Create a VPC in AWS with address as 172.20.0.0. Create a public subnet and a private subnet. Connect Internet Gateway to the public subnet. Create an EC2 instance connect it to the public subnet. Create another EC2 instance and connect it to private subnet. Display that you are able to connect using putty to the EC2 instance in the public subnet.

Copy the key of the second EC2 instance to the EC2 instance in the public subnet. Display that you are able to ssh to the EC2 instance in the private subnet from the EC2 instance in the public subnet.

1 **create vpc**

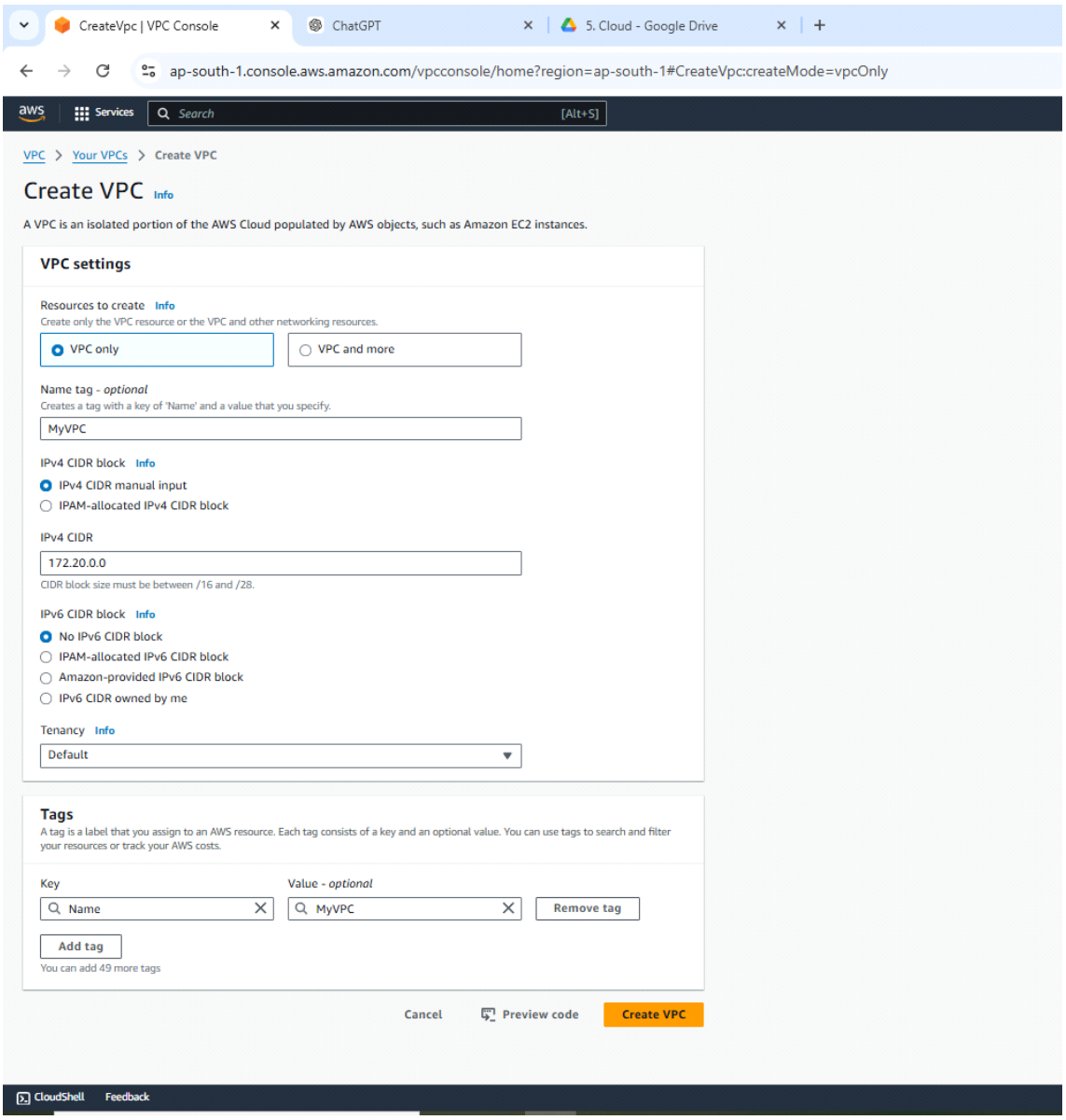
In the "Create VPC" wizard:

Name tag: "MyVPC"

IPv4 CIDR block: 172.20.0.0/16

Leave other options as default (IPv6, Tenancy).

