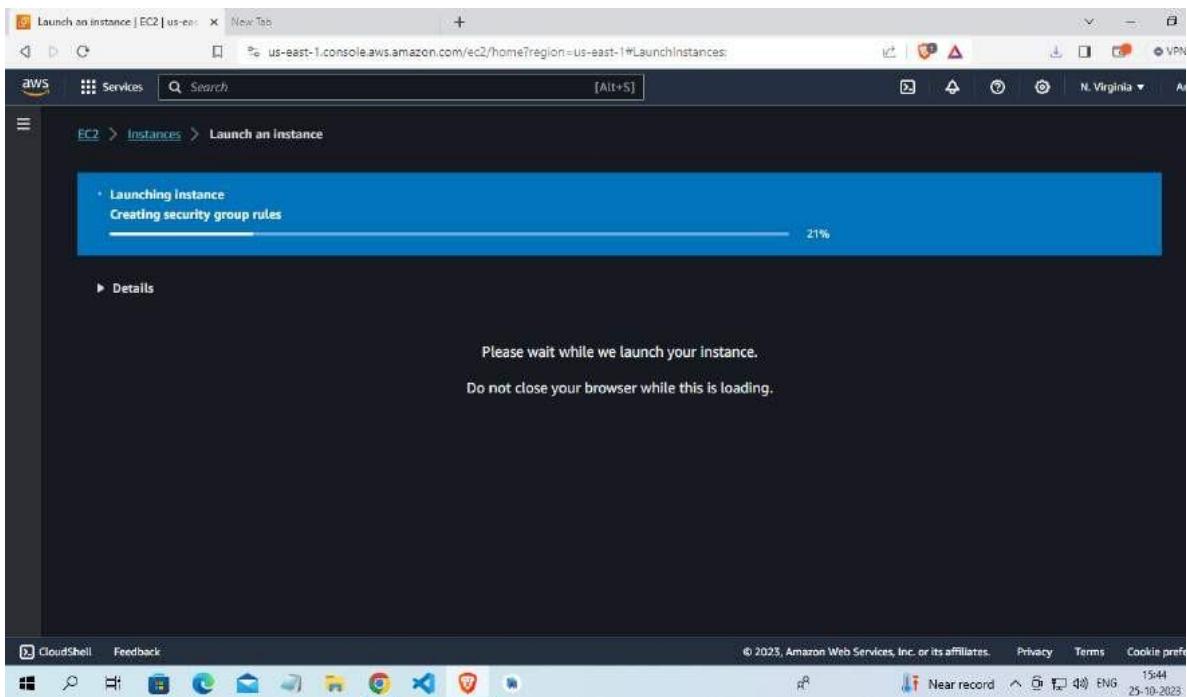
Step 4:

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.

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), and Elastic Block Store (Volumes, Snapshots). The main content area has a title 'Instances (1) Info' with a search bar. A table lists one instance: 'Test5' (Instance ID: i-0eaf8c7a775a5fe18, State: Running, Type: t2.micro, Status: 2/2 checks passed, Alarm status: No alarms, Region: us-east-1). Below the table is a section titled 'Select an instance' with a dropdown menu. The bottom right corner shows system information: 30°C, Smoke, 09:24, 26-10-2023.

The screenshot shows the AWS EC2 Instance details page for instance 'Test5'. The left sidebar is identical to the previous screenshot. The main content area is titled 'Instance summary for i-0eaf8c7a775a5fe18 (Test5) Info' (Updated less than a minute ago). It displays various instance details in a grid format:

Instance ID	i-0eaf8c7a775a5fe18 (Test5)	Public IPv4 address	Private IPv4 addresses
IPv6 address	-	107.22.137.25 [open address]	172.31.42.89
Hostname type	IP name: ip-172-31-42-89.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-42-89.ec2.internal	Public IP DNS ec2-107-22-137-25.compute-1.amazonaws.com [open address]
Answer private resource DNS name IPv4 (A)		Instance type t2.micro	Elastic IP addresses -
Auto-assigned IP address	107.22.137.25 [Public IP]	VPC ID vpc-017f7ae99646ecbeb [open address]	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. [Learn more]
IAM Role	-	Subnet ID subnet-02cb264838b23bd2 [open address]	Auto Scaling Group name -
IMDSv2			

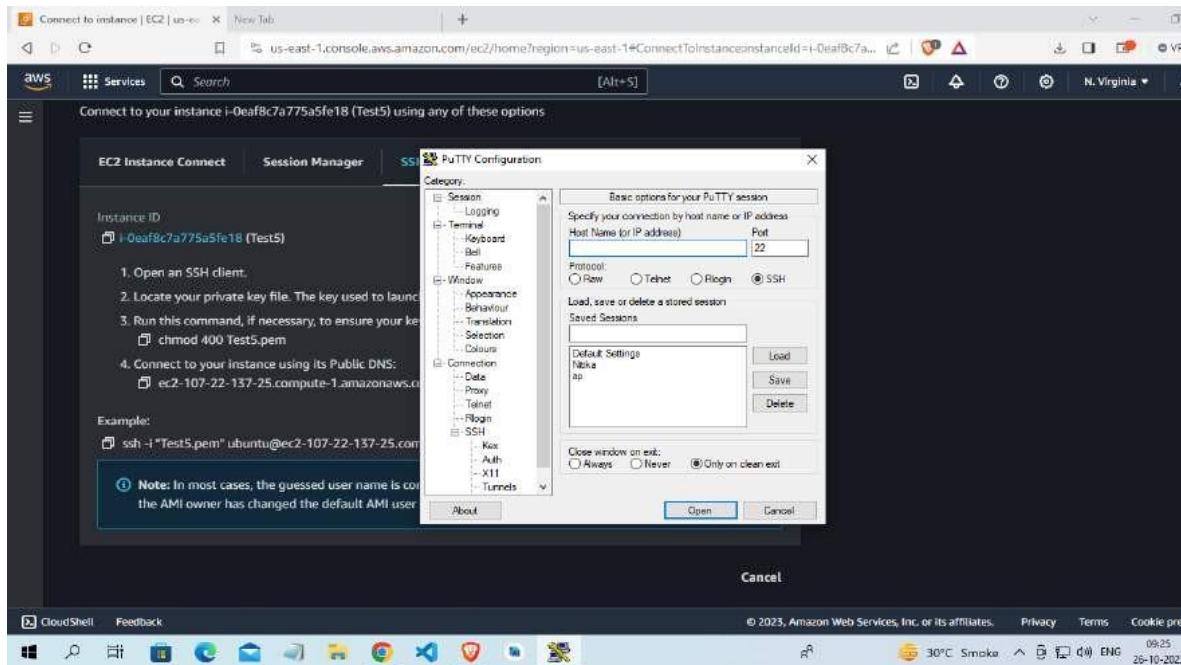
The bottom right corner shows system information: 30°C, Smoke, 09:24, 26-10-2023.

Step 7:

Download PuTTYgen .

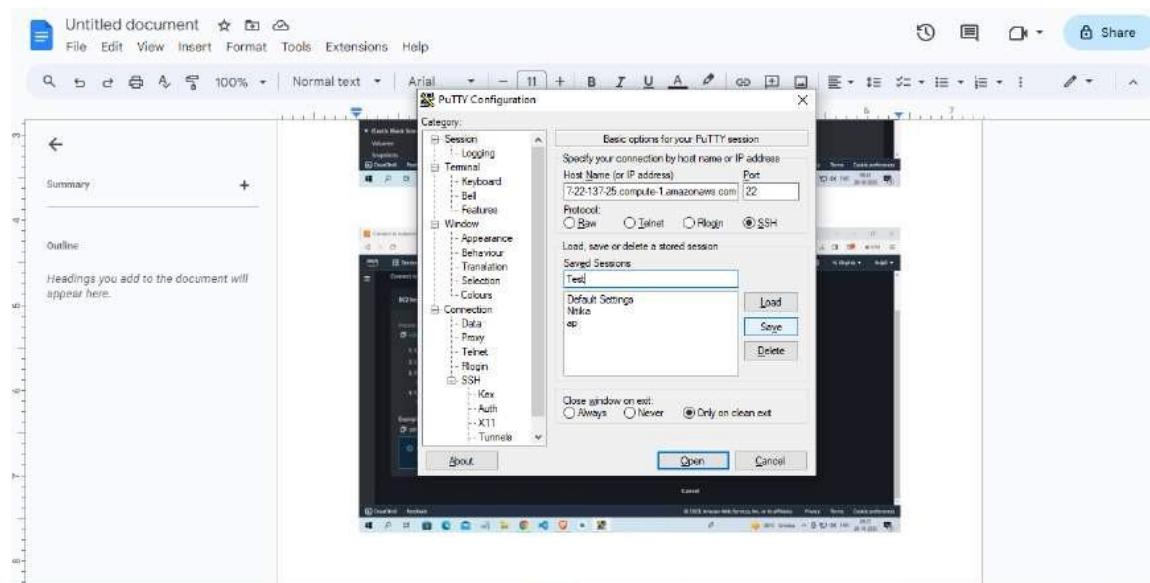
Step 8:

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



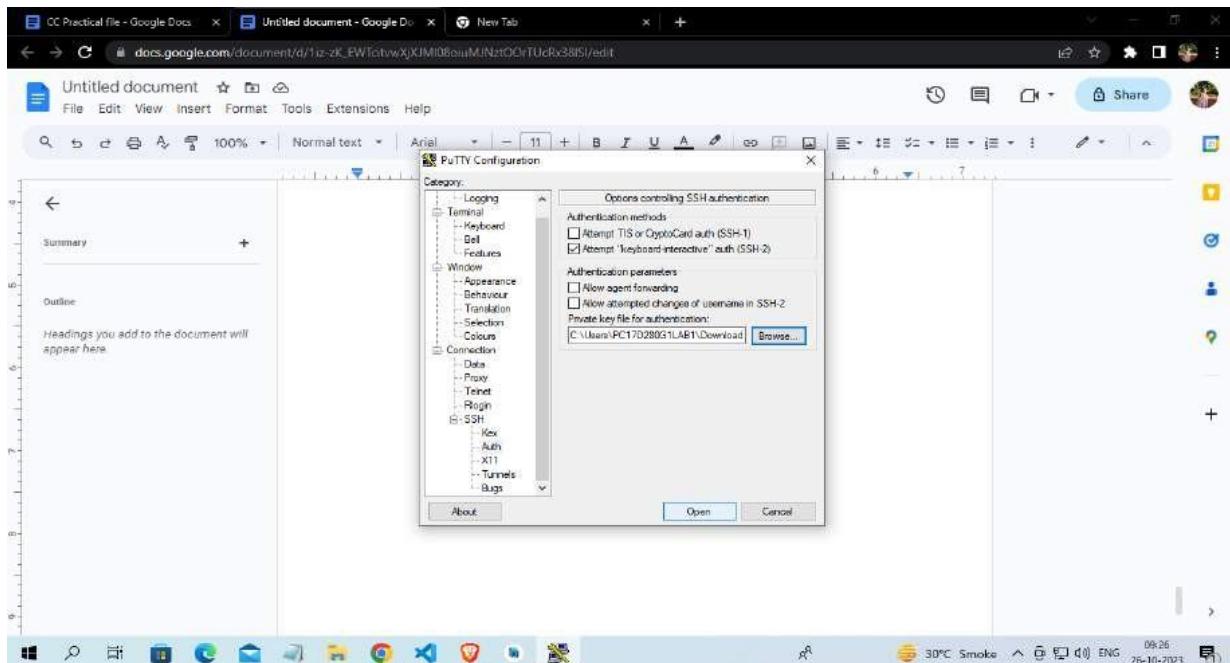
Step 9:

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



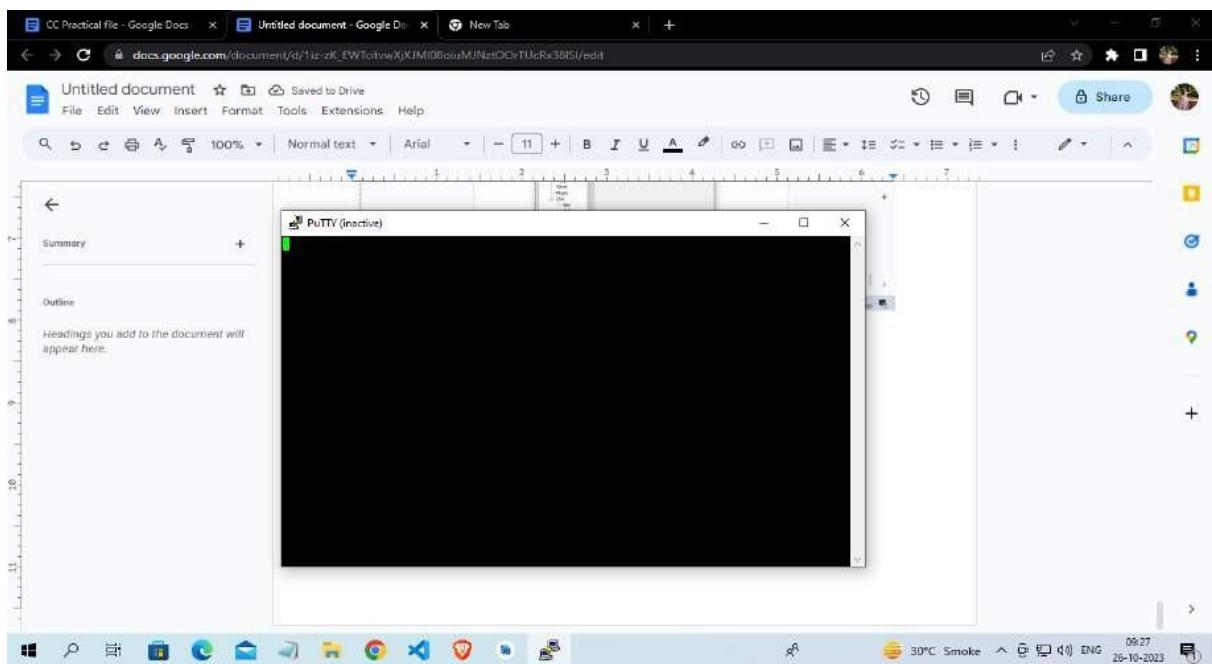
Step 10:

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

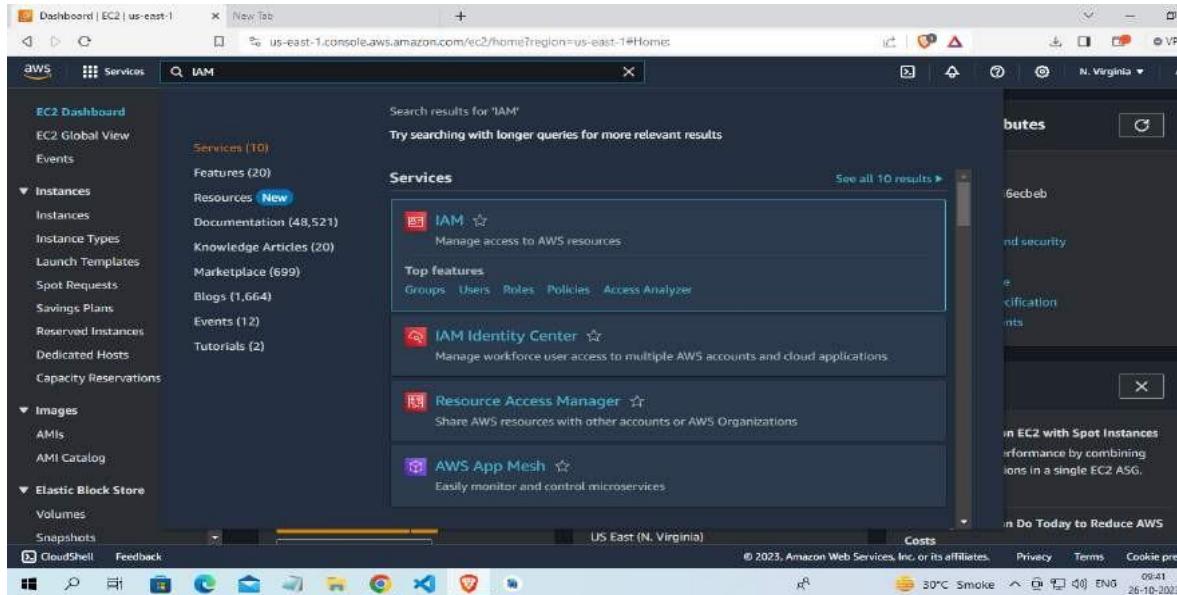


Practical 12:

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, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), and Access reports (Access analyzer, Archive rules, Analyzers, Settings). The main area displays 'Security recommendations' with a red notification badge (1). It lists 'Add MFA for root user' (warning icon) and 'Root user has no active access keys' (green checkmark 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 is identical to the previous dashboard view. The main content area shows a table titled 'Users (0) Info' with a note: 'An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.' The table has columns for User name, Path, Group, Last activity, MFA, and Password age. A search bar is at the top of the table. At the bottom of the table, it says 'No resources to display'. A prominent orange 'Create user' button is located at the top right of the main content area. 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 a Windows desktop environment.
 The top screenshot shows the 'Create user | IAM | Global' step 5 'Review and create' screen. It displays a user name 'Anjali Kumari' and a checked checkbox for 'Provide user access to the AWS Management Console - optional'. A callout box asks if the user wants to provide console access to a person or an IAM user, with the 'Specify a user in Identity Center - Recommended' option selected. A note below says, 'We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.' Another note for 'I want to create an IAM user' is present. A link to learn more about programmatic access is at the bottom.
 The middle screenshot shows the 'Create user | IAM | Global' step 4 'Retrieve password' screen, which is mostly blank.
 The bottom screenshot shows the 'IAM Identity Center (successor to AWS Single Sign-On)' landing page. It features a large heading, a sub-headline 'Manage workforce access to multiple AWS accounts and cloud applications.', and a callout box titled 'Enable IAM Identity Center' with 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.' An 'Enable' button is shown. Below this, a 'Getting started' section has a 'Get started with IAM Identity Center' link. The browser status bar at the bottom indicates the URL is 'us-east-1.console.aws.amazon.com/singlesignon/home?region=us-east-1&userCreationOrigin=IAM#/'.
 The browser interface includes standard windows like CloudShell, Feedback, and a taskbar with various icons.

