DAY 16 **DATE:17/05/2025**

NAME: ANNIE JOHN

USER ID:27739

Batch: 25VID0885_DC_Batch4

TITLE: AMAZON VPC (VIRTUAL PRIVATE CLOUD) AND CREATION OF VPC, INSTANCE IN EC2 AND S3 BUCKET

- **What is Amazon VPC?**

  Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a logically isolated virtual network that you define. It closely mirrors a traditional on-premises network setup, while benefiting from the scalable infrastructure of AWS.

- # Amazon VPC Concepts

  Amazon VPC serves as the **networking layer** for Amazon EC2 and other AWS services. Below are the core concepts and components associated with VPC:

  ### 1. Virtual Private Cloud (VPC)
  A logically isolated section of the AWS cloud dedicated to your account where you can define your own virtual network topology, including IP address ranges, subnets, route tables, and network gateways.

  ### 2. Subnet
  A subset of your VPC's IP address range where you can place groups of isolated resources. Subnets can be **public** (accessible from the internet) or **private** (not accessible directly from the internet).

  ### 3. CIDR Block
  CIDR (Classless Inter-Domain Routing) block defines the IP address range for the VPC or subnet. For example, `10.0.0.0/16` provides 65,536 private IP addresses.

  ### 4. Route Table
  A set of rules (routes) used to determine the path of outbound traffic from the subnets to other networks, such as the internet or other VPCs.

  ### 5. DHCP Options Set
  Configuration options such as domain name, DNS servers, and NTP servers that are passed to instances launched in the VPC.

  ### 6. Internet Gateway
  A horizontally scaled, redundant, and highly available gateway that enables communication between instances in your VPC and the internet.

  ### 7. Egress-Only Internet Gateway
  Used with IPv6 traffic, this gateway allows outbound communication to the internet but blocks inbound traffic, enhancing security for IPv6-enabled instances.

  ### 8. VPC Endpoint

Enables private connectivity between your VPC and supported AWS services, without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect.

9. NAT Gateway

A managed service that allows instances in a private subnet to initiate outbound connections to the internet while preventing inbound connections initiated by the internet.

10. NAT Instance

An EC2 instance configured to provide the same function as a NAT Gateway. This is a more manual and less scalable alternative to the managed NAT Gateway.

11. Carrier Gateway

Used in Wavelength Zones, it enables connectivity between your VPC and a telecommunications carrier network, supporting both inbound and outbound traffic.

12. Prefix Lists

A set of one or more CIDR blocks that you can use as a reference in security group and route table configurations. These can also be shared across AWS accounts via Resource Access Manager (RAM).

13. Security Groups

Virtual firewalls attached to EC2 instances that control inbound and outbound traffic. Security groups are stateful and apply at the instance level.

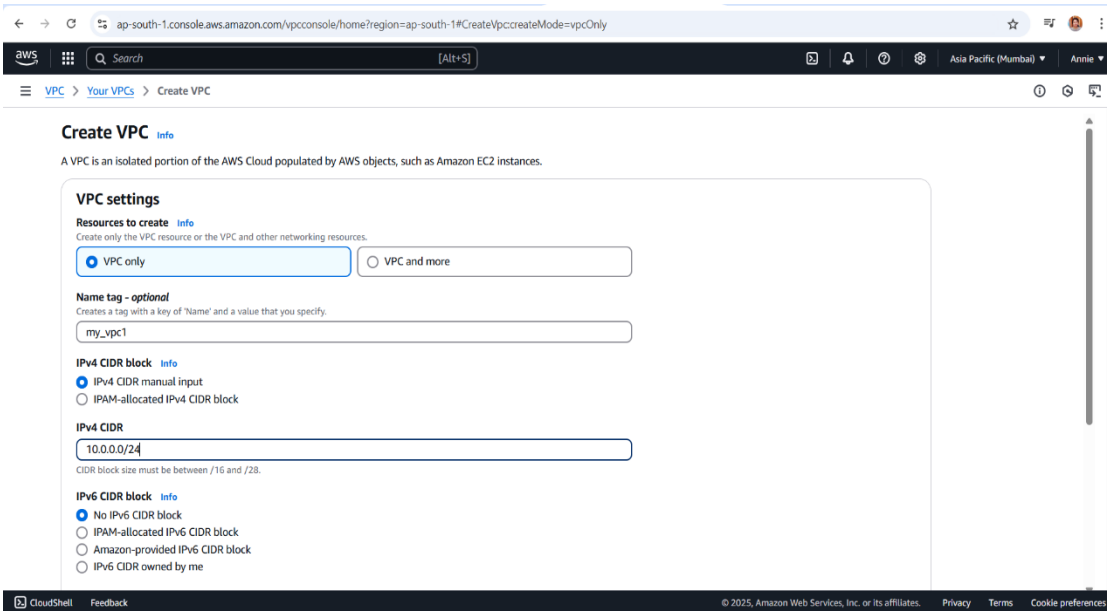
14. Network ACLs (Access Control Lists)

An optional layer of stateless security at the subnet level that controls inbound and outbound traffic. Unlike security groups, NACLs are not stateful and can be used to explicitly allow or deny traffic.

➢ Create your VPC

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/

2. On the **VPC Dashboard**, choose **Create VPC**.
   Under **VPC Settings** select VPC only and give your VPC name and **IP CIDR block 10.0.0.0/24(IPV4).**



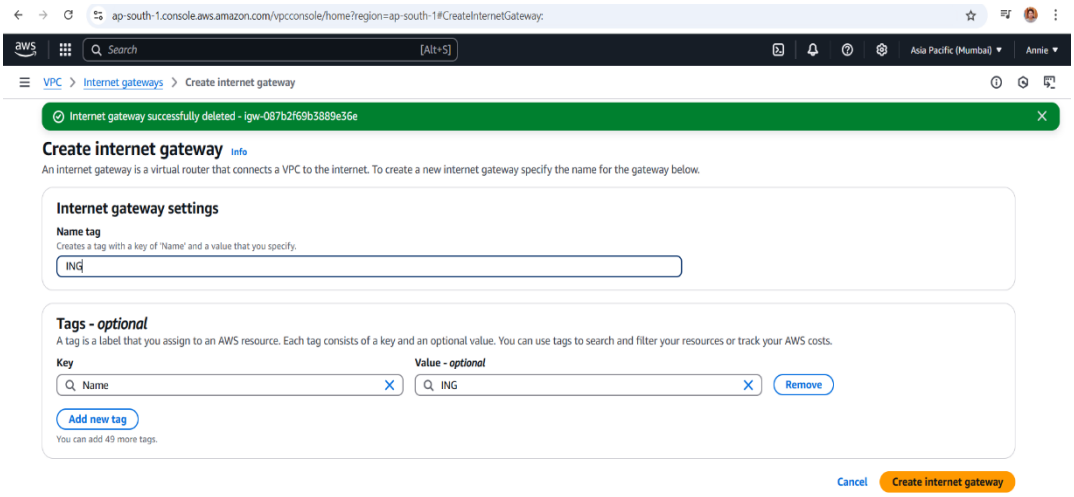
3. Choose **Create VPC**.
4. Create and attach an **internet gateway** to a VPC

- An **Internet Gateway** is a horizontally scaled, redundant, and highly available VPC component that enables communication between instances in your VPC and the internet.
- Key Functions:

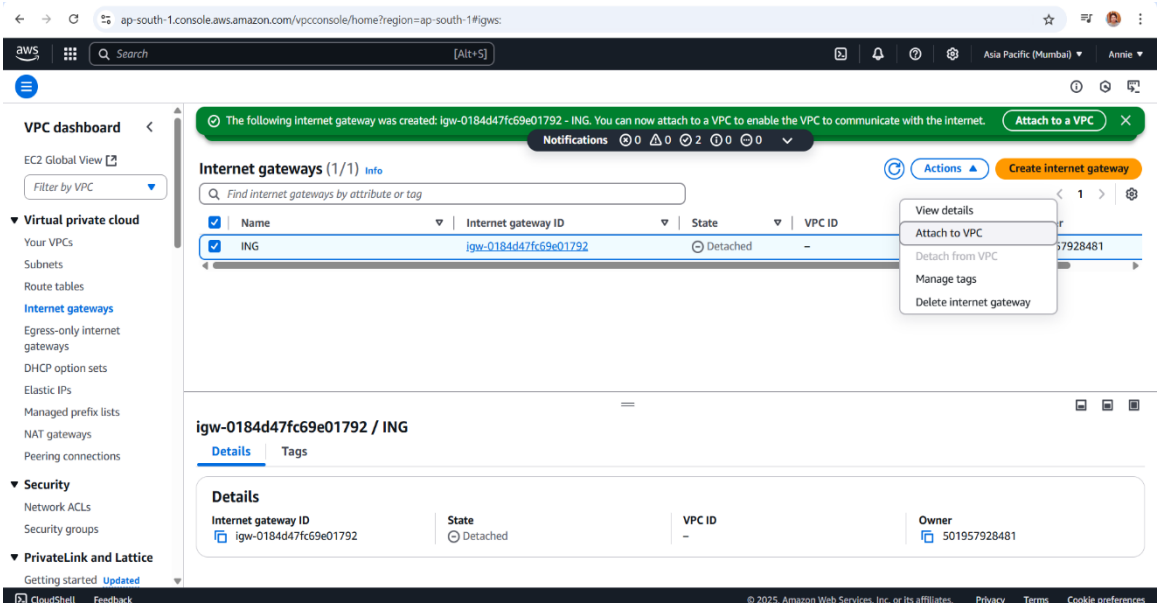
i. Route Target: Acts as a target in your route tables for traffic destined for the internet.
ii. NAT Functionality: Performs network address translation (NAT) for instances with public IPv4 addresses, allowing them to communicate with the internet.
iii. IPv6 Support: Fully supports both IPv4 and IPv6 traffic.
iv. No Extra Charges: There is no cost to create or use an internet gateway.
v. No Bottlenecks: It does not limit bandwidth or create availability risks.