Click Create VPC.



Step 2: Create Public and Private Subnets

After creating the VPC, go to the Subnets section.

Create the Public Subnet:

Click Create subnet.

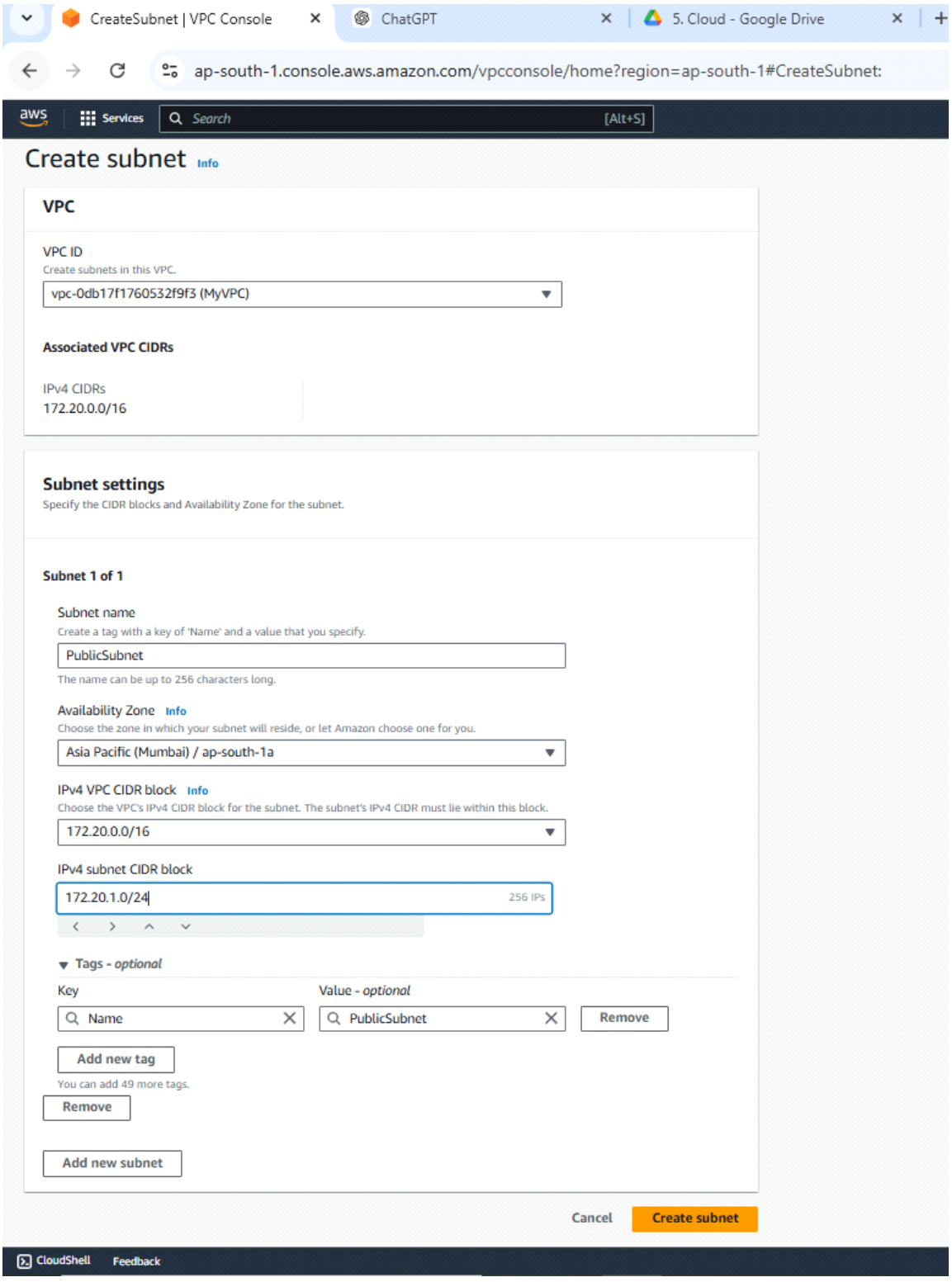
Select the VPC created in Step 1.

Name tag: "PublicSubnet"

Availability Zone: Choose one (e.g., us-east-1a).

CIDR block: 172.20.1.0/24

Click Create subnet.


