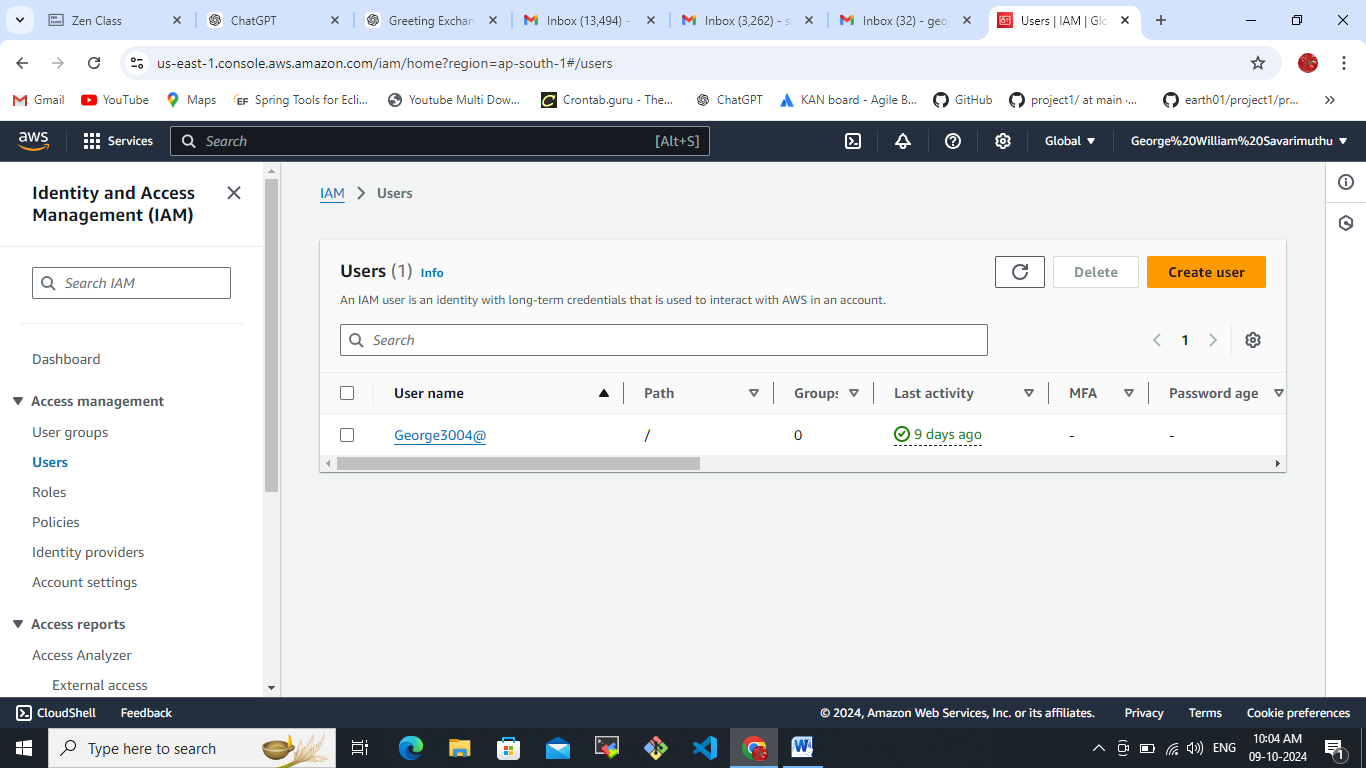
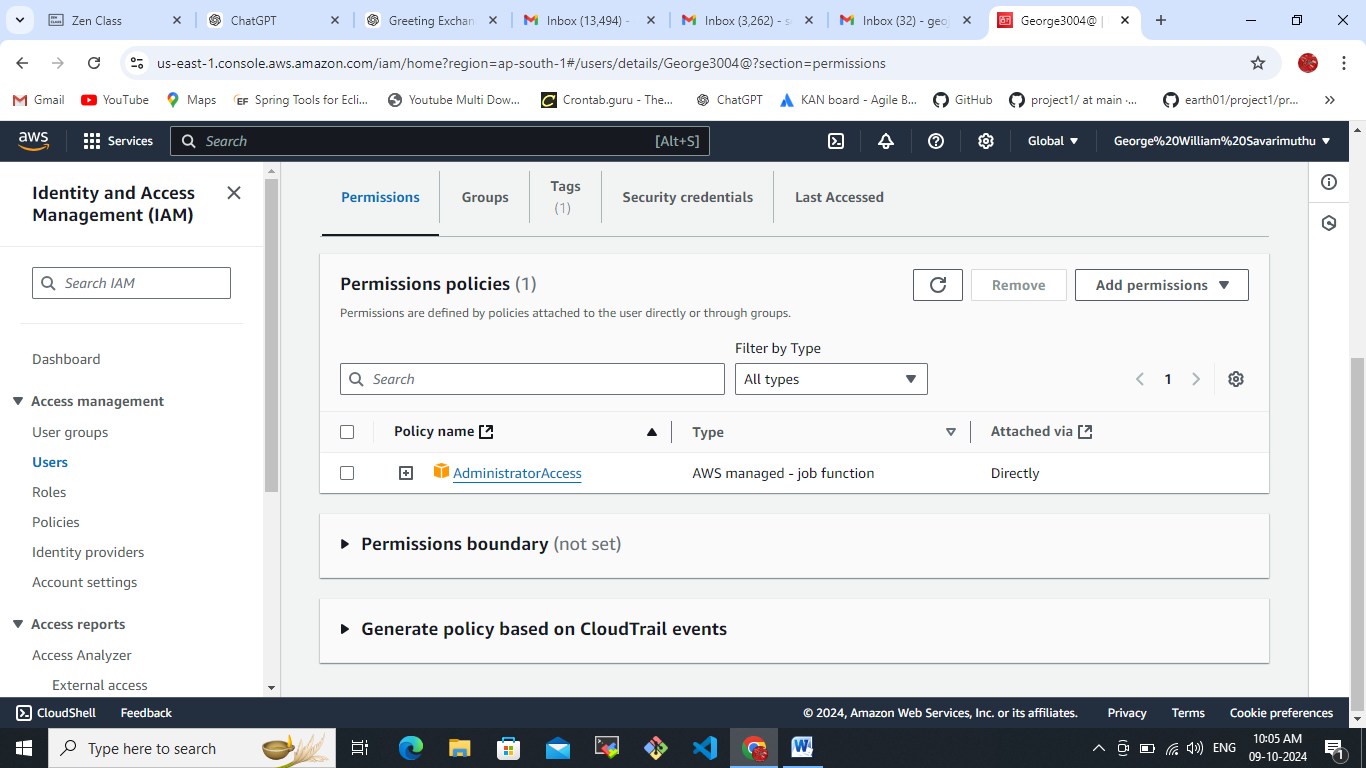