- **Steps to Create and Attach an Internet Gateway**

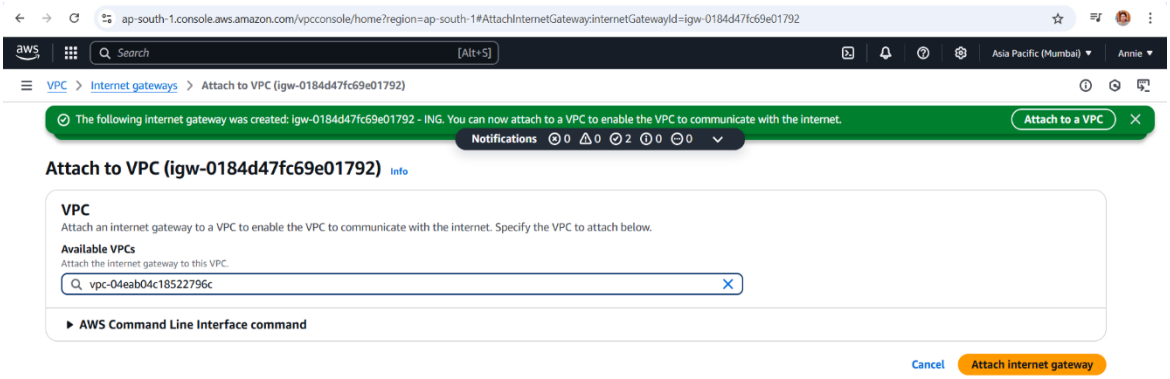
1. **Navigate to Internet Gateways**->select **Create Internet Gateway**->**Name the Internet Gateway (Optional)**->**Create the Internet Gateway**->Click **Create Internet Gateway**.



- **Attach the Internet Gateway to a VPC**
  - Select the internet gateway you just created.
  - Click **Actions → Attach to VPC**.



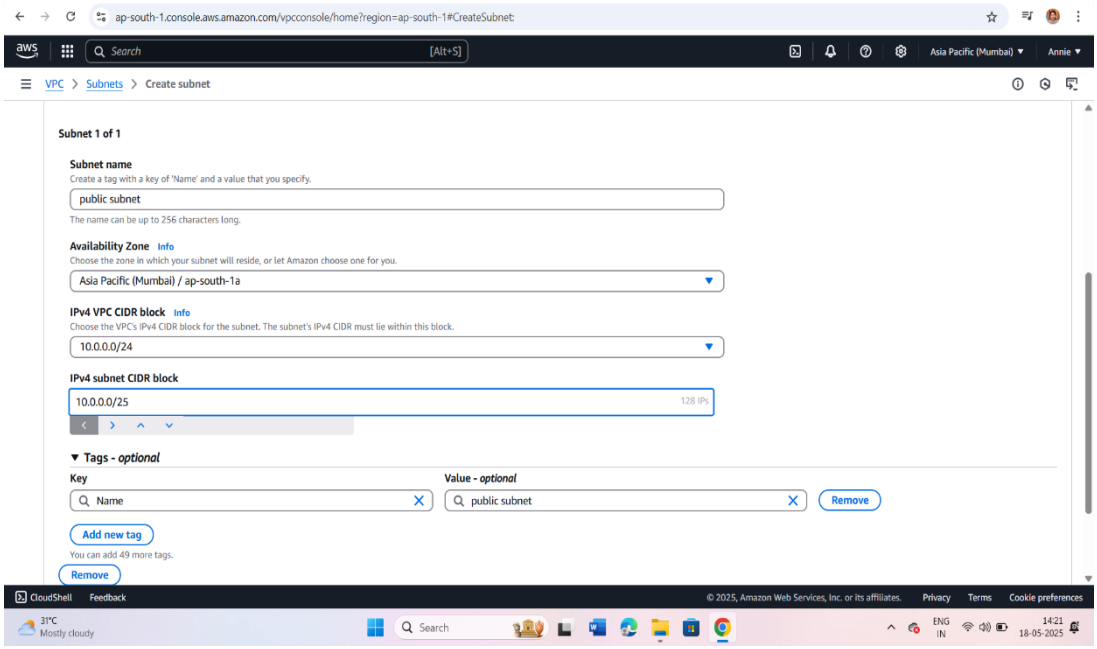
  - Choose your VPC from the dropdown list.
  - Click **Attach Internet Gateway**.



5. Create Subnets

- **Steps to Create a Public Subnet**

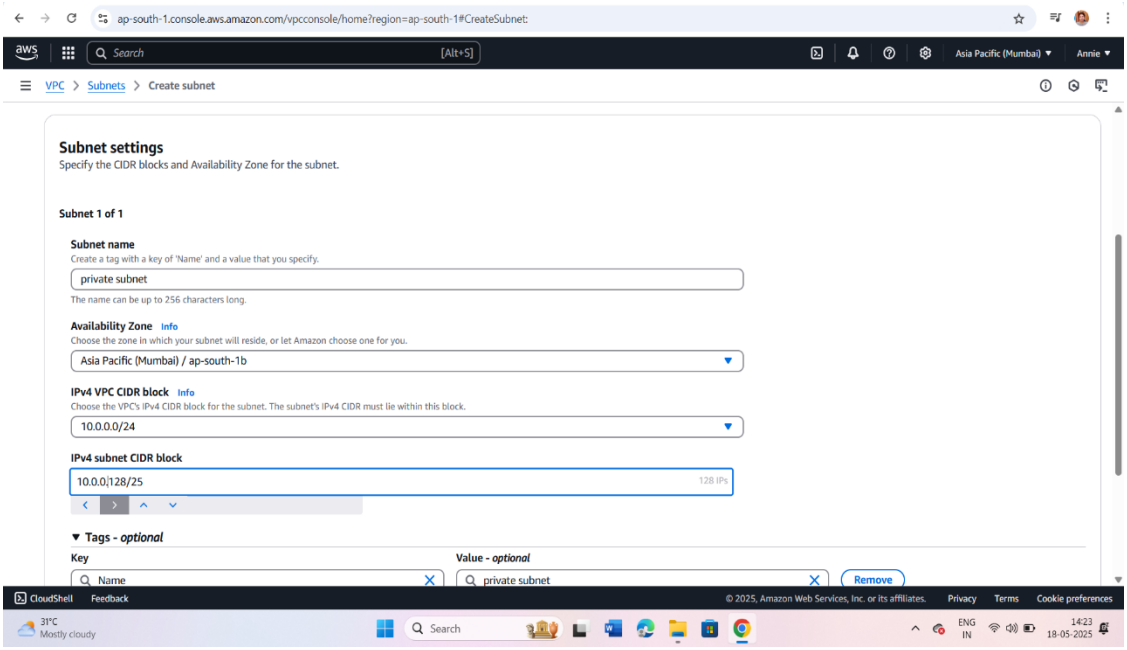
  Go to Subnets ->Click Create Subnet->Enter Subnet Name->Choose VPC Select the same VPC you created earlier->Choose a Different Availability Zone)->Enter CIDR Block (eg:10.0.0.0/25 -dividing 256 ip to 128)->Click Create Subnet.



- **Steps to Create a Private Subnet**

  Go to Subnets ->Click Create Subnet->Enter Subnet Name->Choose VPC Select the same VPC you created earlier->Choose a Different Availability

Zone)->Enter CIDR Block (eg:10.0.0.128/25 -dividing 256 ip to 128)->Click Create Subnet.



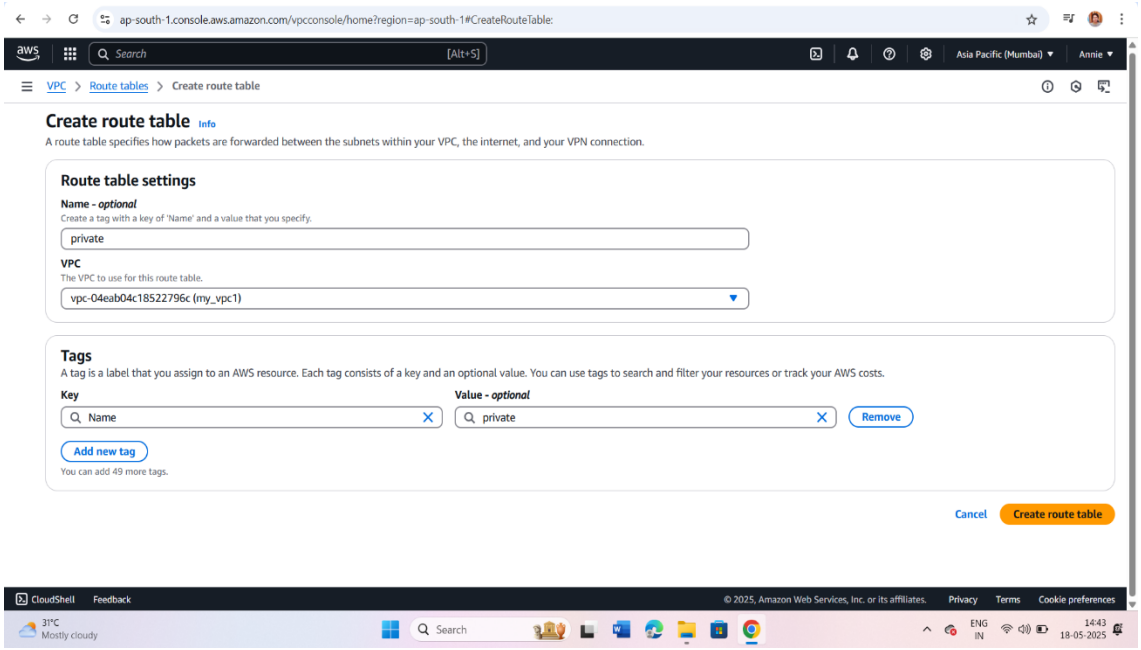
6. Route tables:

A *route table* contains a set of rules, called *routes*, that are used to determine where network traffic from your subnet or gateway is directed.
The following are the key concepts for route tables.