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.

The screenshot shows two views of the AWS IAM Identity Center interface. The top view is a modal dialog titled 'Enable IAM Identity Center' with a warning message about AWS Organizations. It lists four benefits of using AWS Organizations: enabling single payer, centralized cost tracking, creating and inviting other AWS accounts, and applying policy-based controls. A button at the bottom right says 'Create AWS organization'. The background shows a dark-themed 'Getting started' screen with text about connecting IAM Identity Center to AWS accounts. The bottom view is the main 'Dashboard' page, which includes a sidebar with navigation links like 'Dashboard', 'Users', 'Groups', 'Settings', 'Multi-account permissions', 'AWS accounts', 'Permission sets', 'Application assignments', 'Applications', and 'Related consoles' (IAM). The main content area displays 'Recommended setup steps' (Step 1: Choose your identity source, Step 2: Manage access to multiple AWS accounts) and a 'Settings summary' section showing 'Identity source: Identity Center directory', 'Region: US East (N. Virginia) | us-east-1', and 'AWS access portal URL: https://d-90678913c2.awsapps.com/start'. The browser status bar at the bottom indicates the URL is 'us-east-1.console.aws.amazon.com/singlesignon/home?region=us-east-1&userCreationOrigin=IAM#/instance'.

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 includes links for Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Storage Lens, Dashboards, and AWS Organizations settings. A 'Feature spotlight' section is also present. The main content area displays search results for 's3' under 'Features' and 'Resources'. Under 'Features', there are sections for 'Imports from S3' (with a 'DynamoDB feature' link), 'Batch Operations' (with an 'S3 feature' link), 'Buckets' (with an 'S3 feature' link), and 'Access points' (with an 'S3 feature' link). Under 'Resources', there is a link to 'Introducing resource search'. On the right side of the main content area, there is a 'Create bucket' button. The bottom of the screen shows the Windows taskbar with various 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 one in the previous screenshot. The main content area is titled 'Amazon S3 > Buckets'. It features an 'Account snapshot' section with a 'View Storage Lens dashboard' button. Below this is a 'Buckets' table with a single row: 'No buckets'. A message states 'You don't have any buckets.' Below the table is a large 'Create bucket' button. 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 contains '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 after the bucket creation. A green banner at the top says 'Successfully created bucket "test0203"'. Below it, the 'Account snapshot' section displays the newly created bucket. The table lists 'test0203' with 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:

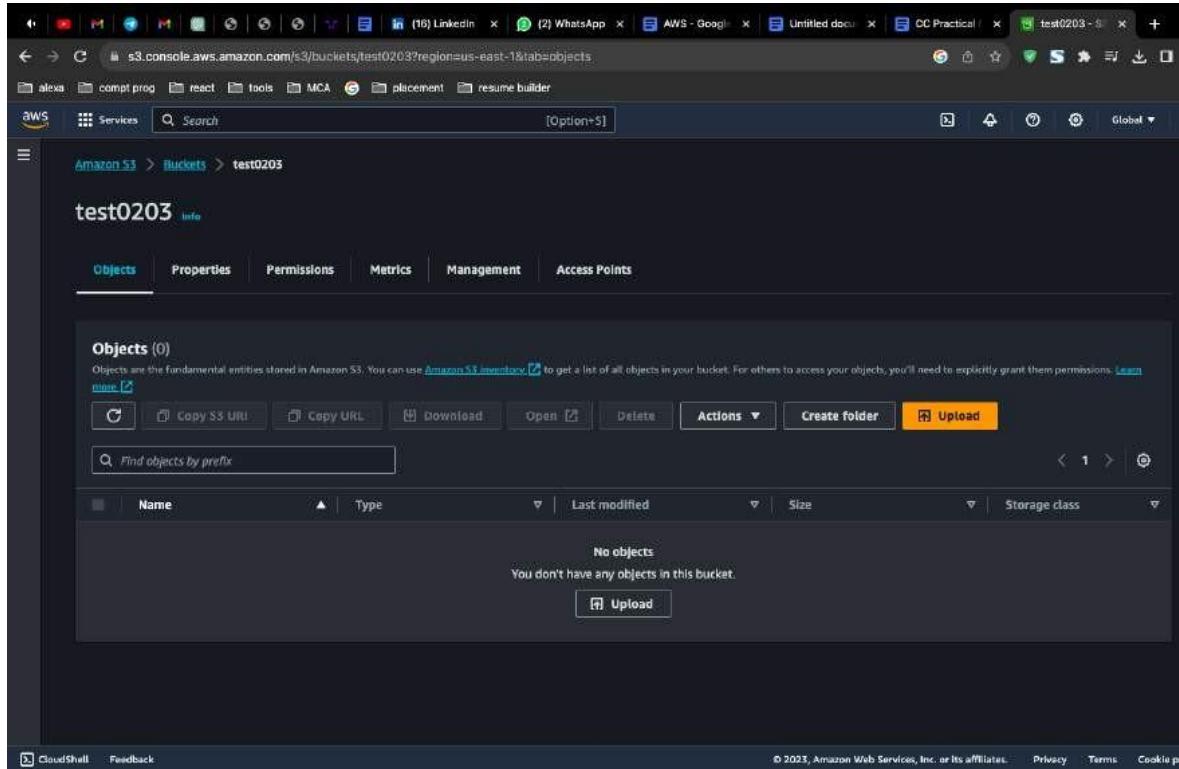
Upload an object on bucket created by using S3 AWS service

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 dashed box with the placeholder text "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". It includes sections for "Destination details" (Bucket settings that impact new objects stored in the specified destination) and "Permissions" (Grant public access and access to other AWS accounts). At the bottom right are "Cancel" and "Upload" buttons.

This screenshot is identical to the one above, showing the AWS S3 Upload interface. It displays the same file ("IMG_1008.JPG") in the "Files and folders" table, the same destination ("s3://test0203"), and the same sections for "Destination details" and "Permissions". The "Upload" button is visible at the bottom right.

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'. The transfer rate is listed as 'Transfer rate: 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 the summary is a 'Files and folders' table with one item: 'Files and folders (1 Total, 10.1 MB)'. The table includes columns for Name, Folder, Type, Size, Status, and Error. The status for the single entry is 'Succeeded'. The bottom of the screen shows standard browser navigation and search bars.

The screenshot shows the AWS S3 console interface after the upload has completed successfully. A green header bar displays the message 'Upload succeeded' with a link to 'View details below.'. Below this, the 'Upload: status' section shows the same summary table as the previous screenshot, but with a green checkmark icon next to 'Succeeded' and a green '100.00%' completion percentage. The 'Files and folders' table now lists the uploaded file: 'IMG_1008.JPG' (Type: Image/jpeg, Size: 10.1 MB, Status: Succeeded). The bottom of the screen shows standard browser navigation and search bars.

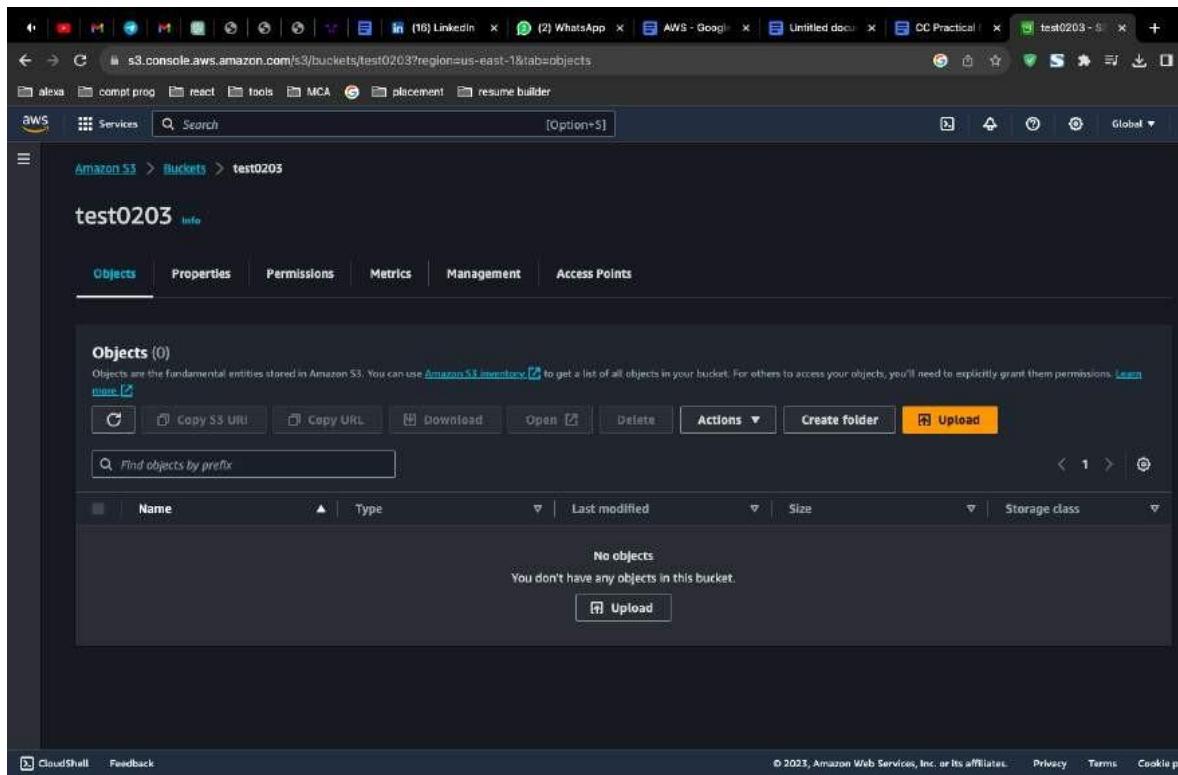
Practical 15:

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. At the top, the URL is `s3.console.aws.amazon.com/s3/upload/test0203?region=us-east-1`. The main area is titled 'Upload' with a sub-section 'Info'. A note says: '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](#)'.