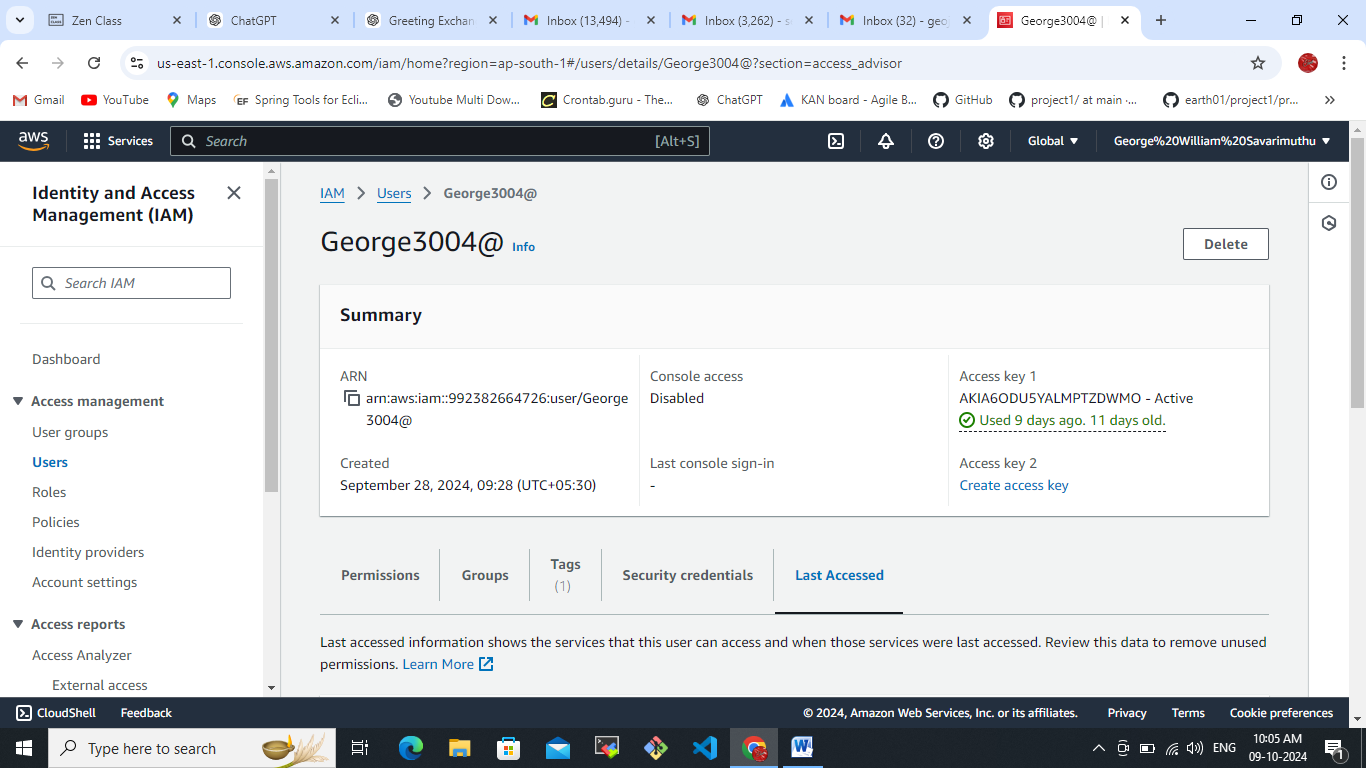
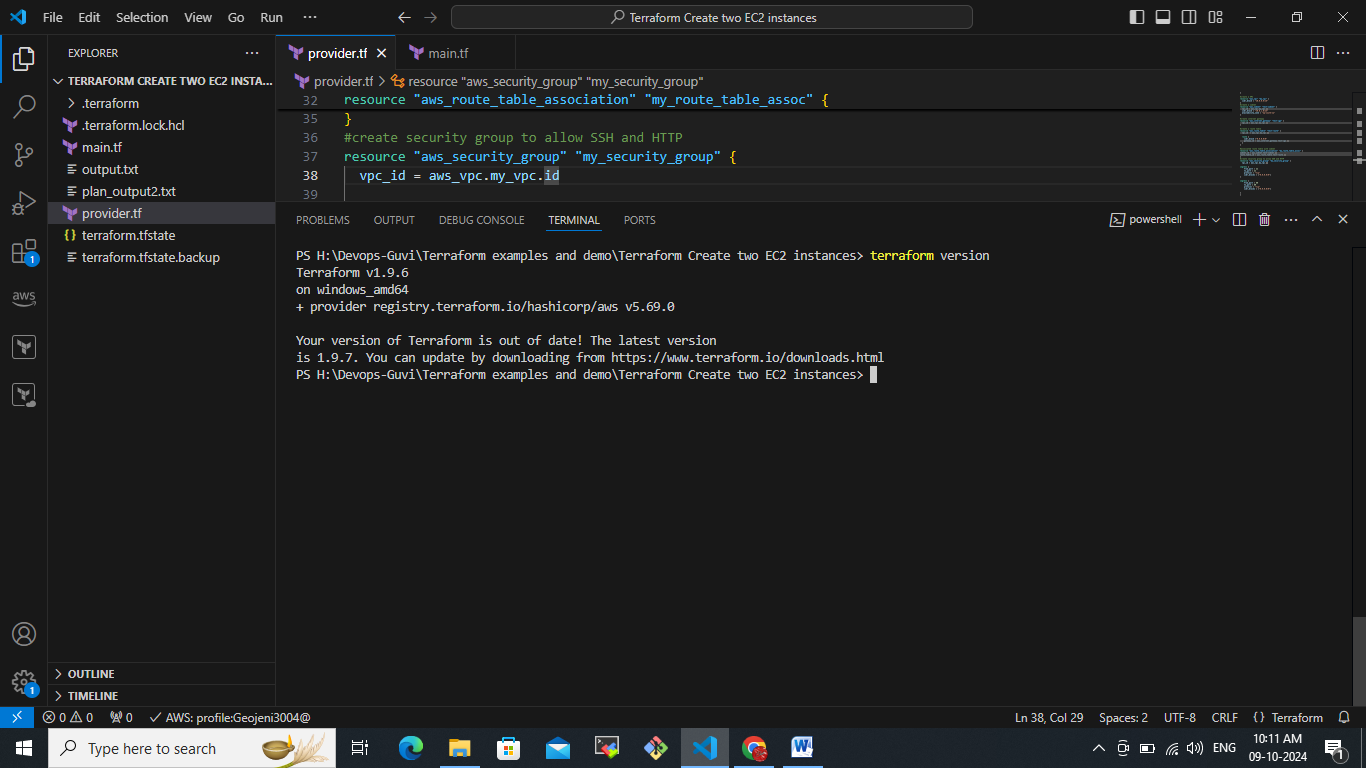