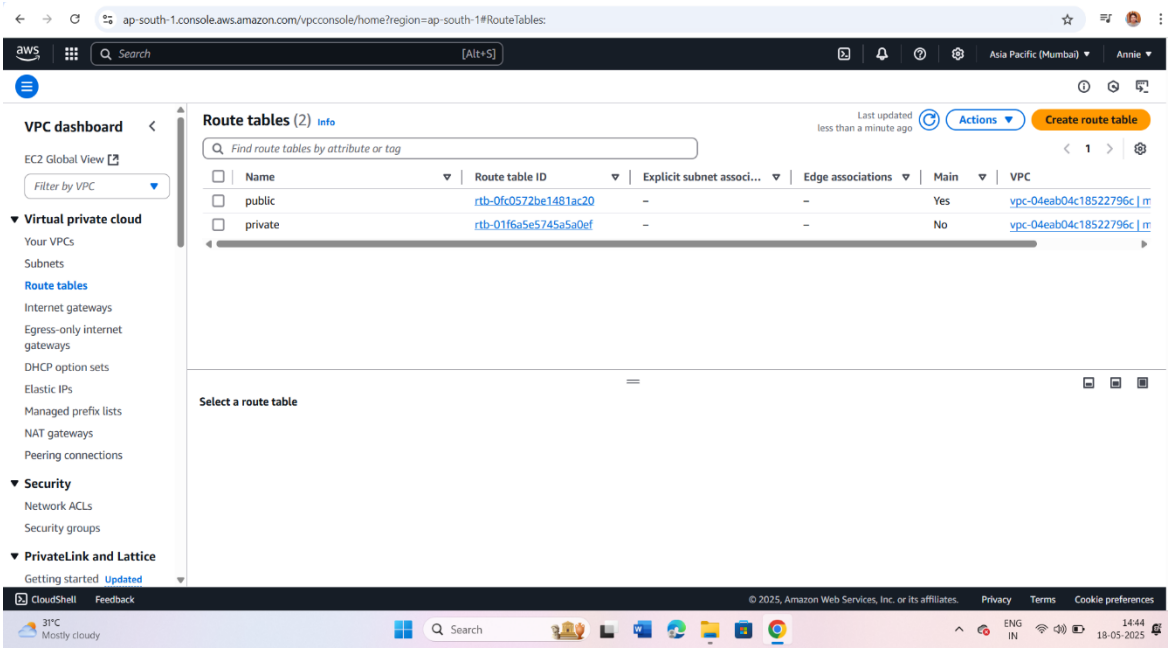
- **Main route table**—The route table that automatically comes with your VPC. It controls the routing for all subnets that are not explicitly associated with any other route table.
- **Custom route table**—A route table that you create for your VPC.

✓ **Create a custom route table**

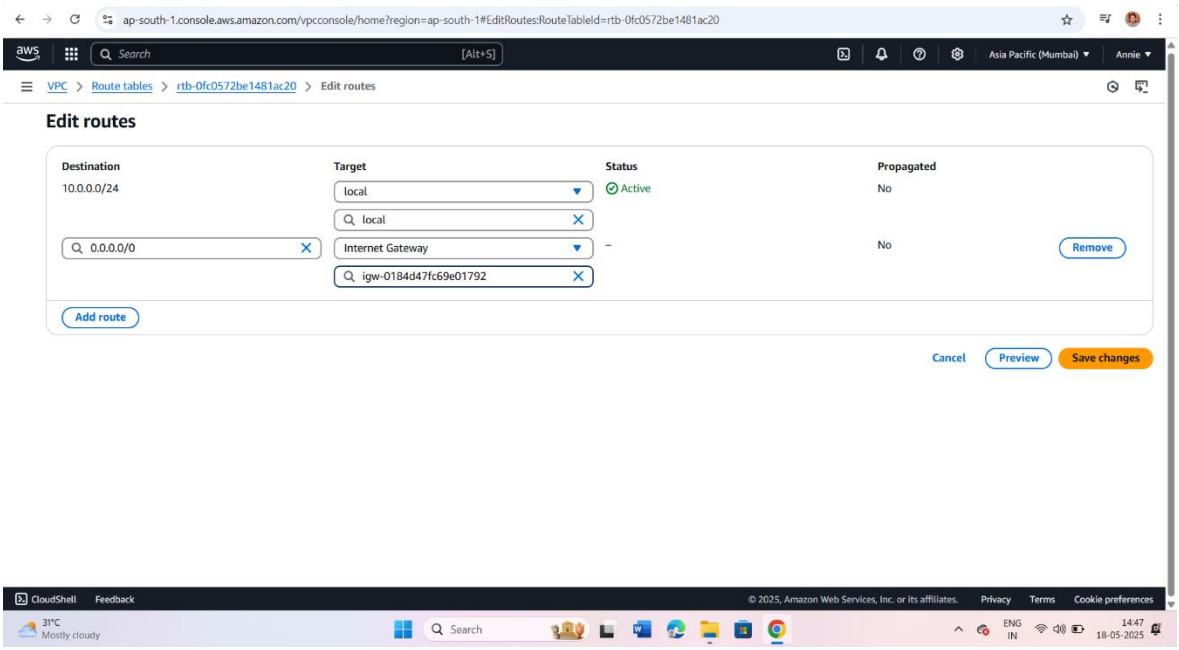
a. Go to Route Tables → Click Create Route Table → Enter a Name (e.g., Private Route Table) → Choose the same VPC → Click Create



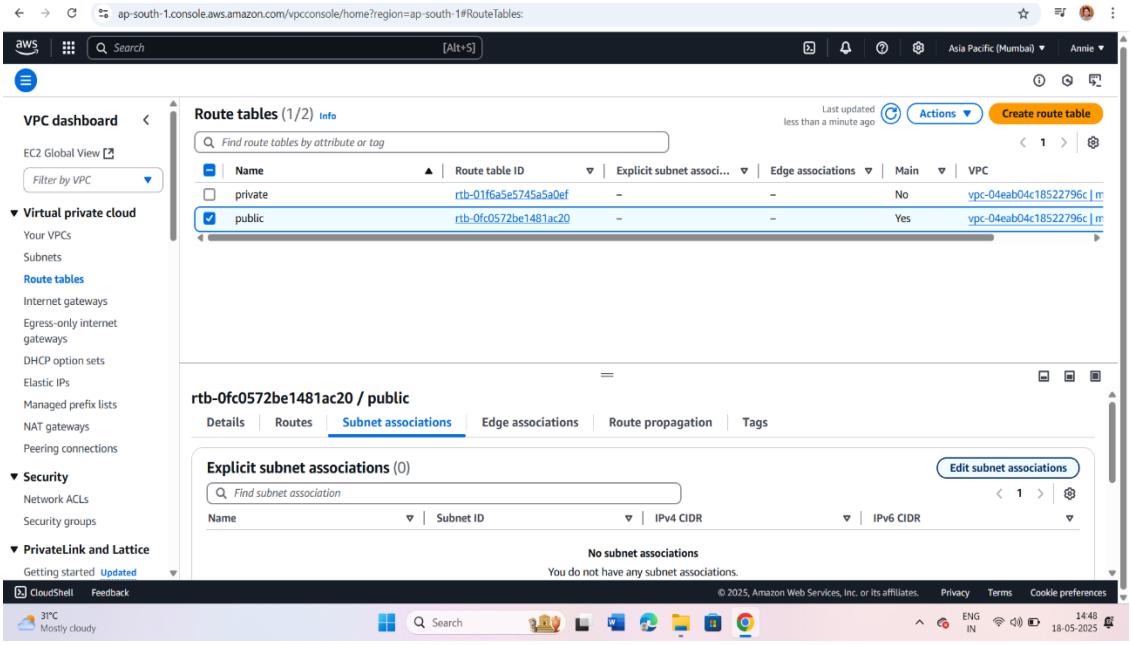
b. Here there will be one default route table name it as public.



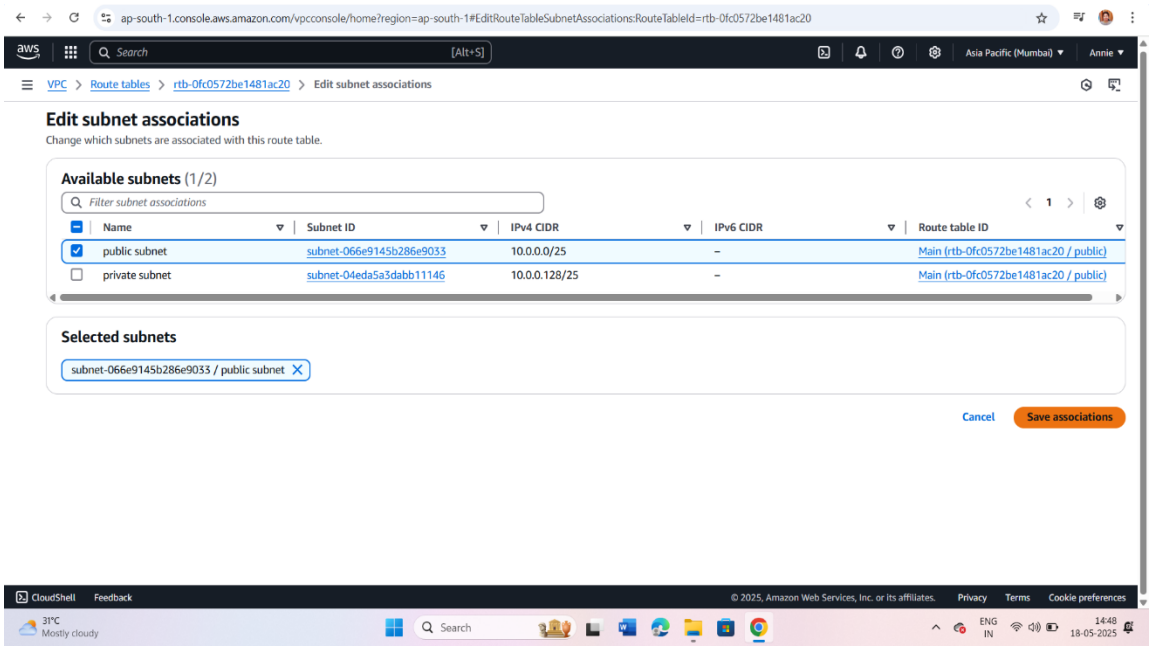
c. Select the Route Table you just created(I choose public)→ Go to Routes tab → Click Edit Routes→ Click Add Route → Enter Destination: 0.0.0.0/0 → Target: Select Internet Gateway→ Click Save Changes