CreateSubnet | VPC Console
ChatGPT
5. Cloud - Google Drive
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateSubnet:
aws
Services
Search
[Alt+S]
Create subnet Info
VPC
VPC ID
Create subnets in this VPC.
vpc-0db17f1760532f9f3 (MyVPC)
Associated VPC CIDRs
IPv4 CIDRs
172.20.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 1
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PublicSubnet
The name can be up to 256 characters long.
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
Asia Pacific (Mumbai) / ap-south-1a
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
172.20.0.0/16
IPv4 subnet CIDR block
172.20.1.0/24
256 IPs
Tags - optional
Key
Value - optional
Name
PublicSubnet
Remove
Add new tag
You can add 49 more tags.
Remove
Add new subnet
Cancel
Create subnet
CloudShell
Feedback

3 **Create the Private Subnet:**

Click Create subnet again.

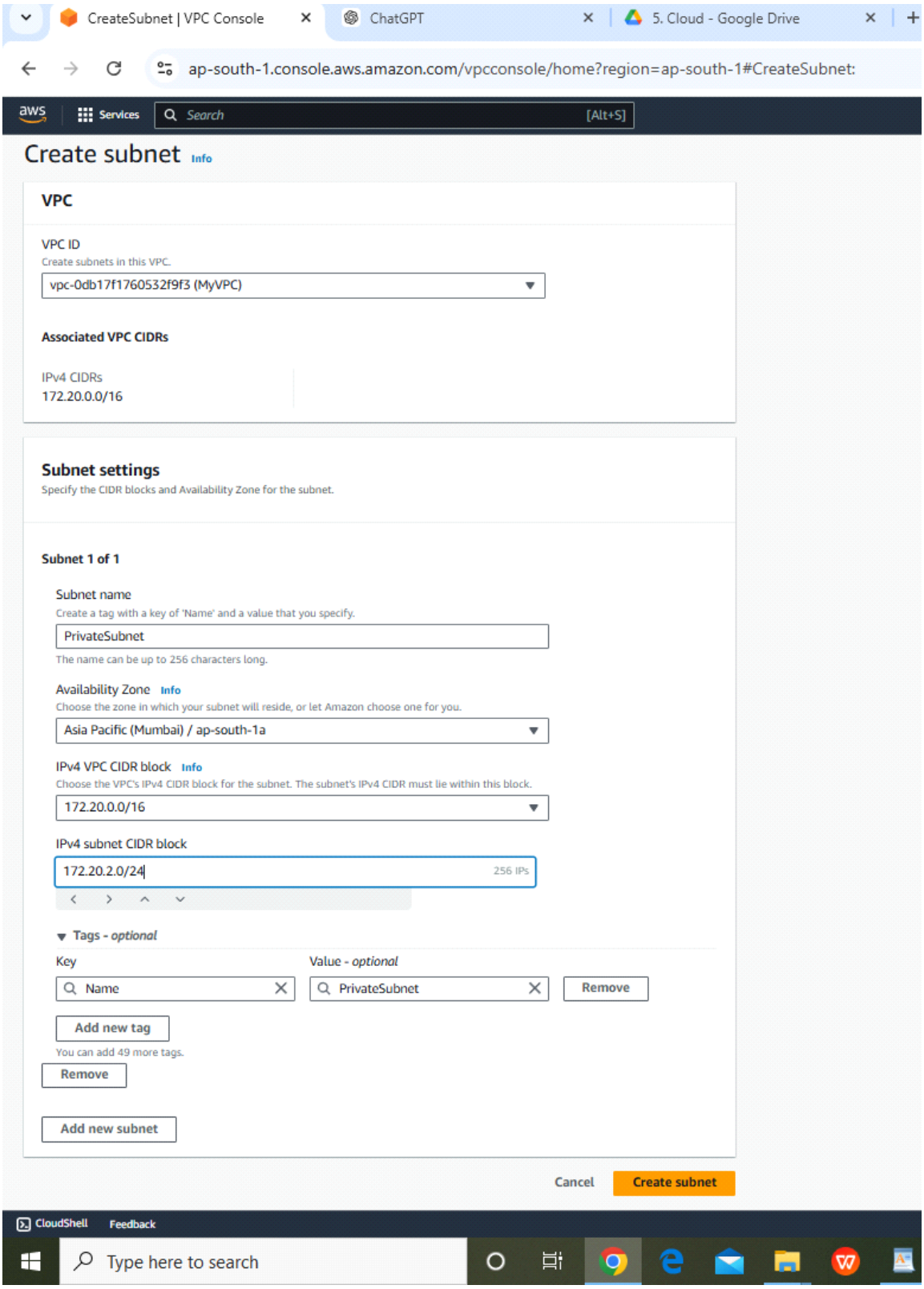
Select the same VPC.

Name tag: "PrivateSubnet"

Availability Zone: Choose the same or different (e.g., us-east-1b).