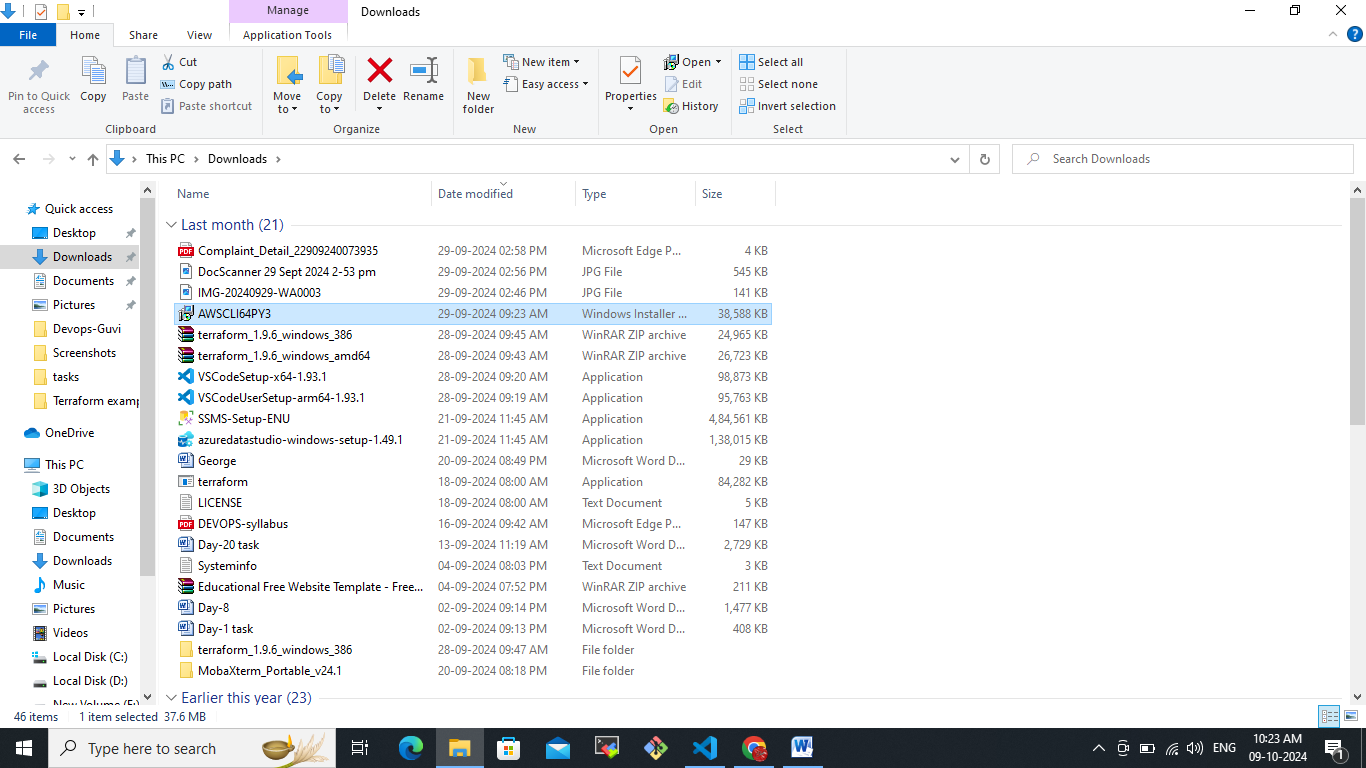
Task 24  
Launch Linux EC2 instances in two regions using