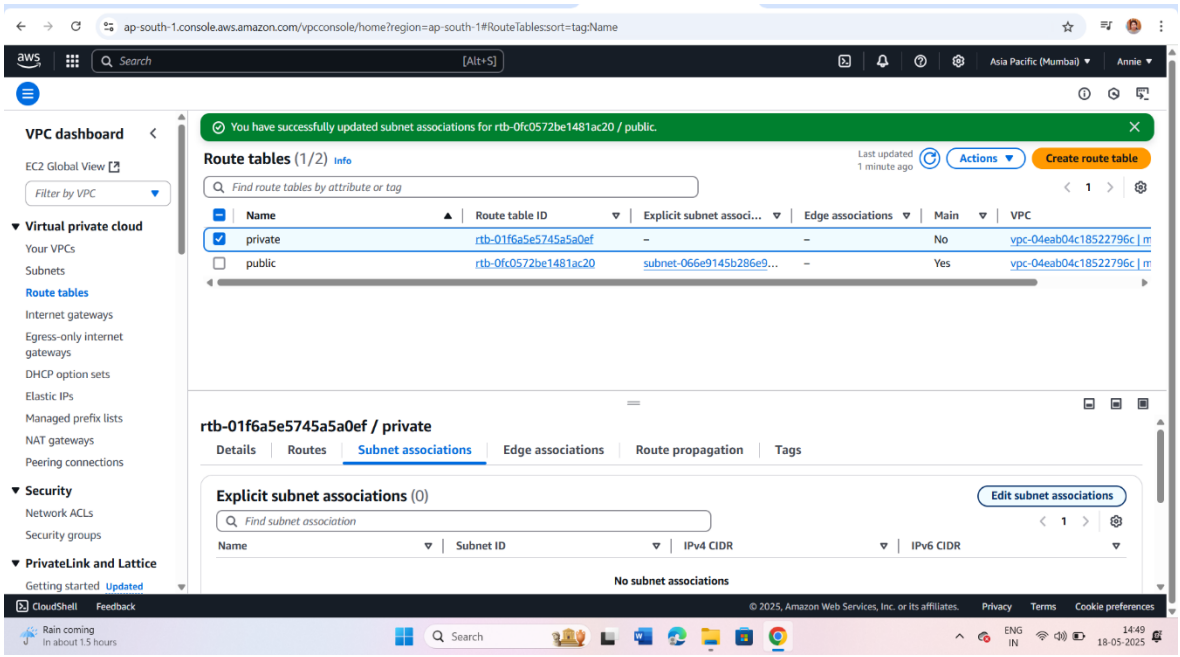
d. Go to Subnet Associations tab → Click Edit Subnet Associations

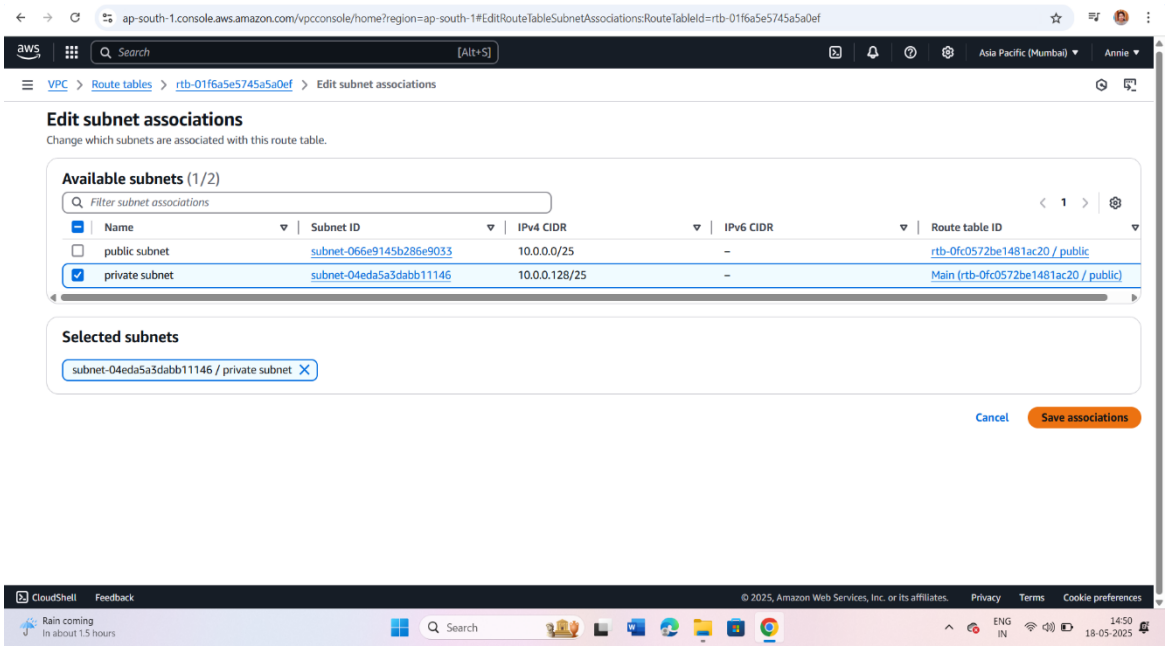


e. Select the Public Subnet → Click Save Associations



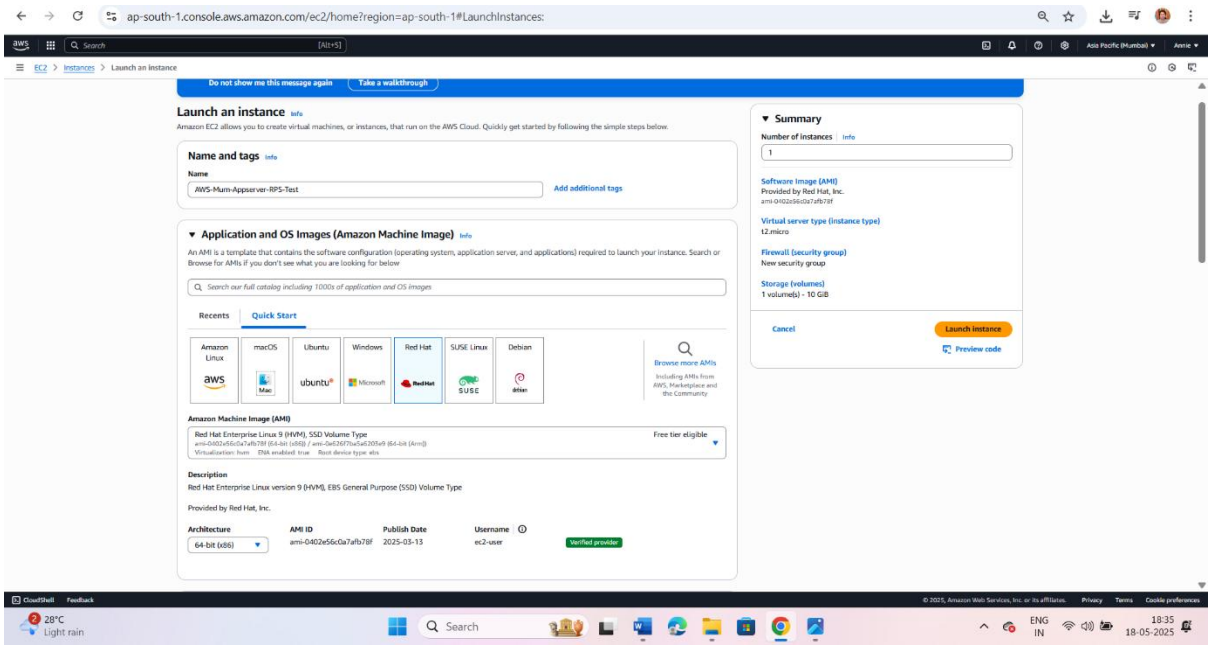
f. Since we are keeping private route table as private, we only do subnet association for private net we will not edit route table.



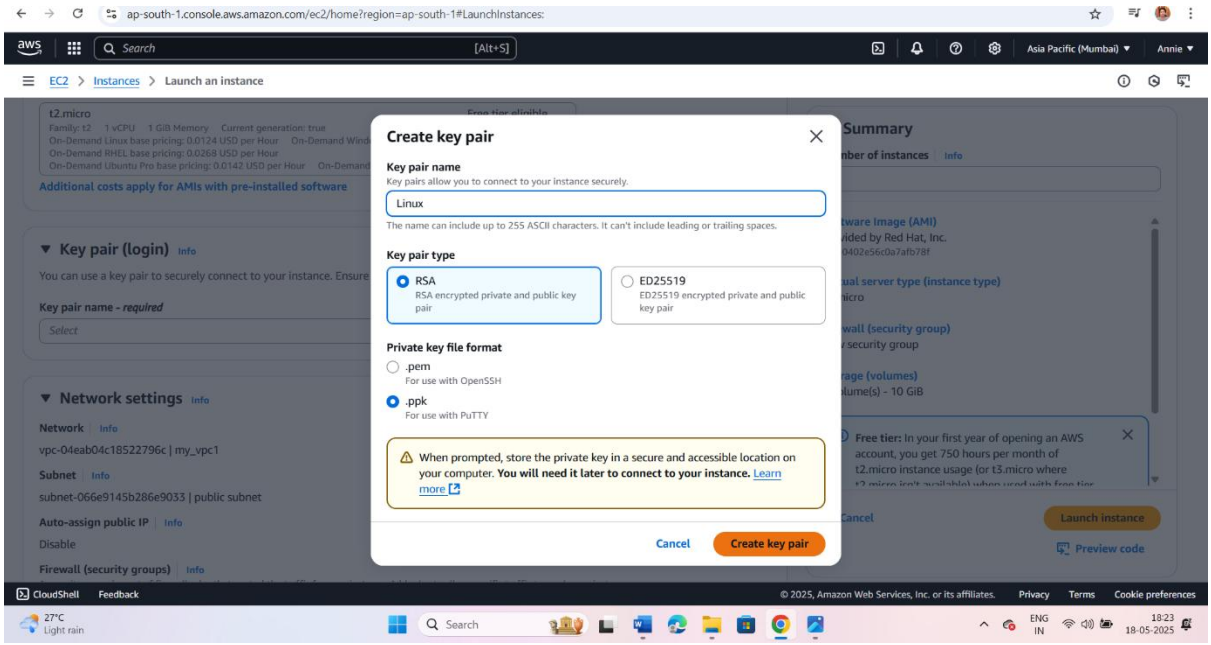


- **EC2 Instance (Amazon Elastic Compute Cloud):**
  An EC2 instance is a virtual server in the AWS cloud used to run applications, host websites, store data, or perform computations — just like a physical computer, but scalable, flexible, and pay-as-you-go.
- Instance creation in EC2