A large box allows dragging and dropping files or choosing 'Add files' or 'Add folder'. Below this is a table titled 'Files and folders (1 Total, 10.1 MB)'. It 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`'. It includes sections for 'Destination details' (Bucket settings that impact new objects stored in the specified destination) and 'Permissions' (Grant public access and access to other AWS accounts).

At the bottom right are 'Cancel' and 'Upload' buttons. The status bar at the bottom indicates 'CloudShell' and 'Feedback'.

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(99.61%)". Below this, the "Upload: status" section displays a summary table. The table has three columns: "Destination" (s3://test0203), "Succeeded" (0 files, 144.0 KB (1.39%)), and "Failed" (0 files, 0 B (0%)). Below the summary, 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" (Type: Image/jpeg, Size: 10.1 MB, Status: Succeeded). The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2023, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie.

This screenshot shows the same AWS S3 console interface after the upload has completed successfully. A prominent green banner at the top left says "Upload succeeded" with a link to "View details below.". The "Upload: status" section now shows a summary table where all metrics are at 100% completion: "Succeeded" (1 file, 10.1 MB (100.00%)) and "Failed" (0 files, 0 B (0%)). The "Files and folders" table now shows the uploaded file "IMG_1008.JPG" with a green "Succeeded" status indicator. The rest of the interface remains consistent with the first screenshot, including the navigation bar, tabs, and footer.

Step 5 :

Go to permissions and click on bucket owner enforced

The screenshot shows the AWS S3 Management Console with the URL <https://s3.console.aws.amazon.com/s3/object/bcii?region=eu-north-1&prefix=4337dda66120f7c3bb5e02c005ff3d92.png&tab=permissions>. The left sidebar is expanded, showing 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'IAM Access Analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards' and 'AWS Organizations settings'), and 'Feature spotlight'. The main content area is titled '4337dda66120f7c3bb5e02c005ff3d92.png' and has tabs for 'Properties', 'Permissions' (which is selected), and 'Versions'. A callout box highlights the 'Access control list (ACL)' section, which contains the message: 'This bucket has the bucket owner enforced setting applied for Object Ownership'. Below this, it says 'When bucket owner enforced is applied, use bucket policies to control access.' A table lists grants: 'Object owner (your AWS account)' with Canonical ID and Group, and 'Everyone (public access)' with Group. The table has columns for 'Grantee', 'Object', and 'Object ACL'. The 'Object ACL' column shows 'Read' for the object owner and 'Read, Write' for everyone.

Step 6:

click on ACL2 enabled

The screenshot shows the 'Edit Object Ownership' page in the AWS S3 Management Console. The URL is <https://s3.console.aws.amazon.com/s3/bucket/bcii/property/oo/edit?region=eu-north-1>. The left sidebar is similar to the previous screenshot. The main content area is titled 'Edit Object Ownership' and has a sub-section 'Object Ownership'. It explains that 'Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.' Two options are shown: 'ACLs disabled (recommended)' (selected) and 'ACLs enabled'. A note below says 'We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.' A warning at the bottom states 'Enabling ACLs turns off the bucket owner enforced setting for Object Ownership' and 'Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.'

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"/> Write	
Canonical ID: 5103057552782131d37 3ad834ed5b8db8bd2b0a4594c dbe1841396e7fca37f		
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	<input checked="" type="checkbox"/> 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"/> 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]# 
```

i-0e9ed085ef8c5e863 (bciiit)
PublicIPs: 13.51.207.215 PrivateIPs: 172.31.36.199

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:

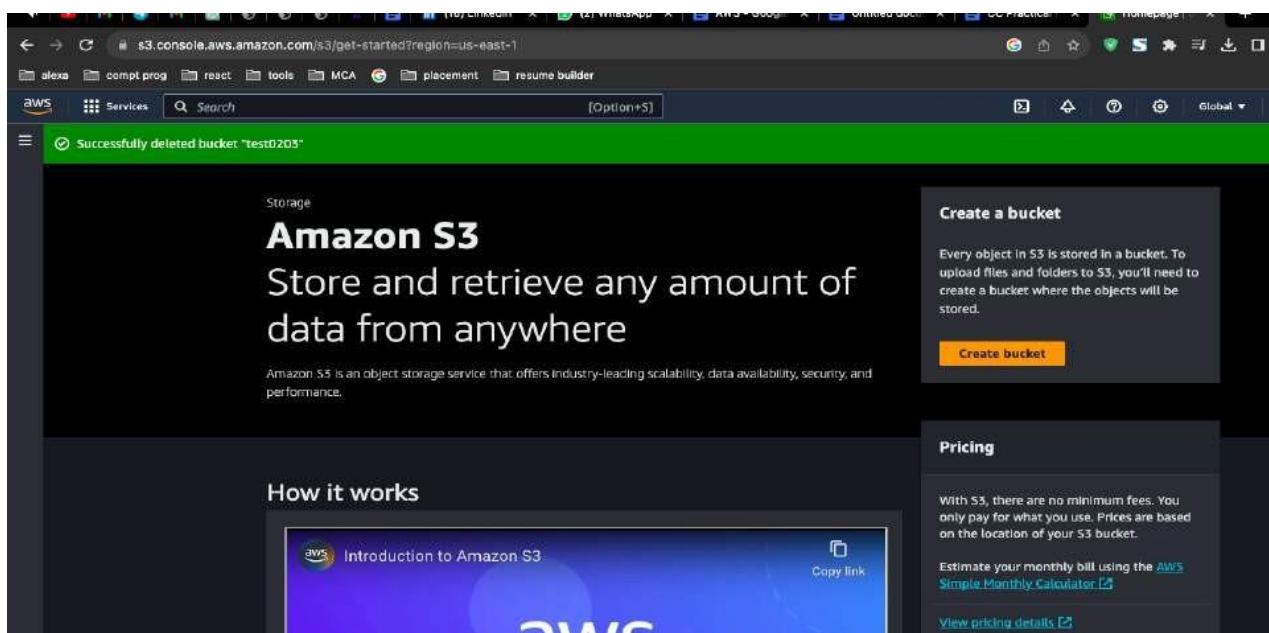
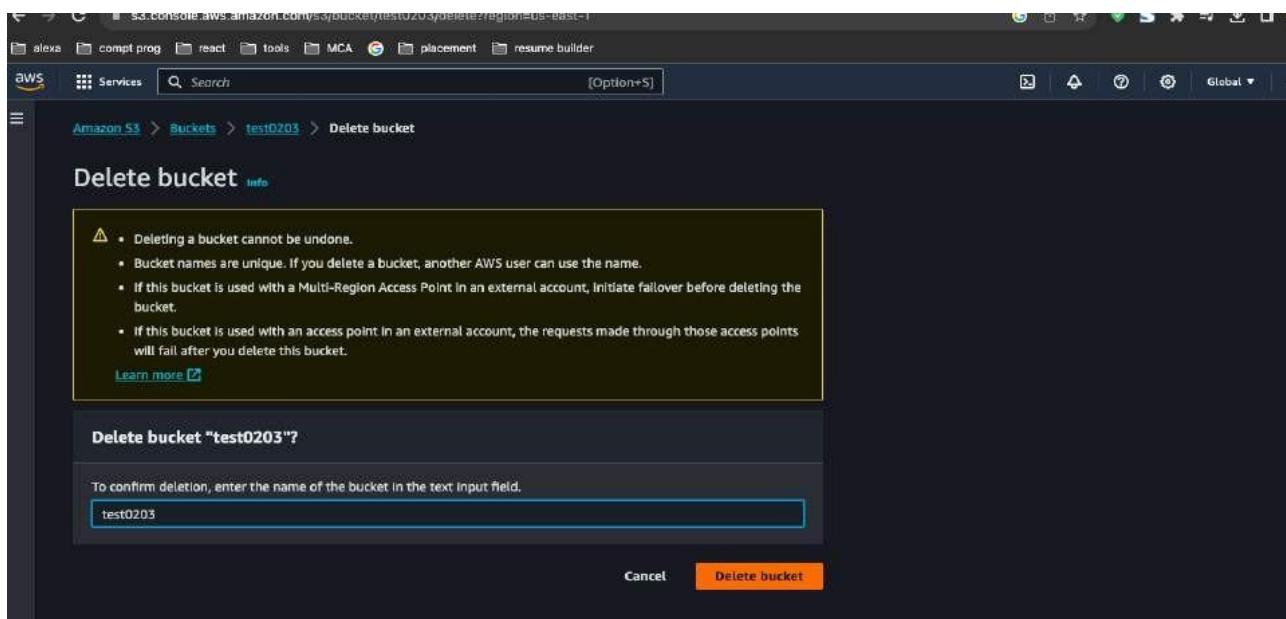
In the S3 dashboard, click on the name of the bucket from which you want to delete objects.

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®ion=us-east-1. The main content area has a title 'Amazon S3 > Buckets'. Underneath, there's an 'Account snapshot' section with a 'View Storage Lens dashboard' button. The main table lists one bucket: 'test0203'. The table has columns for Name, AWS Region, Access, and Creation date. The 'test0203' row shows 'US East (N. Virginia) us-east-1', 'Bucket and objects not public', and 'October 28, 2023, 21:45:13 (UTC-05:30)'. Action buttons for the row include 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. At the bottom of the page, there are links for CloudShell, Feedback, and a copyright notice: © 2023, Amazon Web Services, Inc. or its affiliates.

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 17:

Transfer the object file from S3 service to EC2 launched Linux server install GCC and wget commands in this regard on terminal

Objective : Transfer an object file from S3 to a Linux server running on EC2. Install GCC and wget on the server using terminal commands in AWS.

- 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.

Step 1: Connect to your EC2 instance via SSH.

Open a terminal application.

Type the ssh command followed by the username and IP address of your EC2 instance. For example:

```
ssh username@ec2-public-ip-address
```

Enter your password when prompted.

Step 2: Install wget:

Type the following command to update the package list:

Ubuntu/Debian:

```
sudo apt install wget
```

Step 3: Get the S3 object URL:

1. In the AWS Management Console, navigate to the S3 bucket containing the object file.
2. Right-click the object and select "Get Object URL".
3. Copy the URL to your clipboard.

Step 4: Transfer the object file to EC2 using wget:

1. In the SSH terminal window, type the following command, replacing URL_OF_S3_OBJECT with the URL you copied:

```
wget URL OF S3 OBJECT
```

2. Press Enter. The file will be downloaded to your current directory on the EC2 instance.

Step 5: Verify the file transfer:

1. Type the following command to list the files in your current directory:

```
ls -l
```

2. Make sure the object file is listed.

Step 6: Install GCC:

1. Update the package list again:

2. Install GCC using the appropriate command for your Linux distribution:

Ubuntu/Debian:

```
sudo apt install gcc
```

Step 7: Verify GCC installation:

1. Type the following command to check the installed GCC version:

```
gcc --version
```

2. The command should display the installed GCC version.

Screenshots:

1. Object URL in S3 Console:

The screenshot shows the AWS S3 console interface for a bucket named "havecamerawilltravel.developer". The "Properties" tab is highlighted with a red arrow. Below the tabs, there is a search bar with placeholder text "Type a prefix and press Enter or click here to clear". Underneath are buttons for "Upload", "+ Create folder", "Download", and "Actions". A list of objects is displayed with columns for "Name" and "Last modified". The "test.txt" file is selected, indicated by a checked checkbox and a blue highlight.

Name	Last modified
test.txt	Dec 4, 2019 12:07:29 PM GMT-06:00
test2.txt	Dec 4, 2019 12:16:07 PM GMT-06:00
test3.txt	Dec 4, 2019 12:51:40 PM GMT-06:00

2. wget command in terminal:

```
adam@adams-MacBook-Pro:~$ wget https://en.wikipedia.org/wiki/Wget
--2014-02-09 07:38:43--  https://en.wikipedia.org/w/index.php?title=Wget
Resolving en.wikipedia.org (en.wikipedia.org)... 20
80.154.224.26
Connecting to en.wikipedia.org (en.wikipedia.org)|20
80.154.224.26|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'Wget'

[=>          ] 91,646      583K/s   in 0.2s

2014-02-09 07:38:43 (583 KB/s) - 'Wget' saved [91646]

adam@adams-MacBook-Pro:~$
```

3. File listing showing downloaded object:

17a_Parent_Etc6566931	
5/23/2011 7:17 AM	43 9_15-3.gif
5/23/2011 7:17 AM	49 241873FC07A1
5/23/2011 7:17 AM	49517 ABS.css
5/23/2011 7:17 AM	38756 54head.jpg
5/23/2011 7:17 AM	508 Administram.gif
5/23/2011 7:17 AM	3127 Administram.jpg
5/23/2018 11:09 AM	3381 Administram.jpg
5/23/2018 11:09 AM	3397 arrow-left-dark.jpg
5/23/2018 11:09 AM	3321 arrow-left.png
5/23/2018 11:09 AM	3267 arrow-right-dark.jpg
5/23/2018 11:09 AM	347 arrow-right-light.jpg
5/23/2018 11:09 AM	855 calendardata2009.xls
5/23/2018 11:09 AM	847 calendardata2009.xls1.jpg
5/23/2018 11:09 AM	2212 calendarwala.gif
5/23/2011 7:17 AM	691 calendarwala.xls
5/23/2011 7:17 AM	1826 calendarwala.xls
5/23/2018 11:09 AM	21982 calendar.xls
7/3/2011 7:17 AM	3855 definefeatures.gif
7/3/2011 7:17 AM	2653 definisa.gif
7/3/2011 7:17 AM	3982 describa.gif
7/3/2011 7:17 AM	12394 describa.jpg
5/23/2018 11:09 AM	50795 doc_header.xls
7/3/2011 7:17 AM	24362 dochasser120x120.jpg
7/3/2011 7:17 AM	23305 Dochasser174x131.jpg
7/3/2011 7:17 AM	24362 dochasser174x131.jpg
7/3/2011 7:17 AM	22864 Dochasser_413-122.jpg

4. GCC installation command:

```
ii 210      3.0+1ldl01 snf04      archiver for zip file
ii 2libigsnf4 432.2.11-df5 snf04      compression library
ii 2libigsnf4-1.1.1-1-1-patched       $ sudo apt install gcc
[sudo] password for muralkrishna:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package gcc is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'gcc' has no installation candidate
muralkrishna@uralkrishna-Inspiron-5570:~/programs$ cd programs
muralkrishna@uralkrishna-Inspiron-5570:~/programs$ gcc welcome
Command 'gcc' not found, but can be installed with:

sudo apt install gcc

muralkrishna@uralkrishna-Inspiron-5570:~/programs$ sudo apt in
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package gcc is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
```

5. GCC version verification:

```
F:\>type wikipedija.c
#include <stdio.h>

int main (void) {
    printf("Pozdravljenia Wikipedija!\n");
}

F:\>gcc --version
gcc (GCC) 3.4.2 (mingw-special)
Copyright (C) 2008 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  The
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR

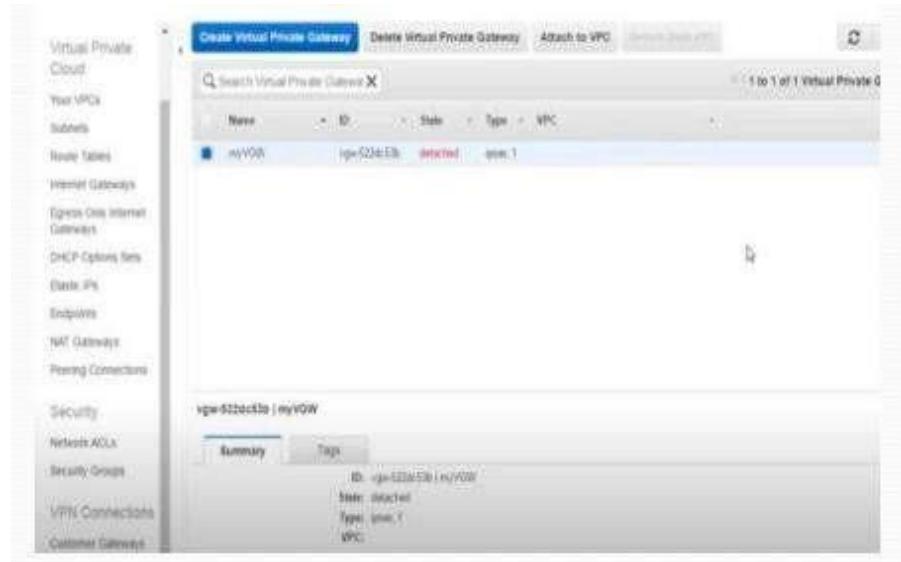
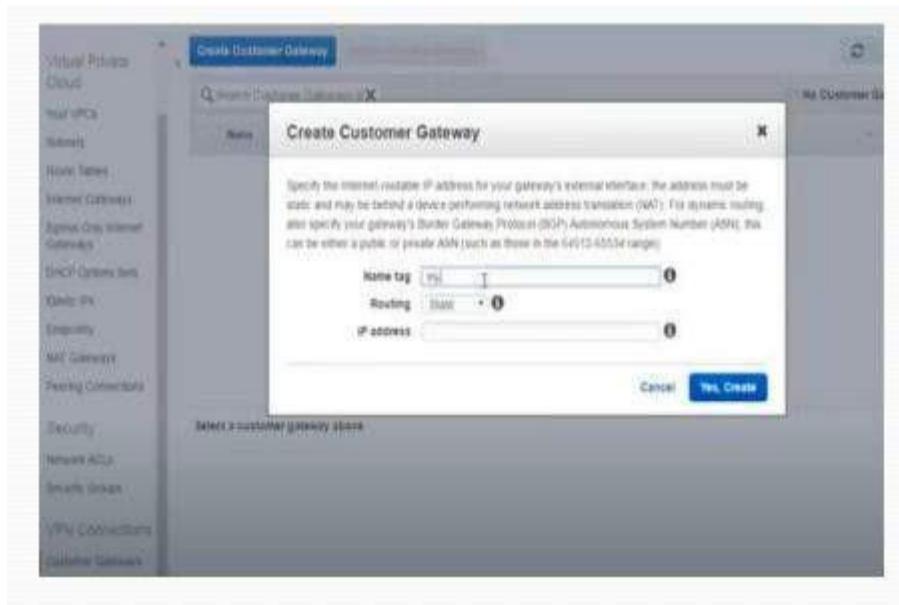
F:\>gcc wikipedija.c -o wikipedija
F:\>wikipedija
Pozdravljenia Wikipedija