a single Terraform file

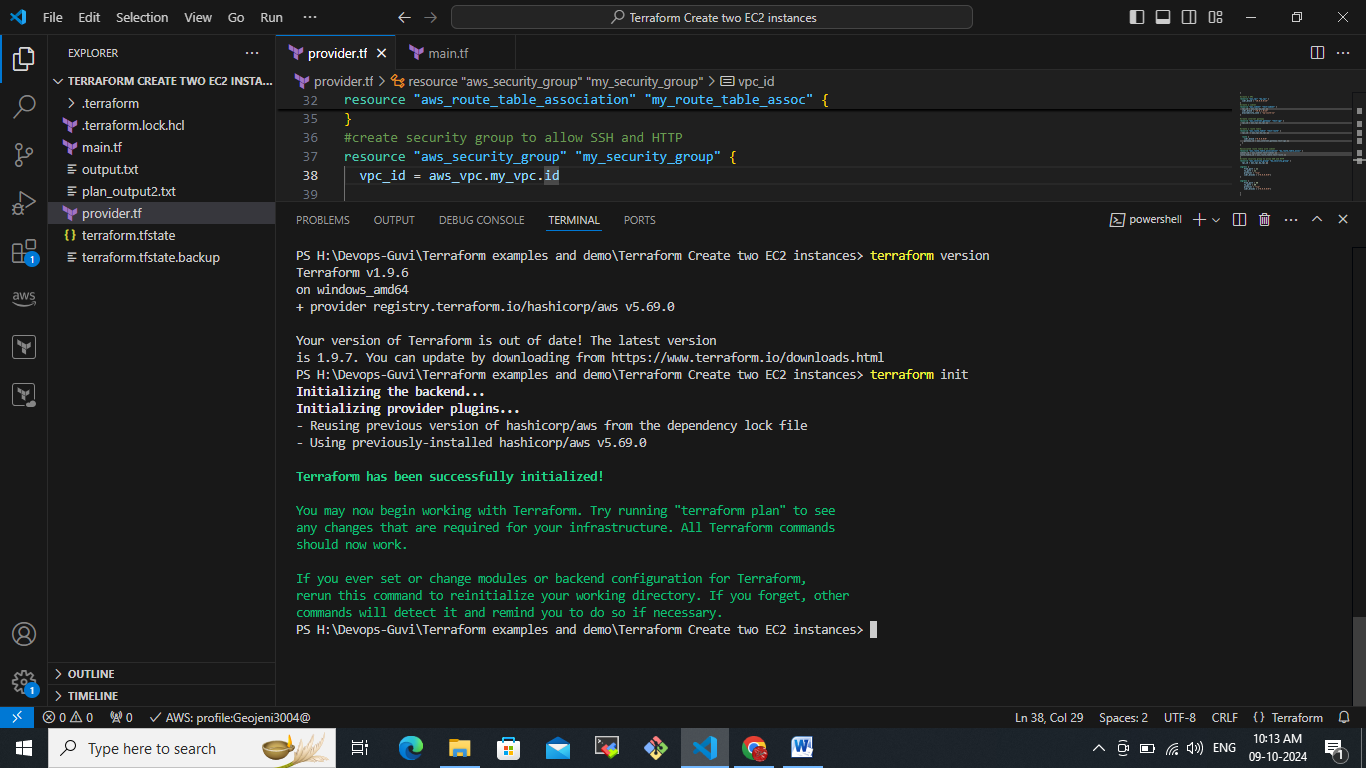
First step create