CIDR block: 172.20.2.0/24

Click Create subnet.


CreateSubnet | VPC Console
ChatGPT
5. Cloud - Google Drive
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateSubnet:
aws
Services
Search
[Alt+S]
Create subnet Info
VPC
VPC ID
Create subnets in this VPC.
vpc-0db17f1760532f9f3 (MyVPC)
Associated VPC CIDRs
IPv4 CIDRs
172.20.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 1
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PrivateSubnet
The name can be up to 256 characters long.
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
Asia Pacific (Mumbai) / ap-south-1a
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
172.20.0.0/16
IPv4 subnet CIDR block
172.20.2.0/24
256 IPs
Tags - optional
Key
Value - optional
Name
PrivateSubnet
Remove
Add new tag
You can add 49 more tags.
Remove
Add new subnet
Cancel
Create subnet
CloudShell
Feedback
Type here to search

Step 3: Create an Internet Gateway (IGW)

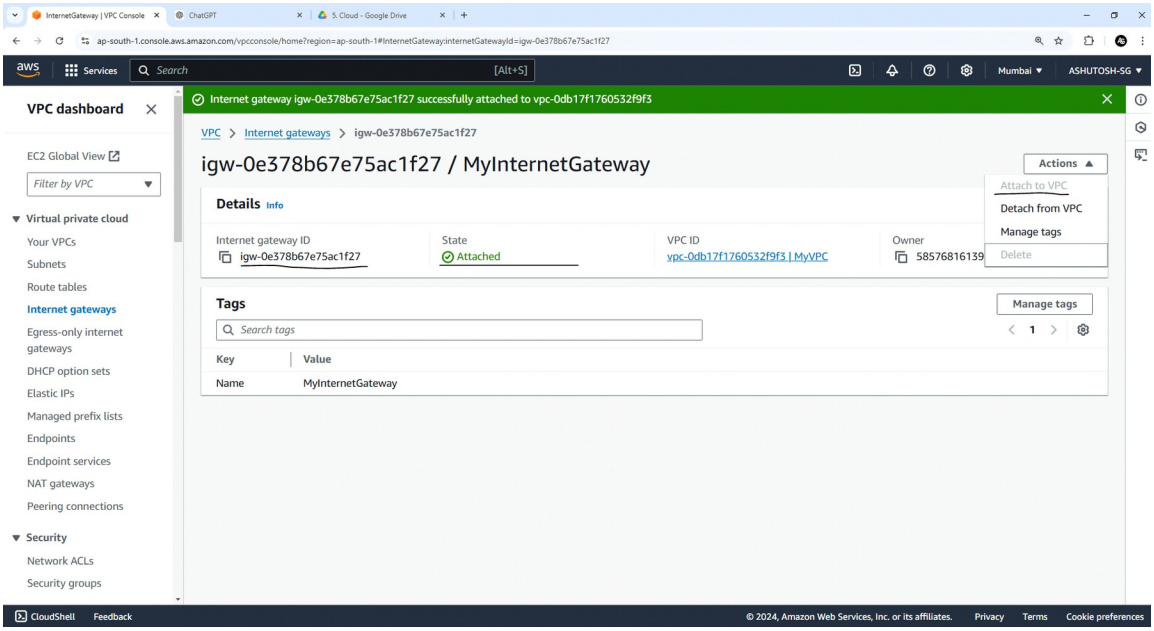
Go to the Internet Gateways section in the VPC Dashboard.

Click Create internet gateway.

Name tag: "MyInternetGateway"

Click Create and then Attach to VPC.

Select the VPC ("MyVPC") and click Attach.



Step 4: Update Route Tables for Public and Private Subnets

Configure the Public Subnet Route Table:

Go to the Route Tables section.

Select the route table associated with the PublicSubnet.