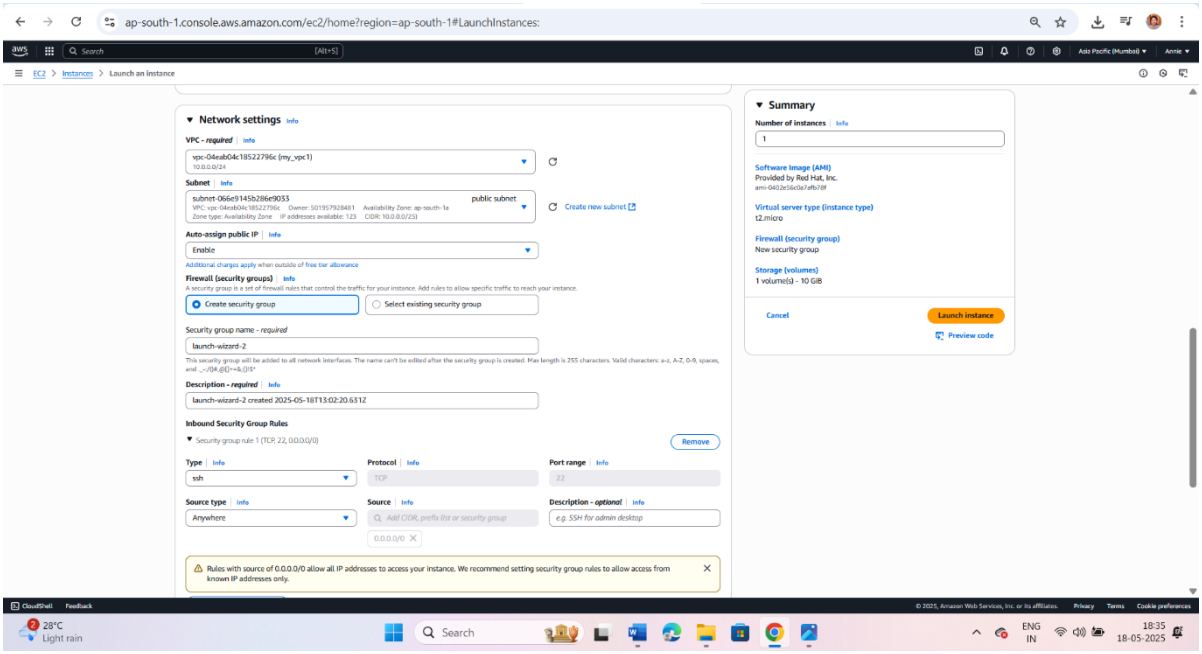
1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. Click on **Launch instance** for creating an instance->give name ->select OS image(choose Red Hat).



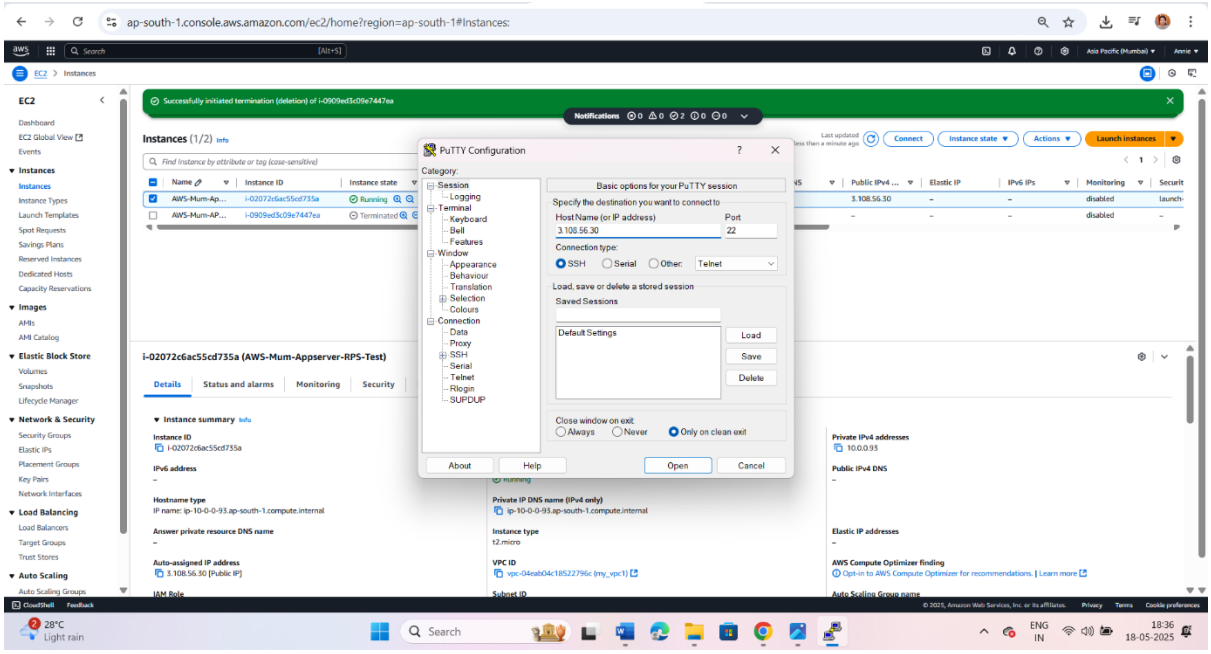
- Create a **key pair** choose RSA**,** give name for **key pair**(since Linux is chosen select file format ‘.ppk’).



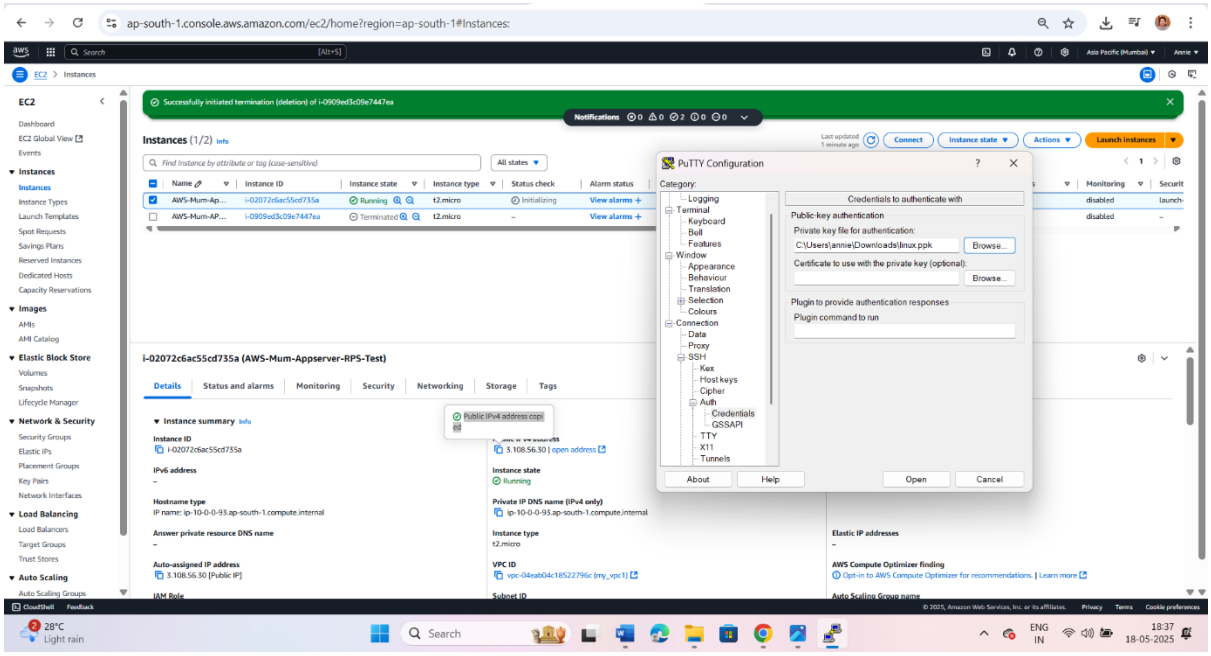
- Edit Network Settings->Enable auto-assign public IP->create security group use ssh.



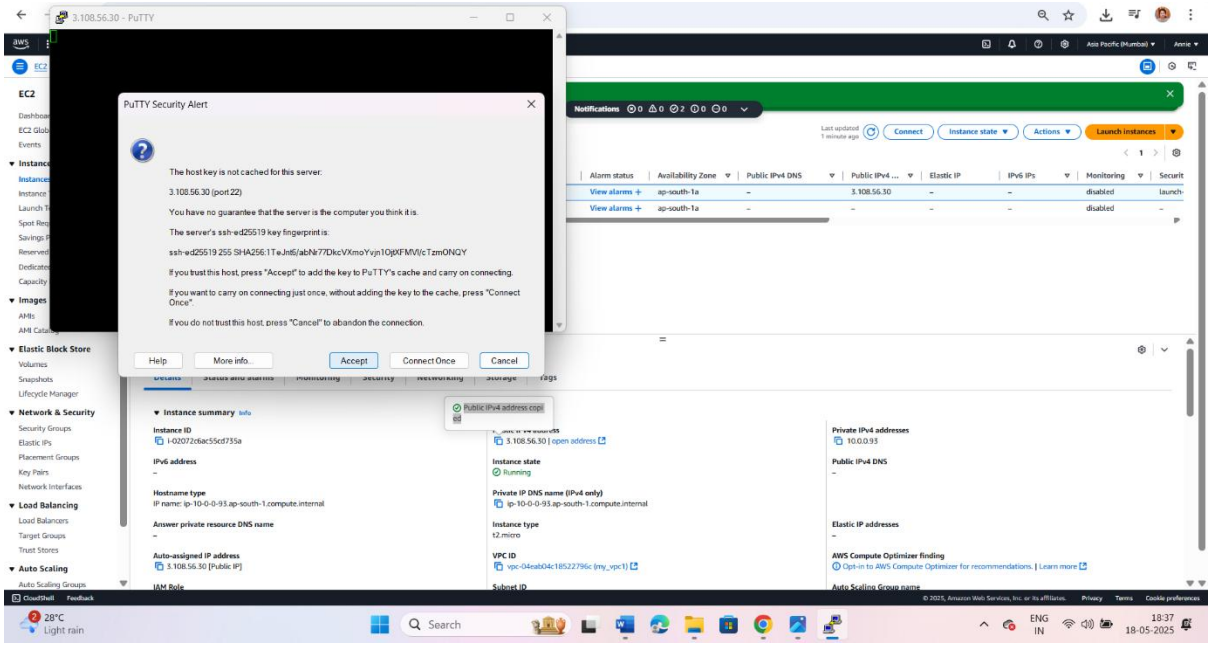
3. Created instance->open **PuTTY**->give public IP.