an IAM user with permissions:  


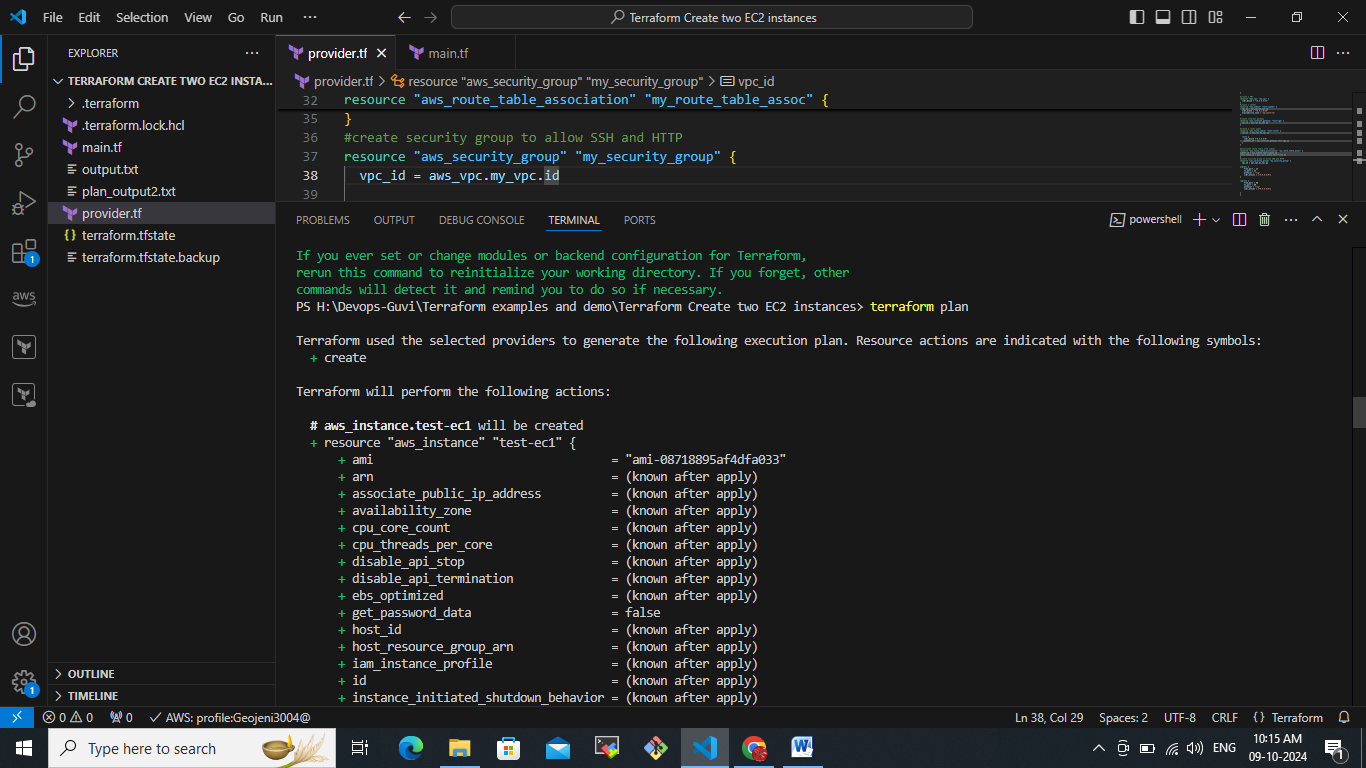