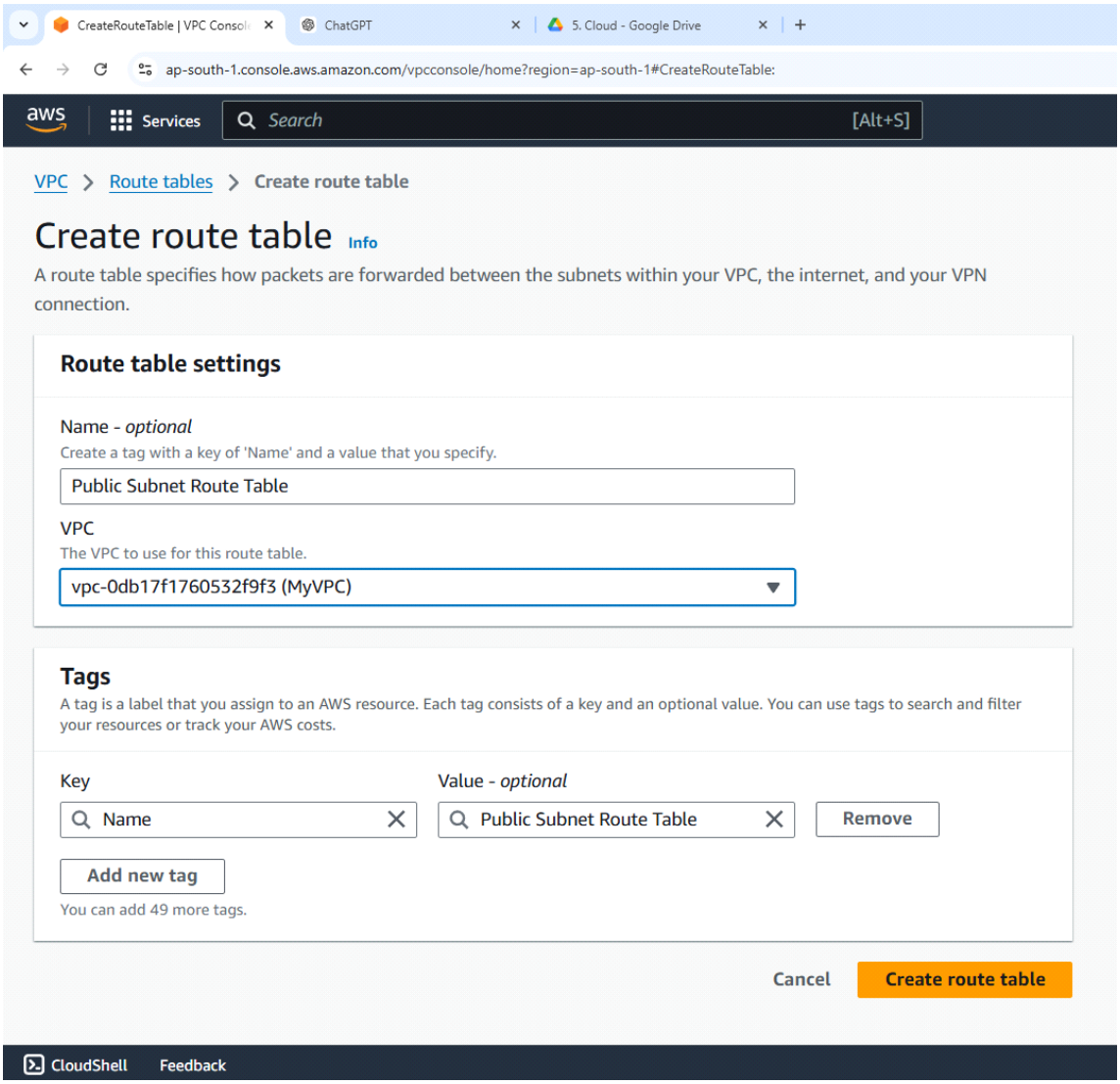
Click on the Routes tab, then Edit routes.

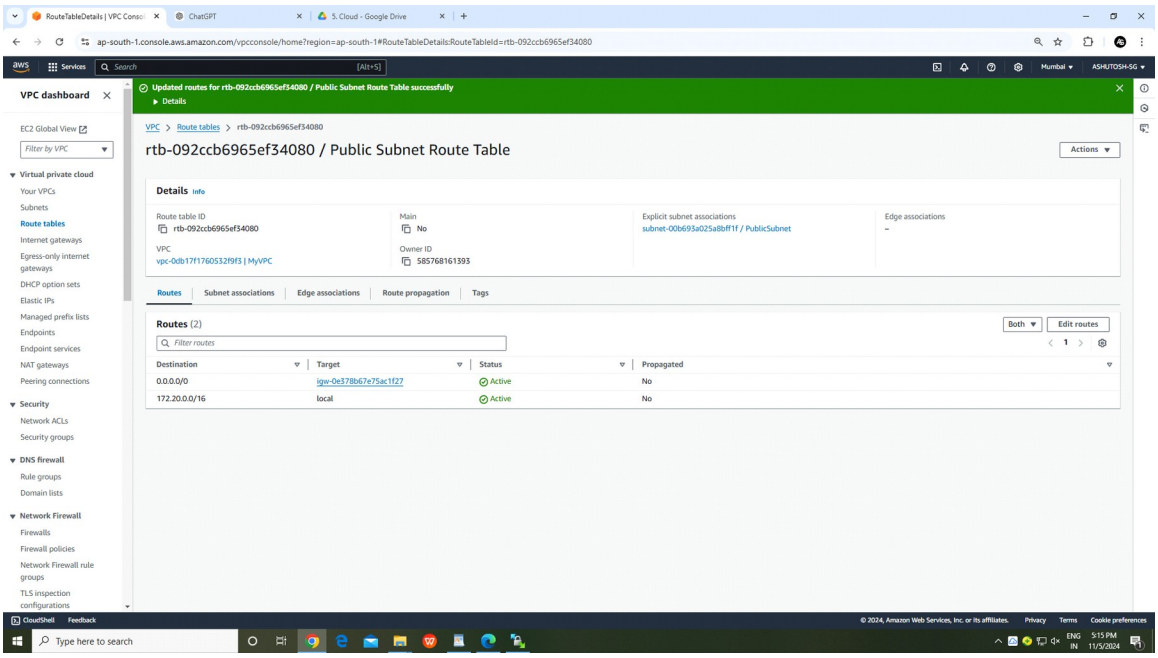
Add a new route:

Destination: 0.0.0.0/0

Target: Select the Internet Gateway created in Step 3.

Click Save routes.





Step 5: Launch EC2 Instance in the Public Subnet

Go to the EC2 Dashboard and click Launch Instance.

Select an Amazon Machine Image (AMI) (e.g., Amazon Linux 2).

Choose an Instance Type (e.g., t2.micro).

In the Network section, select the VPC ("MyVPC") and Subnet ("PublicSubnet").

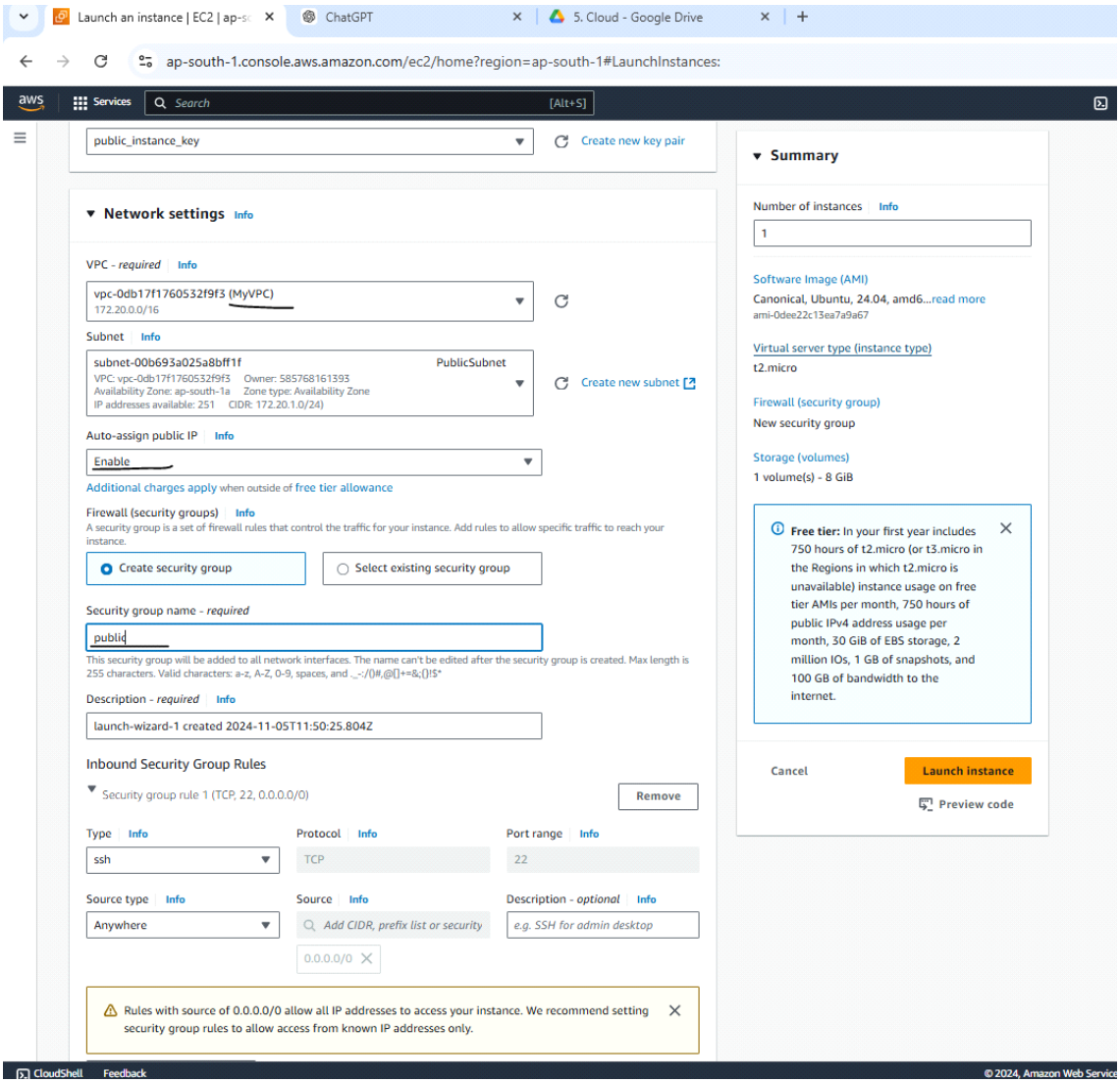
In the Configure Security Group section:

Create a new security group or select an existing one.

Add an inbound rule to allow SSH (port 22) from your IP address.

Key Pair: Create a new key pair **((.pem)**download and save it; you will need it for SSH).

Click Launch Instance.



Step 6: Launch EC2 Instance in the Private Subnet

Follow the same steps as above to create another EC2 instance in the PrivateSubnet.

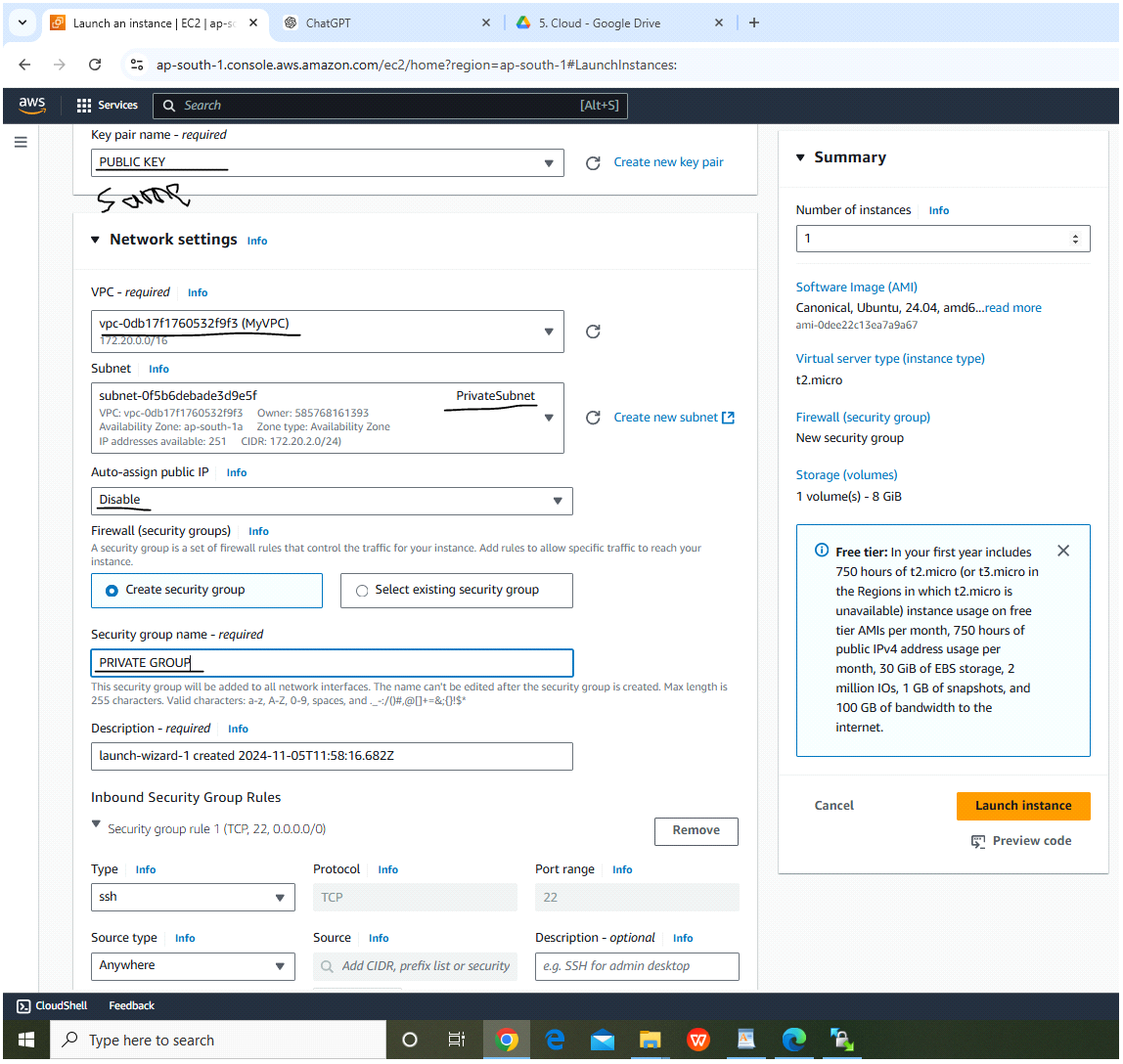
Choose the same AMI (KEY)