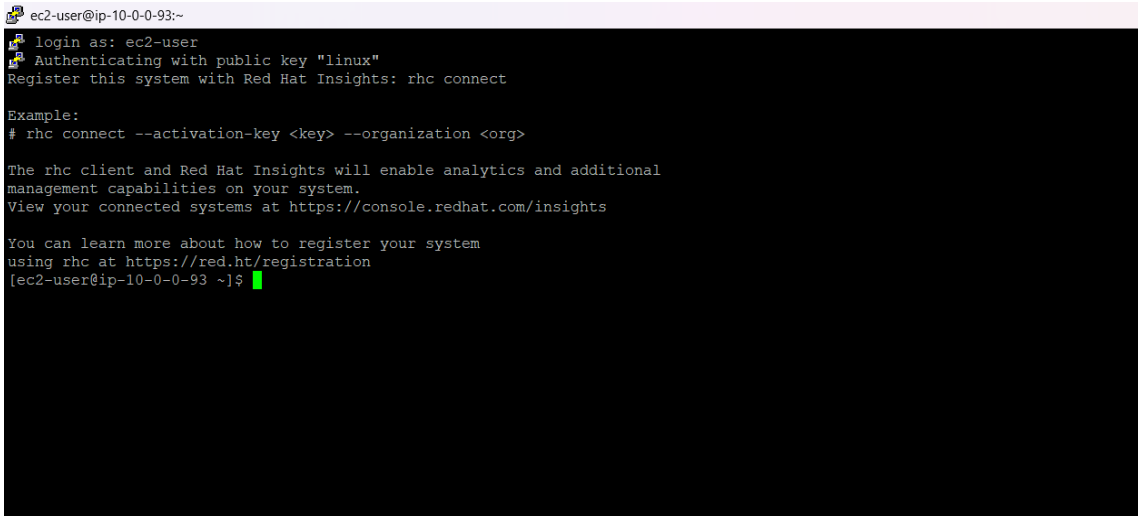
4. In SSH->Auth->Credentials->Browse the downloaded key pair.



5. Click Open->accept.



6. Give your username :-ec2-user



7. Successfully logged into your **EC2 instance** using SSH.

This Means:

- You connected to the EC2 instance as the user: ec2-user
- You're using a **Linux-based EC2 instance** (Red Hat)
- The login used **public key authentication** ("linux" key pair)
- You're now at the **Linux command-line terminal** of your EC2 server.

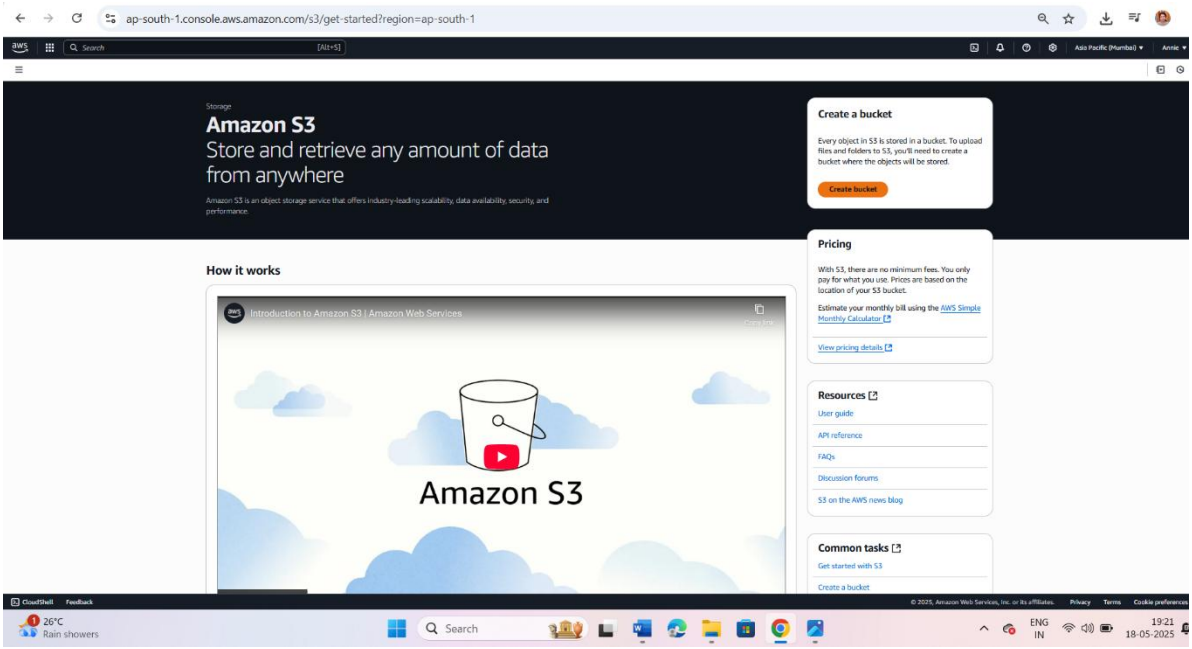
➢ Amazon S3 (Simple Storage Service)

Amazon S3 is a **scalable, durable cloud storage service** offered by AWS. You use it to **store and retrieve any amount of data, at any time, from anywhere** on the web.

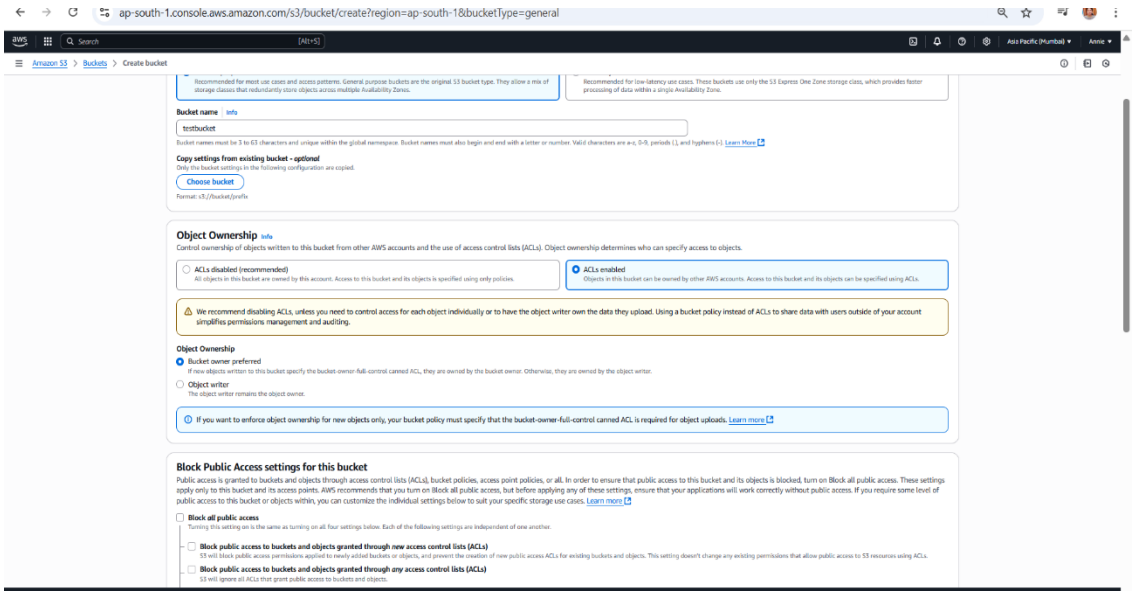
- **What Can You Store in S3?**
  a. Documents, images, videos
  b. Website files (HTML, CSS, JS)
  c. Backups and logs
  d. Data for analytics
  e. Application data (mobile/web apps)

➢ Creating S3

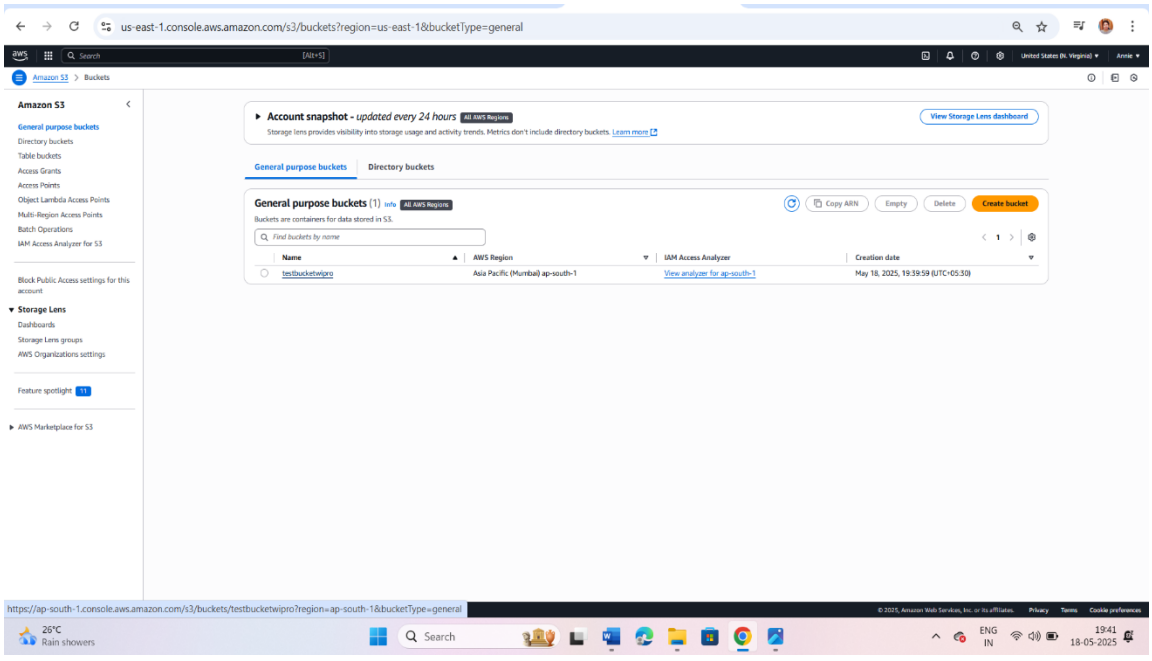
- Open Amazon S3 console.
- Cleck on create bucket.