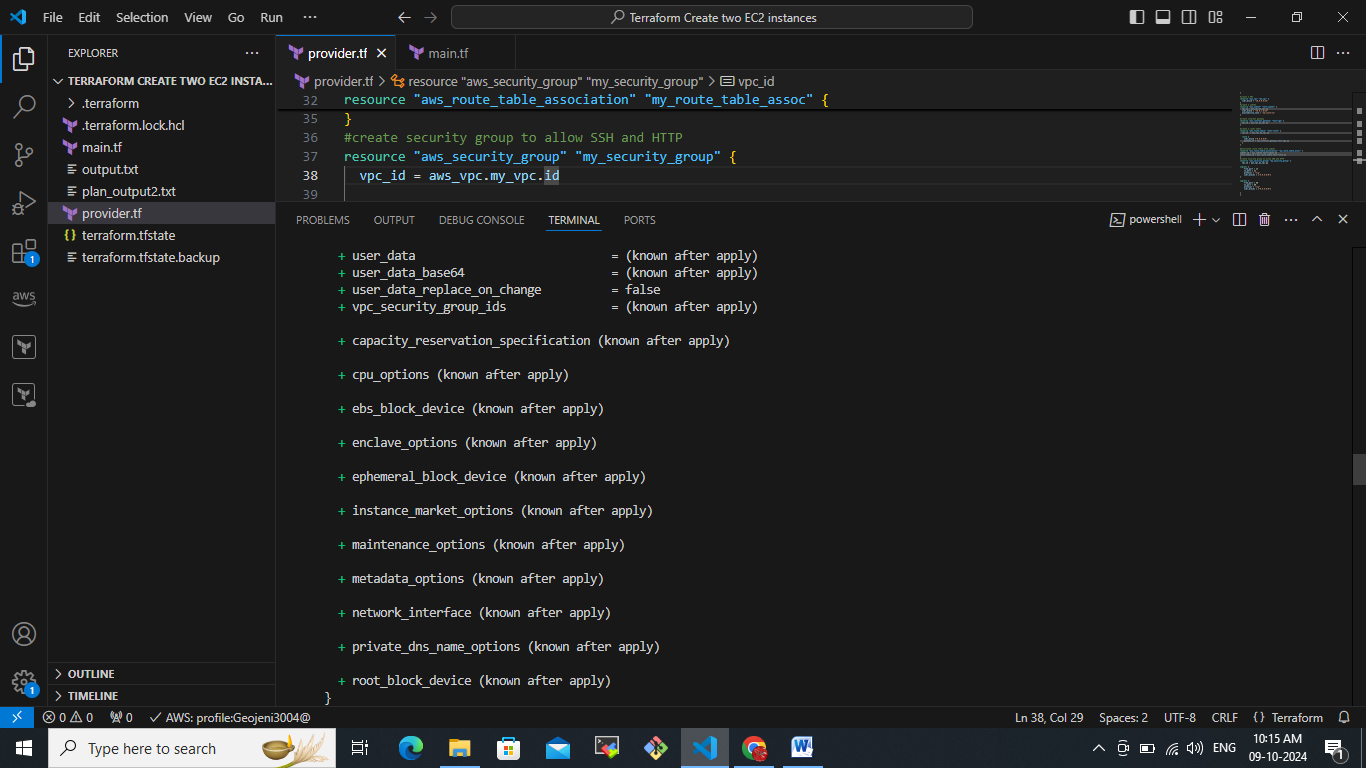
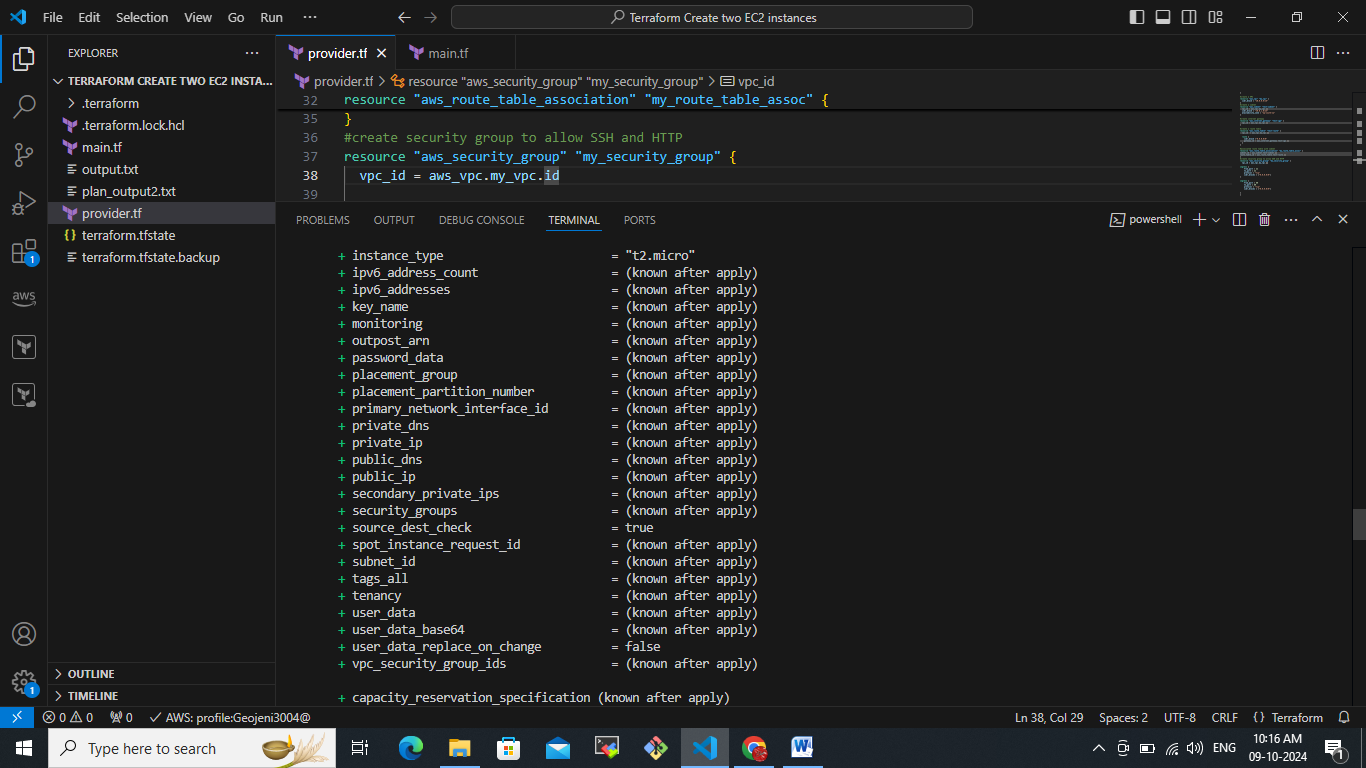