EDIT NETWORK SETTINGS

In the Network section, select the VPC ("MyVPC") and Subnet ("PrivateSubnet").

Click Launch Instance.

ALWAYS REMEMBER



Step 7: Connect to the Public EC2 Instance via SSH (using Putty)

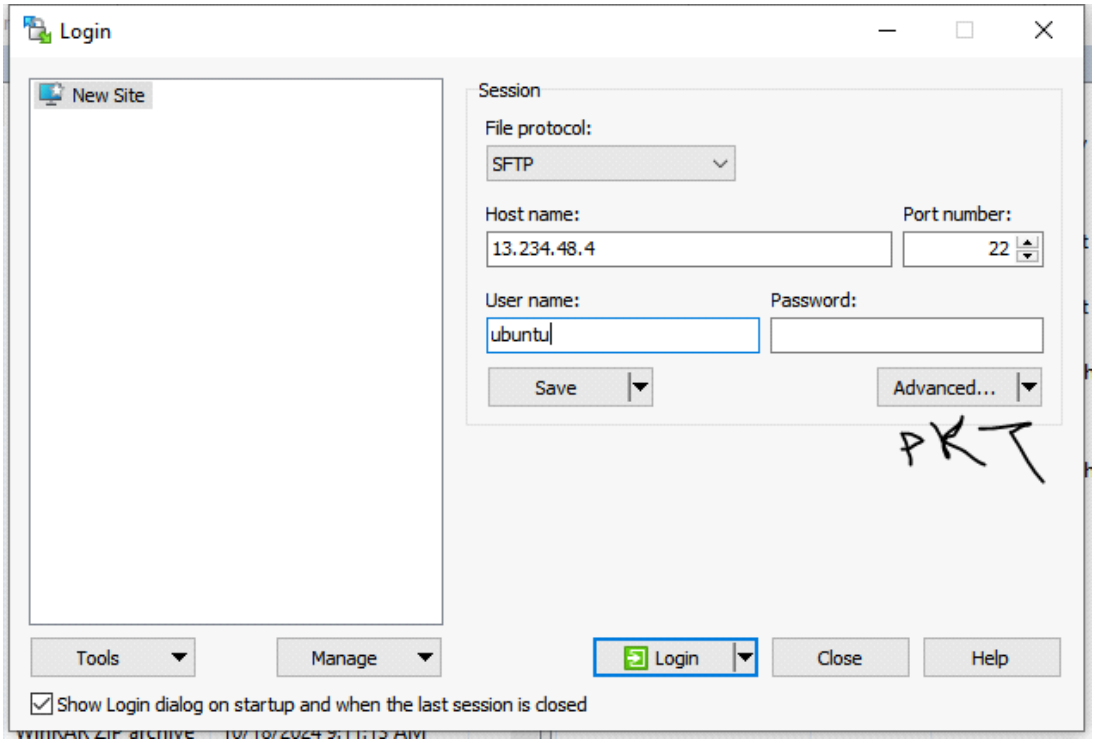
Open Putty (Windows) and enter the Public IP of the public EC2 instance.

Under Connection → SSH → Auth, browse to the private key file (in .pem format) for the public EC2 instance and select it.

Click Open to connect to the EC2 instance. You should be logged into the EC2 instance in the Public Subnet.

- use **putty gen** software to convert .pem to. ppk file

  by **using winscp** software copy pem file to local to ec2 instance



- move pem file to the public instance BY DRAG AND DROP

follow below commands:-

ubuntu@ip-172-20-1-231:~$ ls

'PUBLIC_KEY.pem'

ubuntu@ip-172-20-1-231:~$ chmod 400 PUBLIC_KEY.pem

ubuntu@ip-172-20-1-231:~$ ssh -i "PUBLIC_KEY.pem" 172.20.2.223

Finally we connected the private subnet instance from a public subnet instance (that is also called **Bastion host**)