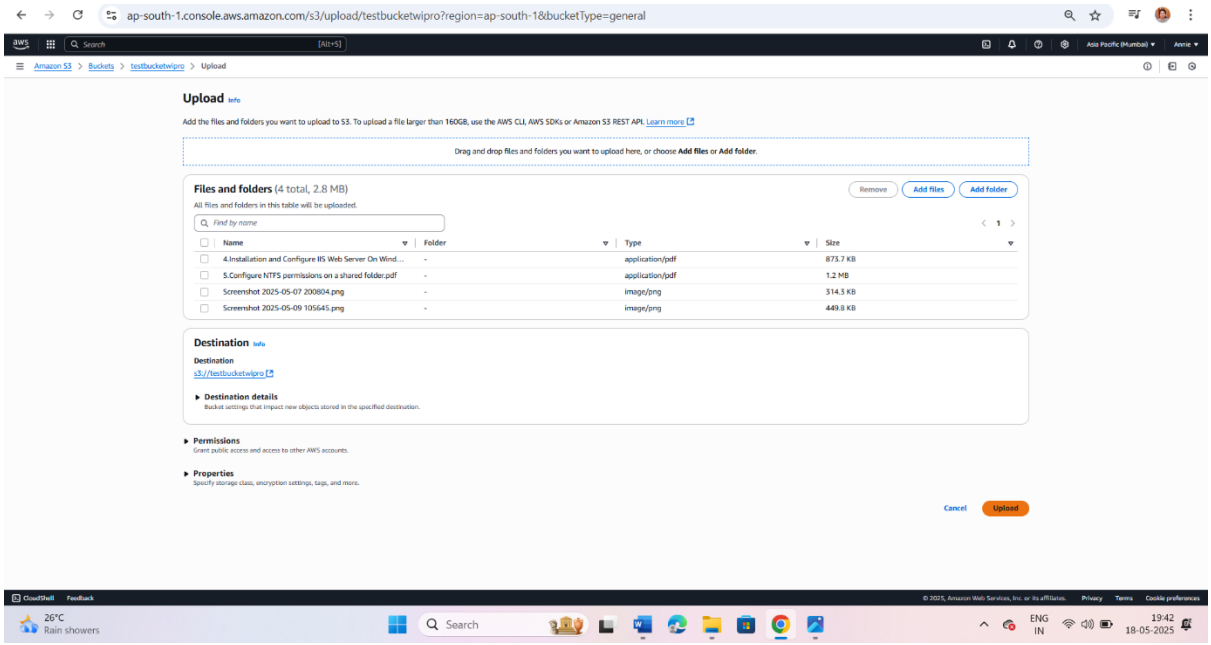
- Give a bucket name should be global one->change object ownership to ACL enabled->uncheck the block public access ->create bucket.



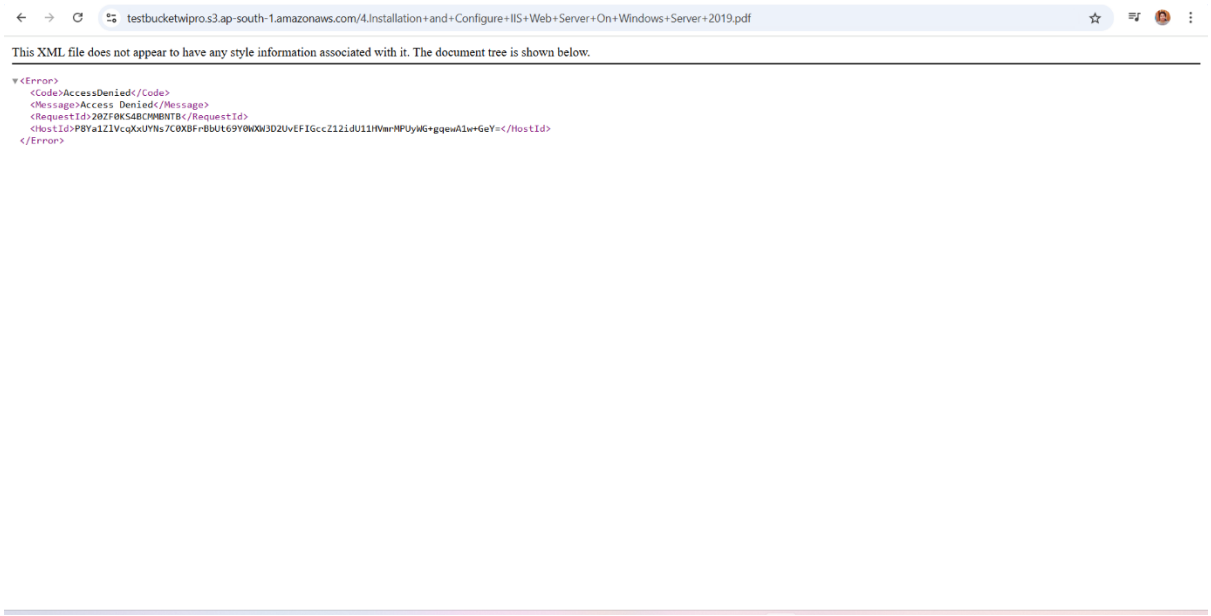
- Created bucket, it was created on location Mumbai but I changed AWS region to Virginia.



- Click on bucket name->upload some documents.

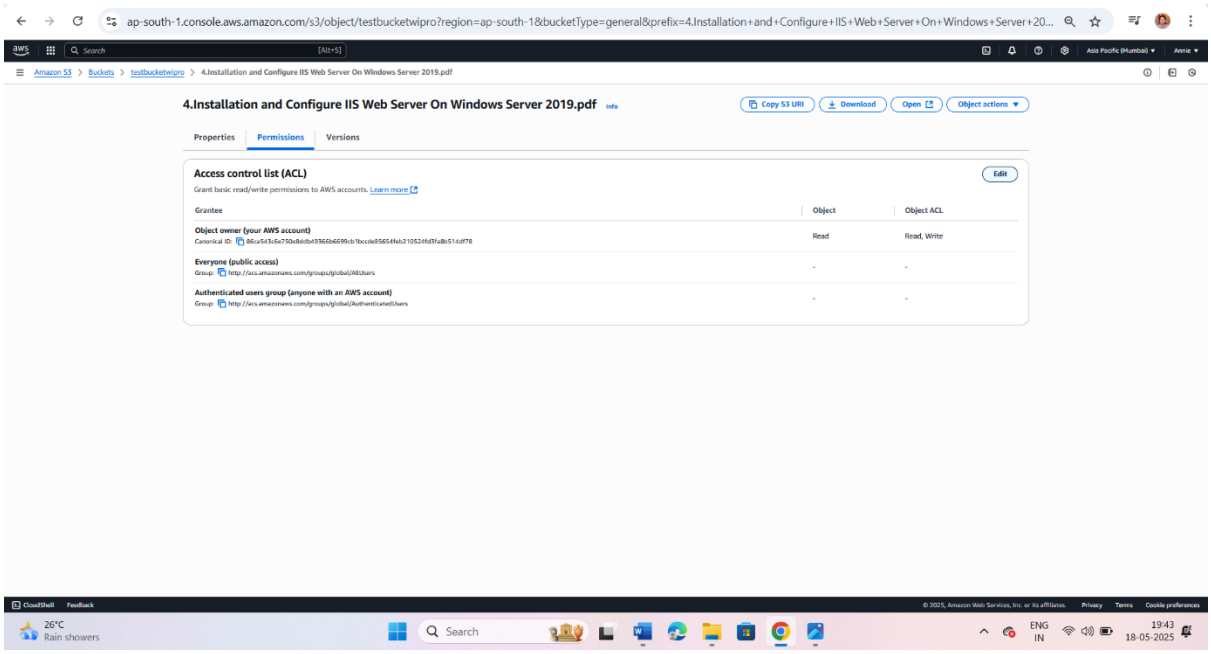


- Copy the object URL of one document and open it from browser we will get the error.

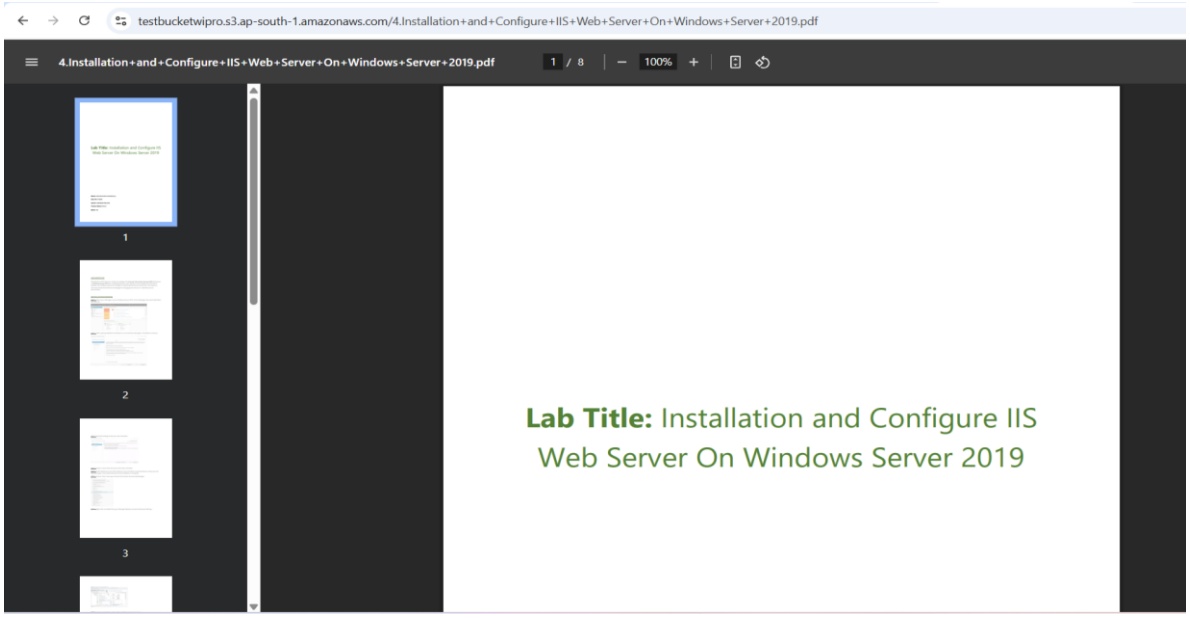


7. Giving file permissions

i. So, the files don’t have permission for public we give permission **individually** by clicking any one of file ->click the permission give object permission for read and save changes.