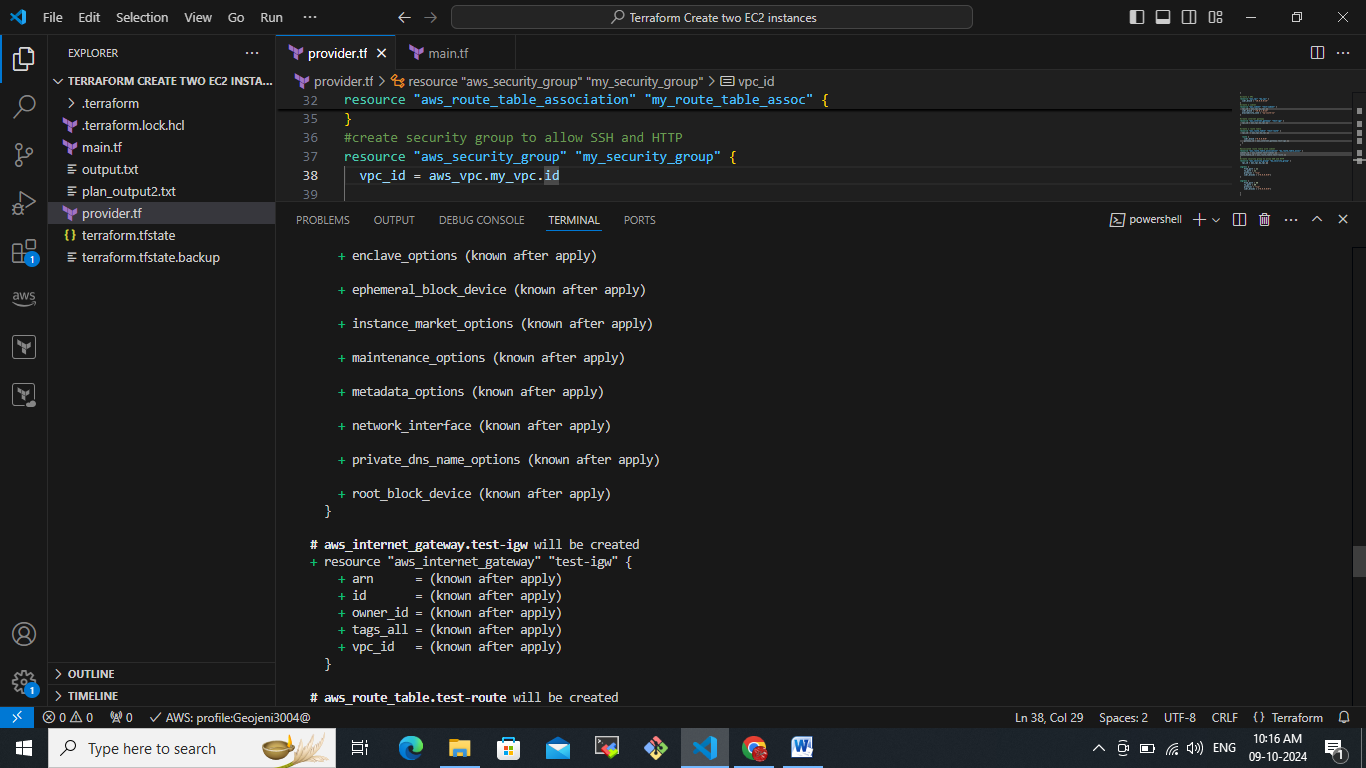
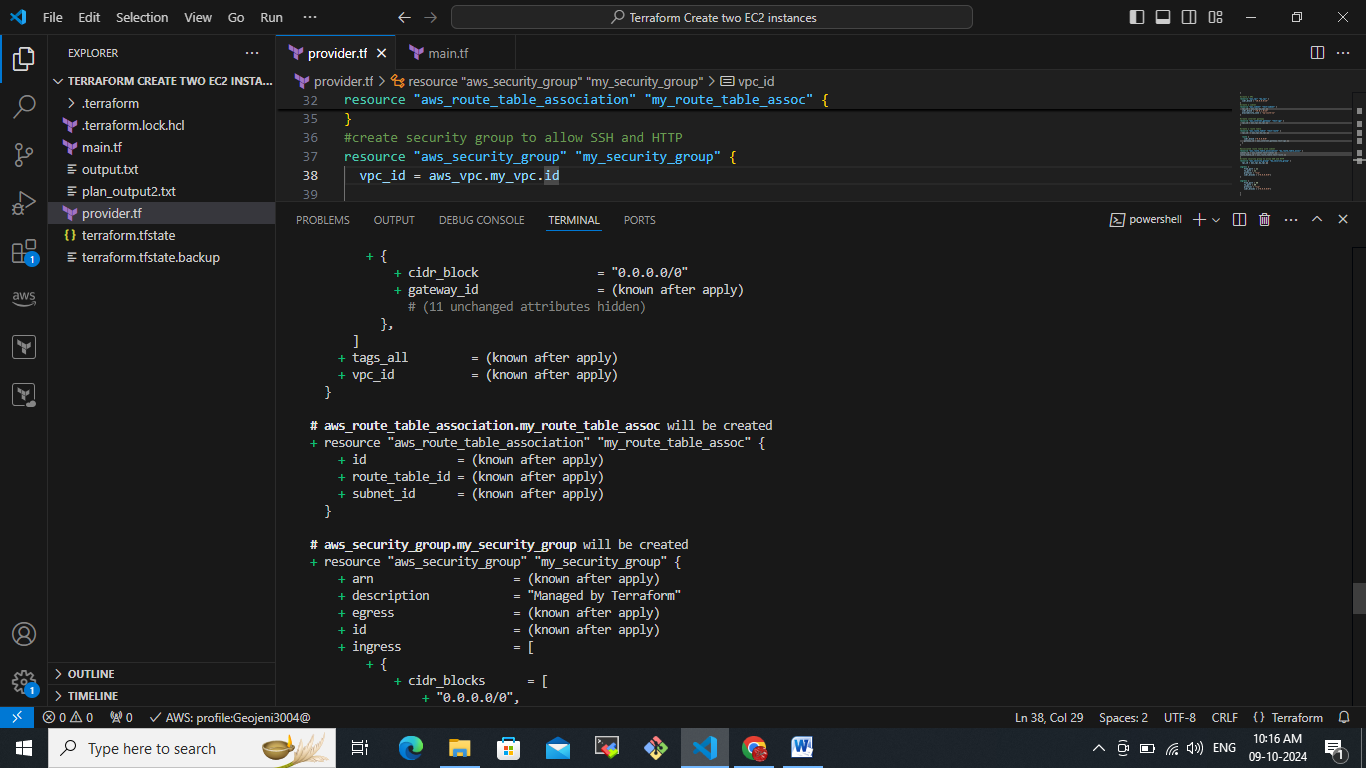
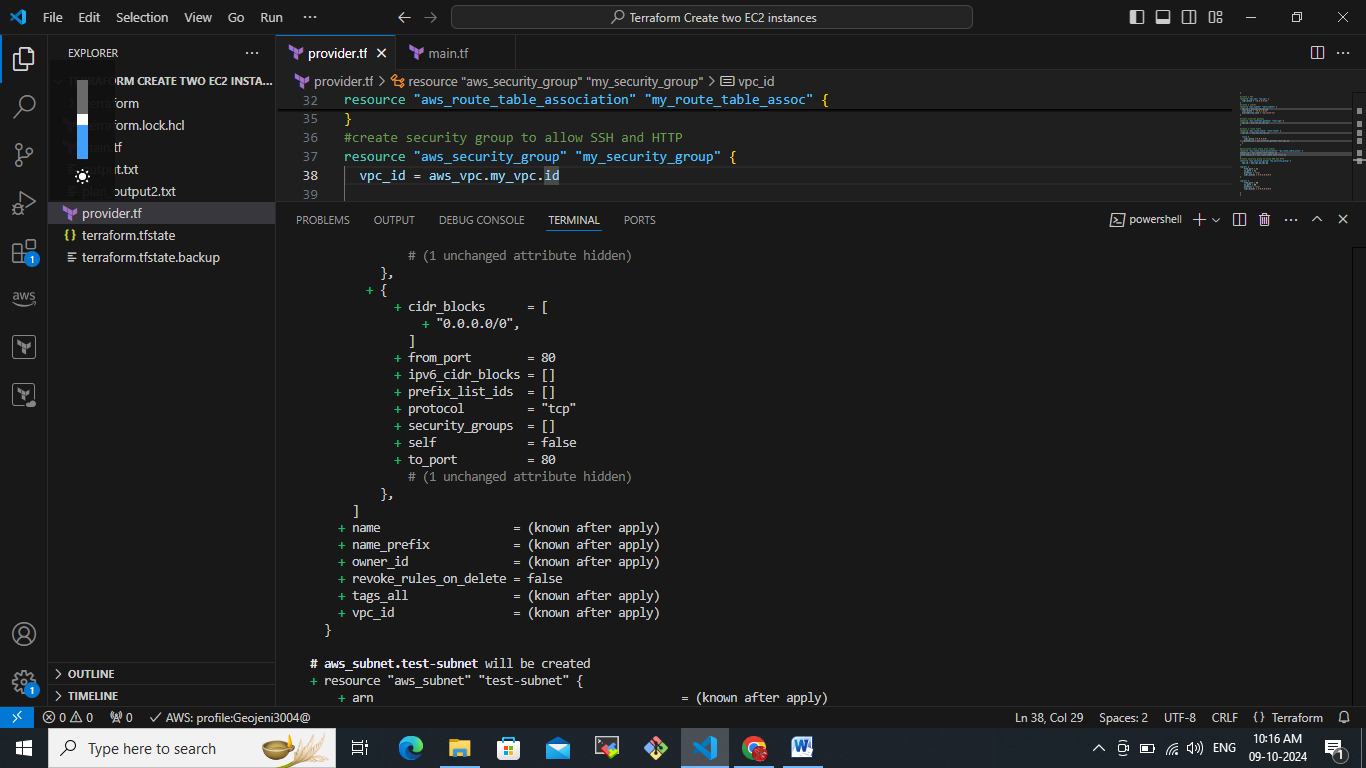