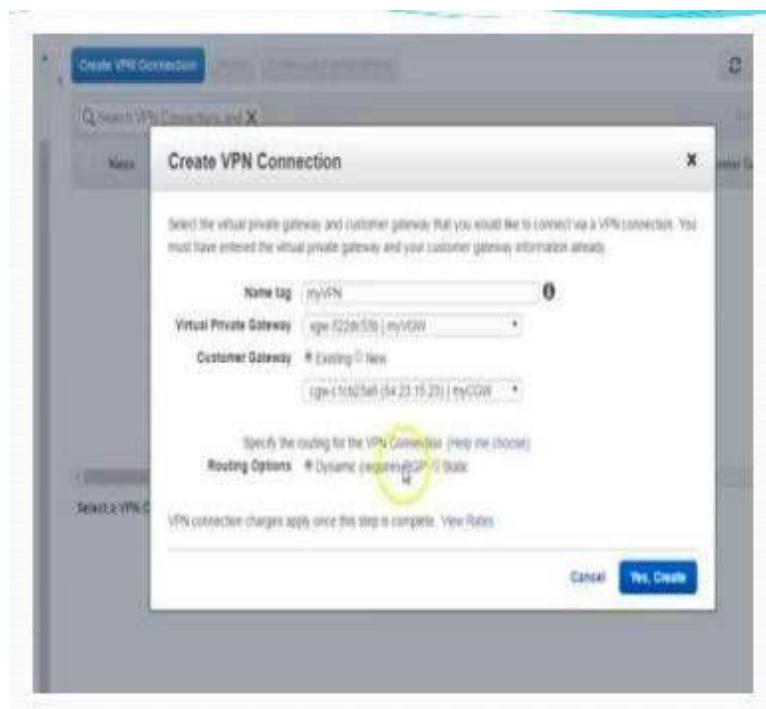
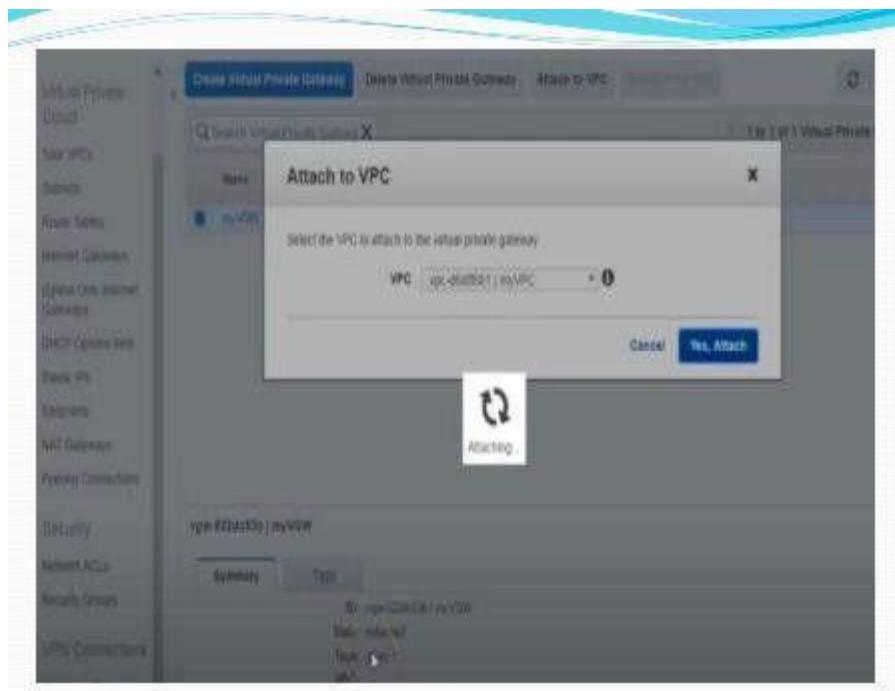
F:\>
```

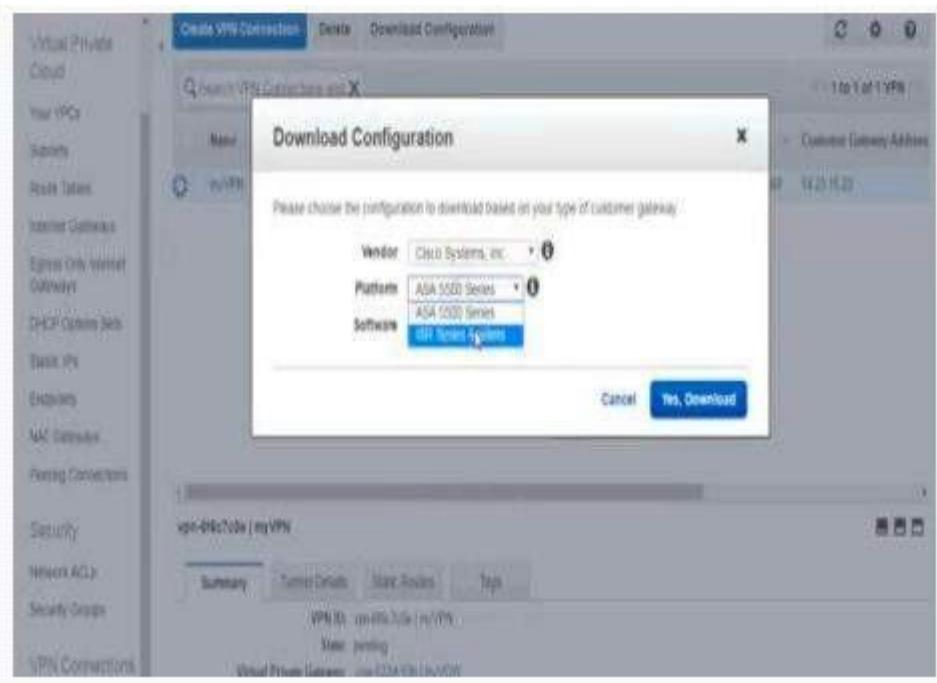
Practical 18: Create VPC and implement EC2 services on it.

The screenshot shows the AWS VPC Dashboard. On the left sidebar, there are several navigation links: Filter by VPC (None), Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups. The main area is titled "Resources" and contains two buttons: "Start VPC Wizard" and "Launch EC2 Instances". Below these buttons, a note says "Note: No instances yet launch in the US East (N. Virginia) region." A list of resources is provided with counts: 1 VPCs, 2 Internet Gateways, 0 Egress-only internet Gateways, 11 Subnets, 0 Route Tables, 0 Elastic IPs, 0 Endpoints, 12 Security Groups, 0 VPN Connections, and 0 Customer Gateways. To the right, the "Service Health" section shows two items: "Amazon VPC - US East (N. Virginia)" with "Service is operating normally" and "Amazon EC2 - US East (N. Virginia)" with "Service is operating normally". There is also a link to "View complete service health details". Below this, the "Additional Information" section includes links to "VPC Documentation", "All VPC Resources", "Filters", and "Report an issue".

The screenshot shows the AWS VPC Dashboard with the "Create VPC" button highlighted. The search bar at the top has "myVPC" entered. The main table displays one VPC entry: "myVPC" with VPC ID "vpc-0d6f9a1", Status "Available", and IPv4 CIDR "10.0.0.0/26". The table also includes columns for "IPv6 CIDR", "DHCP options set", "Route table", and "Network ACL". At the bottom of the table, there is a summary: "vpc-0d6f9a1 | myVPC".







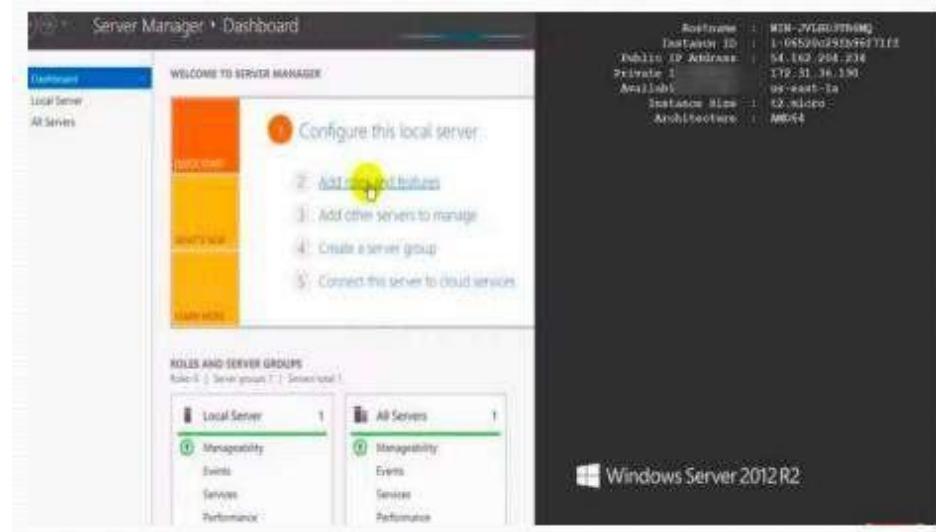
```