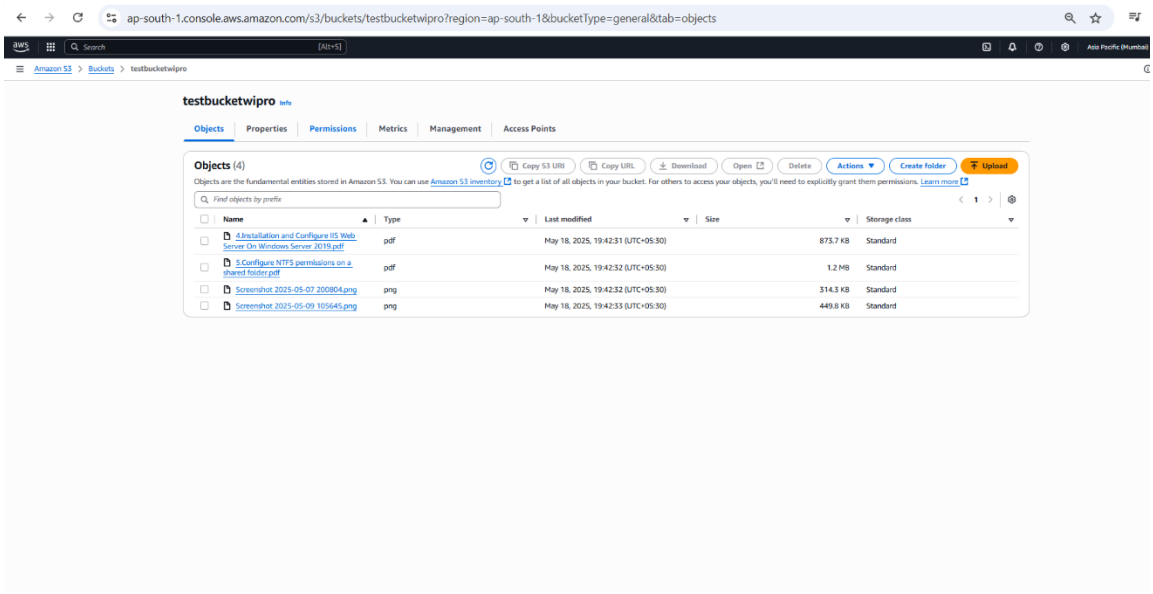


- Again, refresh the browser, the file will be visible.

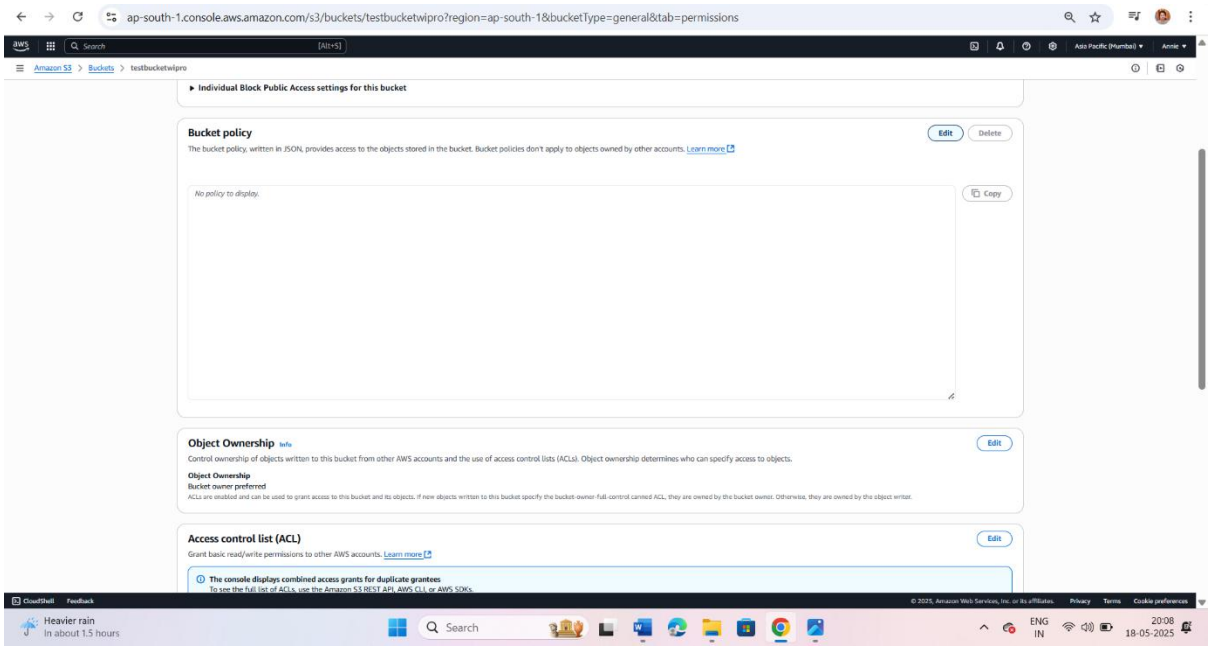


ii. File permission using policy in Jason format(This JSON policy means: "**Allow everyone in the world to read (download) any file inside this S3 bucket.**")

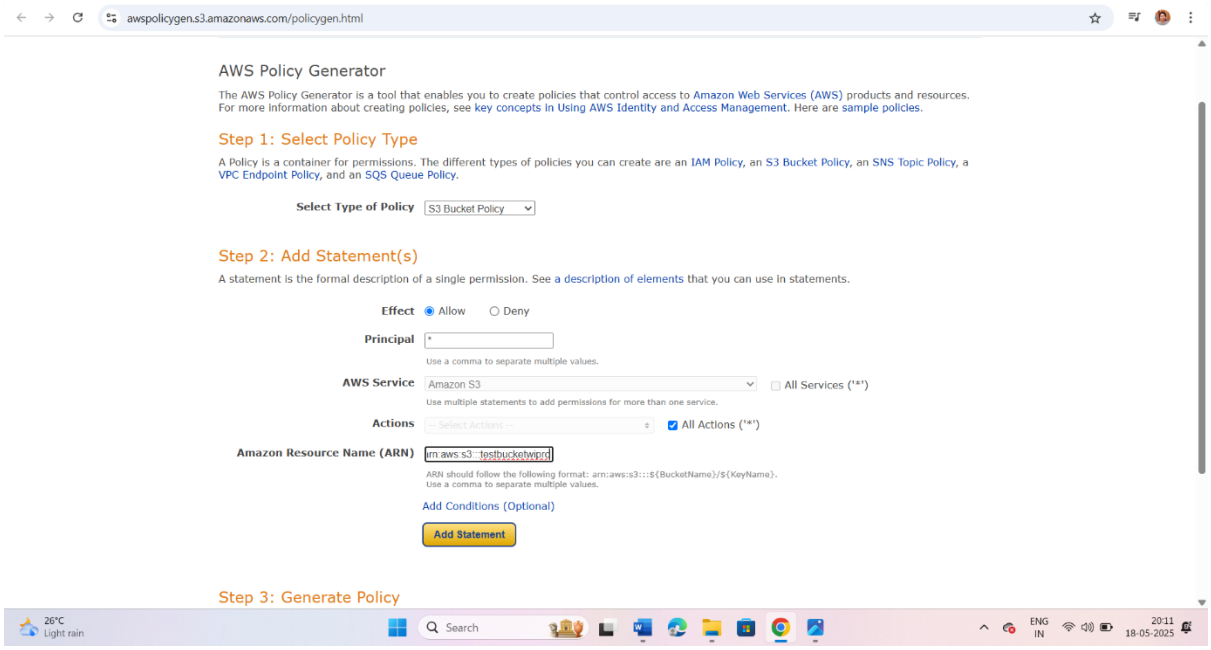
- Go to permissions in bucket.



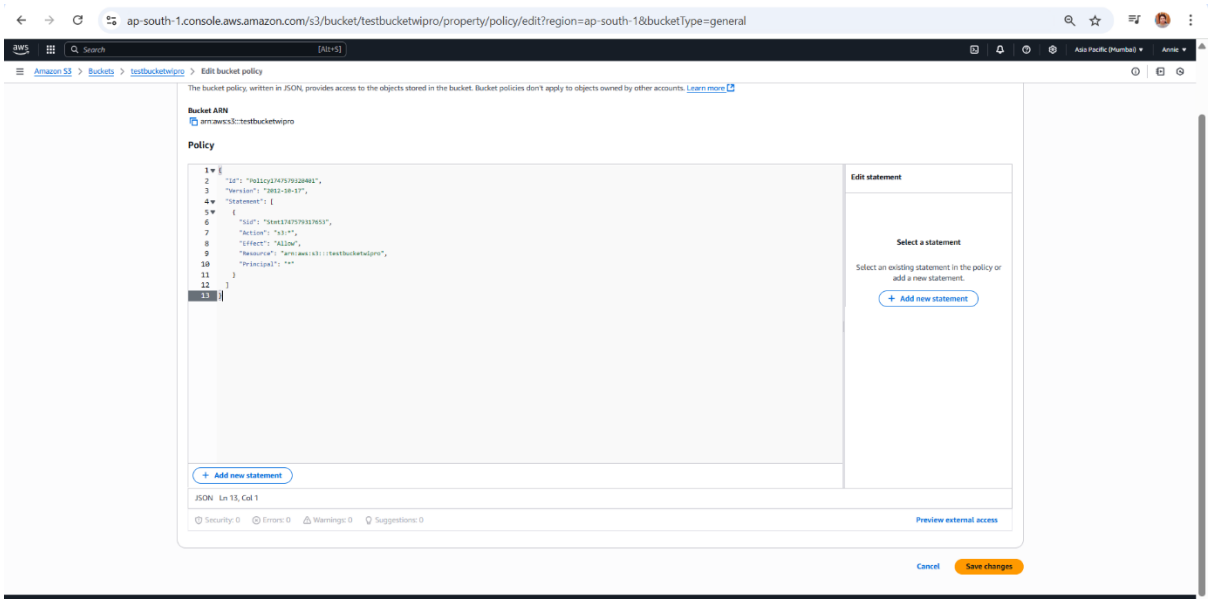
- See the bucket policy is empty.



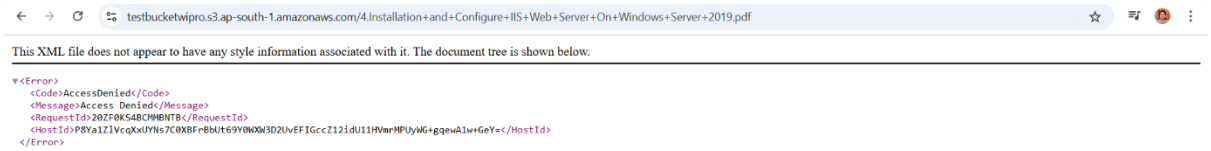
- Go to policy generator->Select policy type as **S3 bucket policy** ->Effect allow->principal as *(everyone)->Select checkbox all actions ->paste the ARN(A **Bucket ARN** (Amazon Resource Name) uniquely identifies an **Amazon S3 bucket**) copied from edit bucket policy->add statement->generate policy->copy it.

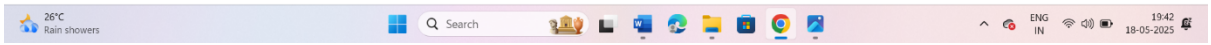


- Paste it over edit policy->save.



- Open a file and check again get an error.





- Again, in edit policy add a change then only public access is allowed for all files(in resources ARN give **/***).