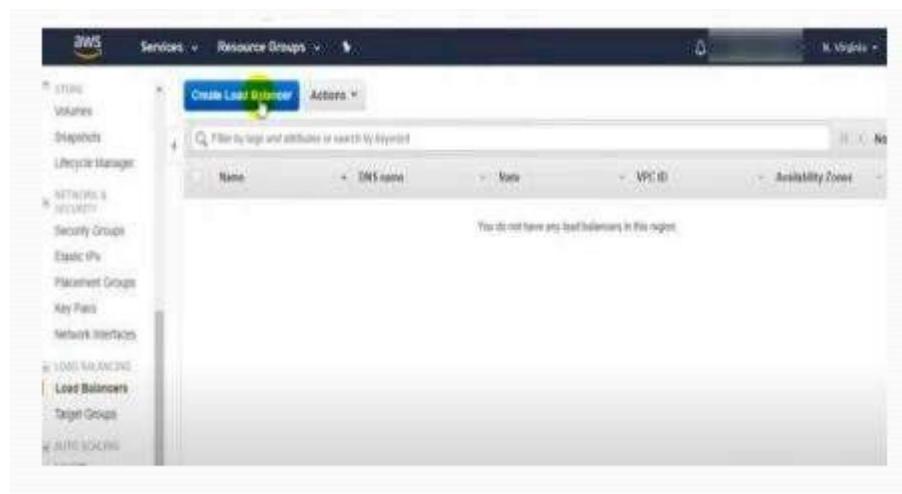
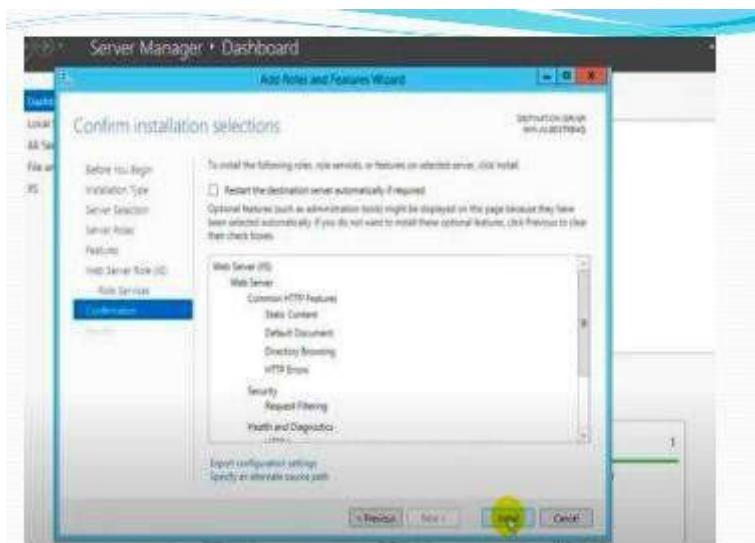
1 ! Amazon Web Services
2 ! Virtual Private Cloud
3 !
4 ! AWS utilizes unique identifiers to manipulate the configuration of
5 ! a VPN Connection. Each VPN Connection is assigned an identifier and is
6 ! associated with two other identifiers, namely the
7 ! Customer Gateway Identifier and Virtual Private Gateway Identifier.
8 !
9 ! Your VPN Connection ID : vpn-6f6c7c0e
10 ! Your Virtual Private Gateway ID : vgw-522dc53b
11 ! Your Customer Gateway ID : cgw-clcb13a1
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> - VPN 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 !
30 ! #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="application-lb"/>
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 listens 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 control 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-31e9400a	default	default VPC security group
sg-0153b6848670c2d	ECS-SG	Launch wizard-1 created 2019-11-09T17:20:21.327+00:00

Step 4: Configure Routing

Name:

Target type:

- Instance
- IP
- Lambda function

Protocol:

Port:

Health checks

Protocol	<input type="text" value="HTTP"/>
Path	<input type="text" value=""/>

[Advanced health check settings](#)

[Cancel](#) [Previous](#) [Next](#)

The screenshot shows the 'Step 5: Register Targets' page. At the top, there are tabs: 1. Configure Load Balancer, 2. Configure Security Settings, 3. Configure Security Groups, 4. Configure Routing, 5. Register Targets (which is underlined), and 6. Review. Below the tabs, the heading 'Step 5: Register Targets' is displayed with the sub-instruction: 'Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and passes the initial health checks.' A note below says: 'To deregister instances, select one or more registered instances and then click Remove.' The main area shows a table titled 'Instances' with two rows:

Instance	Name	Port	Status	Security groups	Zone
i-0b1407ab359fc...	ELBSENG1	80	green	ELBSG1	us-east-1a
i-06026298a077f...	ELBSENG2	80	green	ELBSG1	us-east-1a

Below the table, there is 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. At the top, there are tabs: 1. Configure Load Balancer, 2. Configure Security Settings, 3. Configure Security Groups, 4. Configure Routing, 5. Register Targets, and 6. Review (which is underlined). Below the tabs, the heading 'Step 6: Review' is displayed with the sub-instruction: 'Tags'. The main area shows the configuration details:

- Security groups:** `sg-0110234567890c2e`
- Routing:**
 - Target group: New target group
 - Target group name: TG1
 - 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 Actions

Create target group

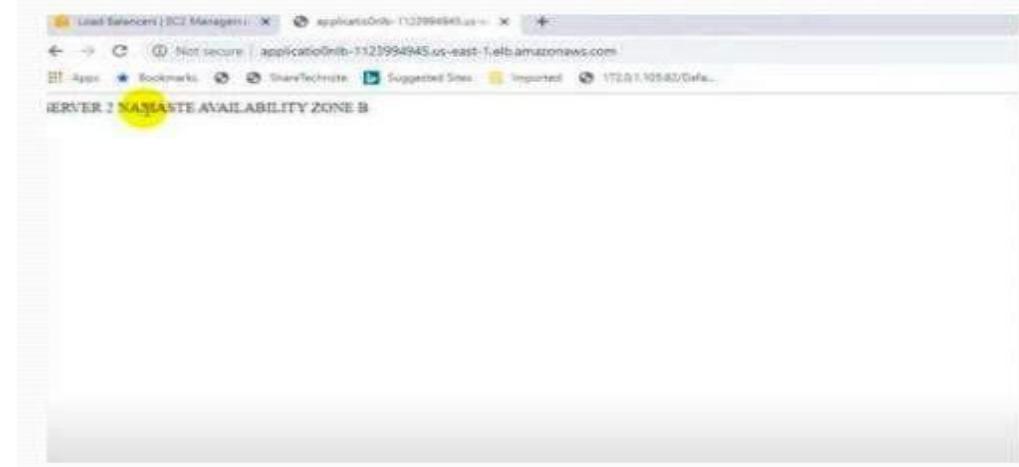
Name	Port	Protocol	Target type	Load Balancer	VPC ID	Region
TG1	80	HTTP	Instance	Application	vpc-7ab9303	us-east-1

Registered targets

Instance ID	Name	Port	Availability Zone	Status	Description
i-0b16174d035a0fbc	ELB SERVER2	80	us-east-1a	initial	Target registration is in progress
i-08205236960718	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.

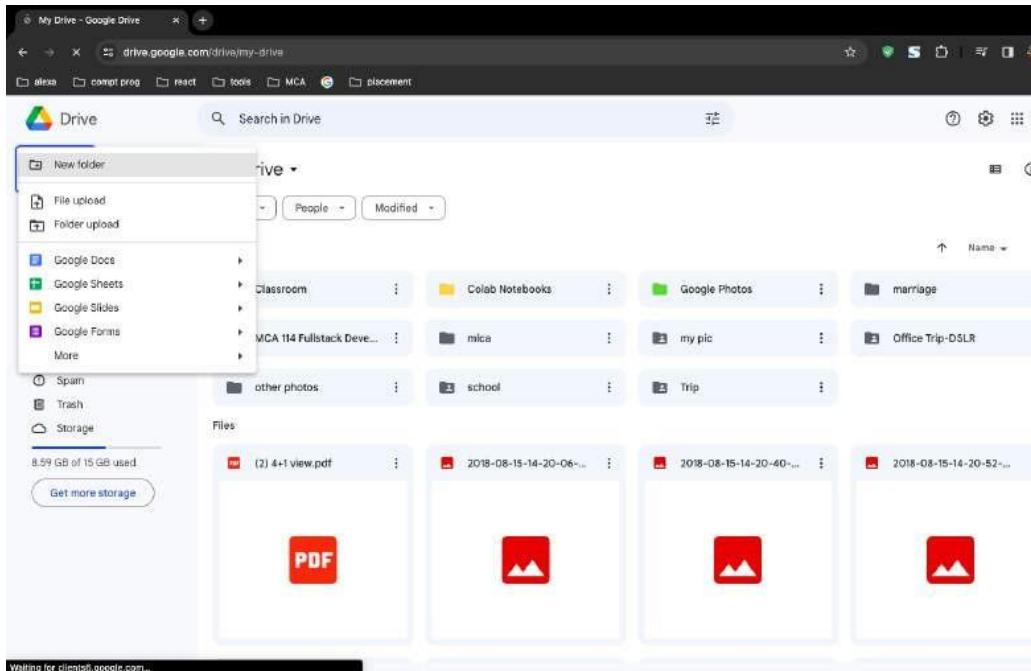
Steps to Create a Private Cloud on Google Drive:

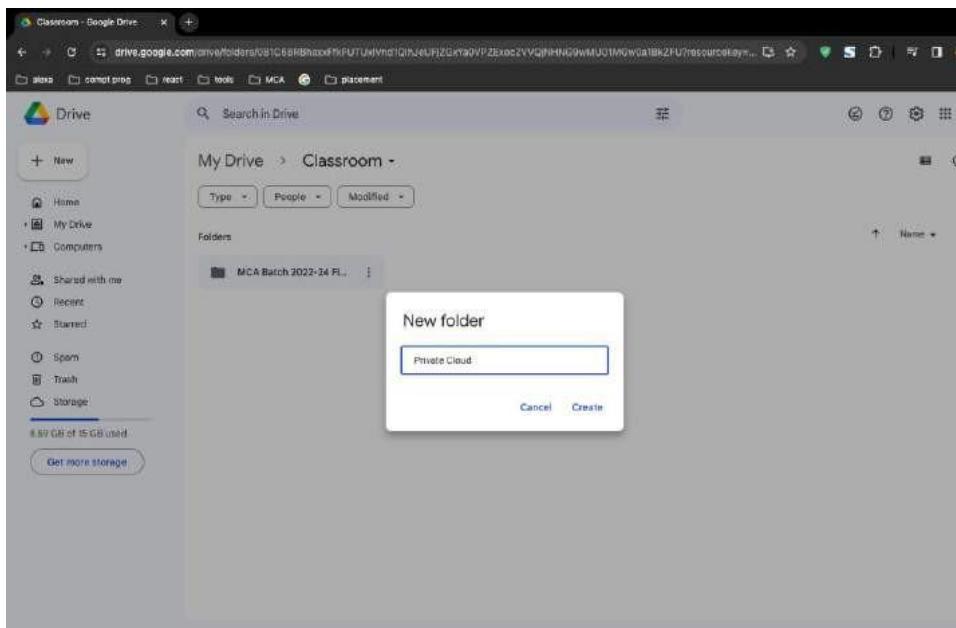
1. Sign in to Google Drive:

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:

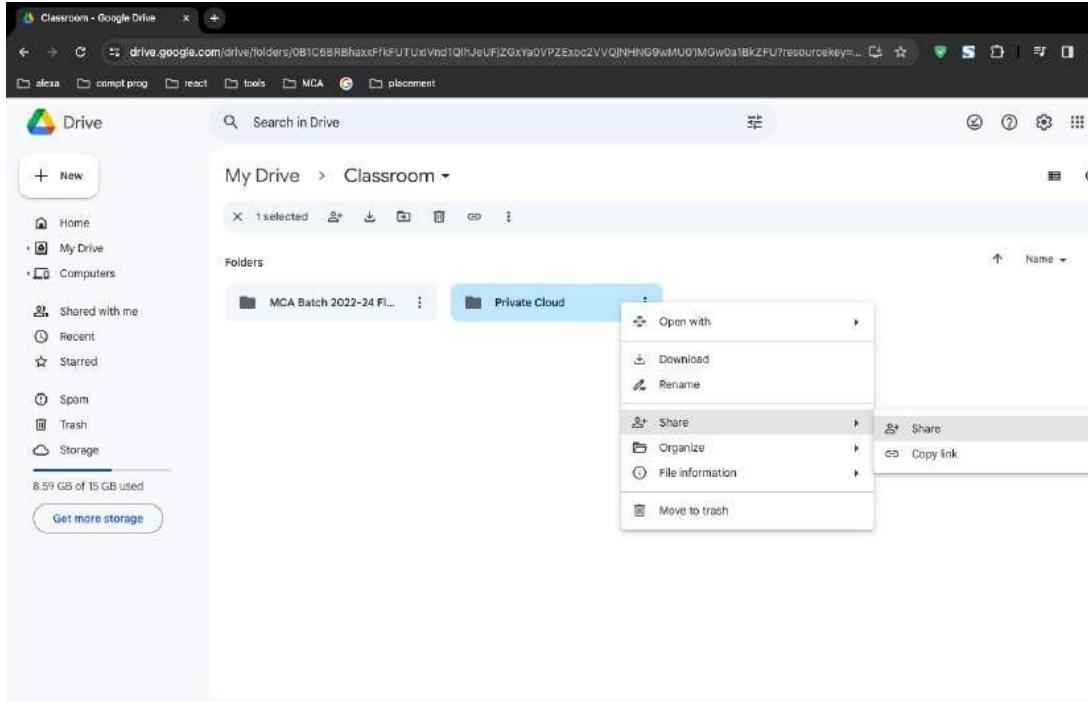
Click on the "+ New" button on the left side.
Choose "Folder" to create a new folder.
Name the folder appropriately, e.g., "Private Cloud."





3. Share the Folder:

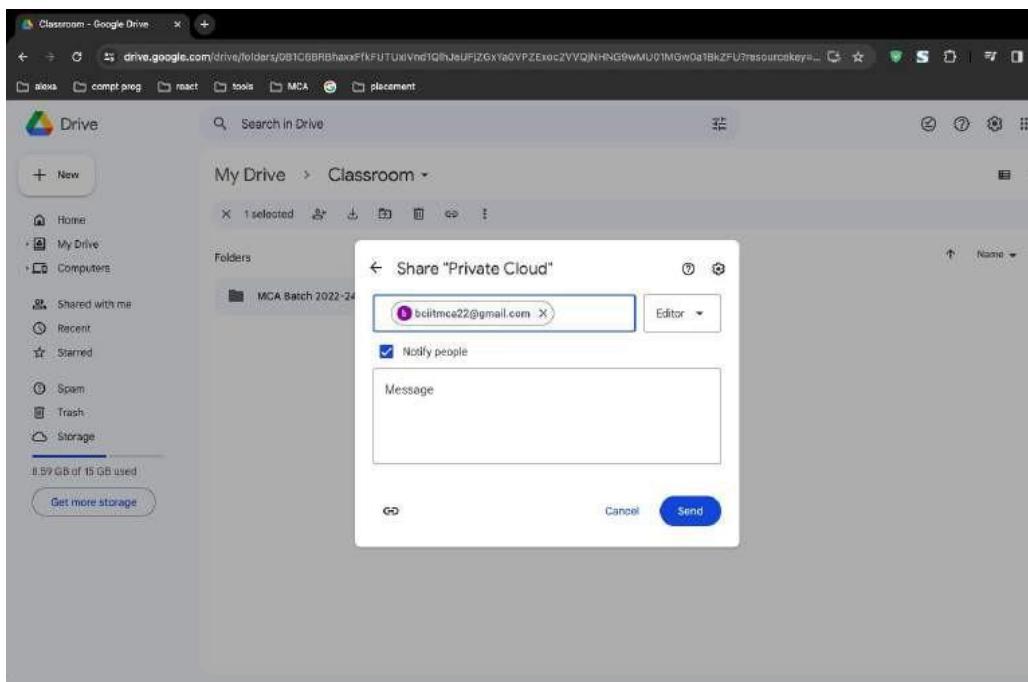
Right-click on the folder you just created.
Select "Share."



4. Add Users:

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

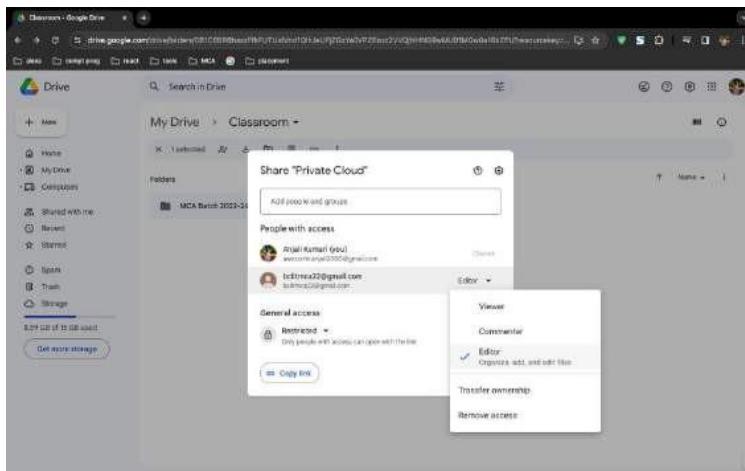
Choose the appropriate access level (e.g., Viewer, Commenter, Editor) based on the level of access you want to provide.



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.