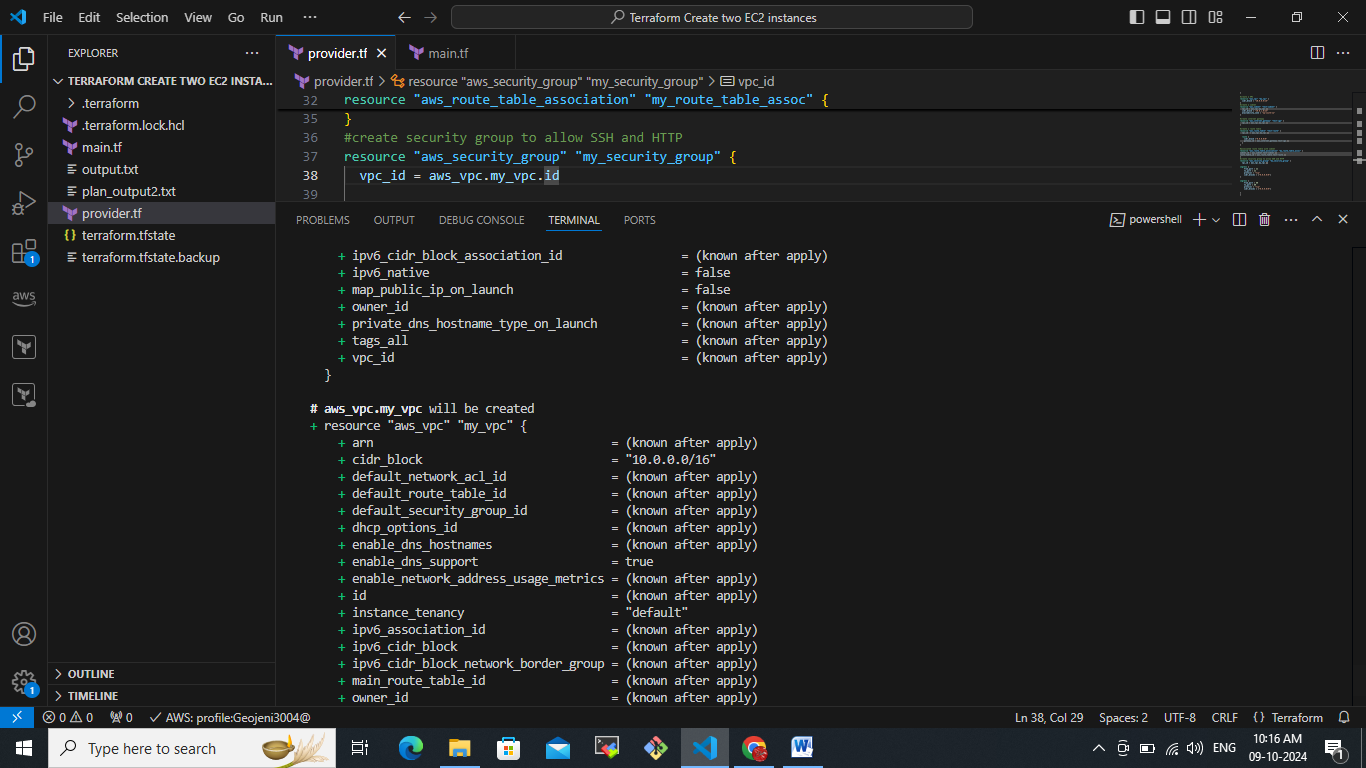
  
Created Access keys:  


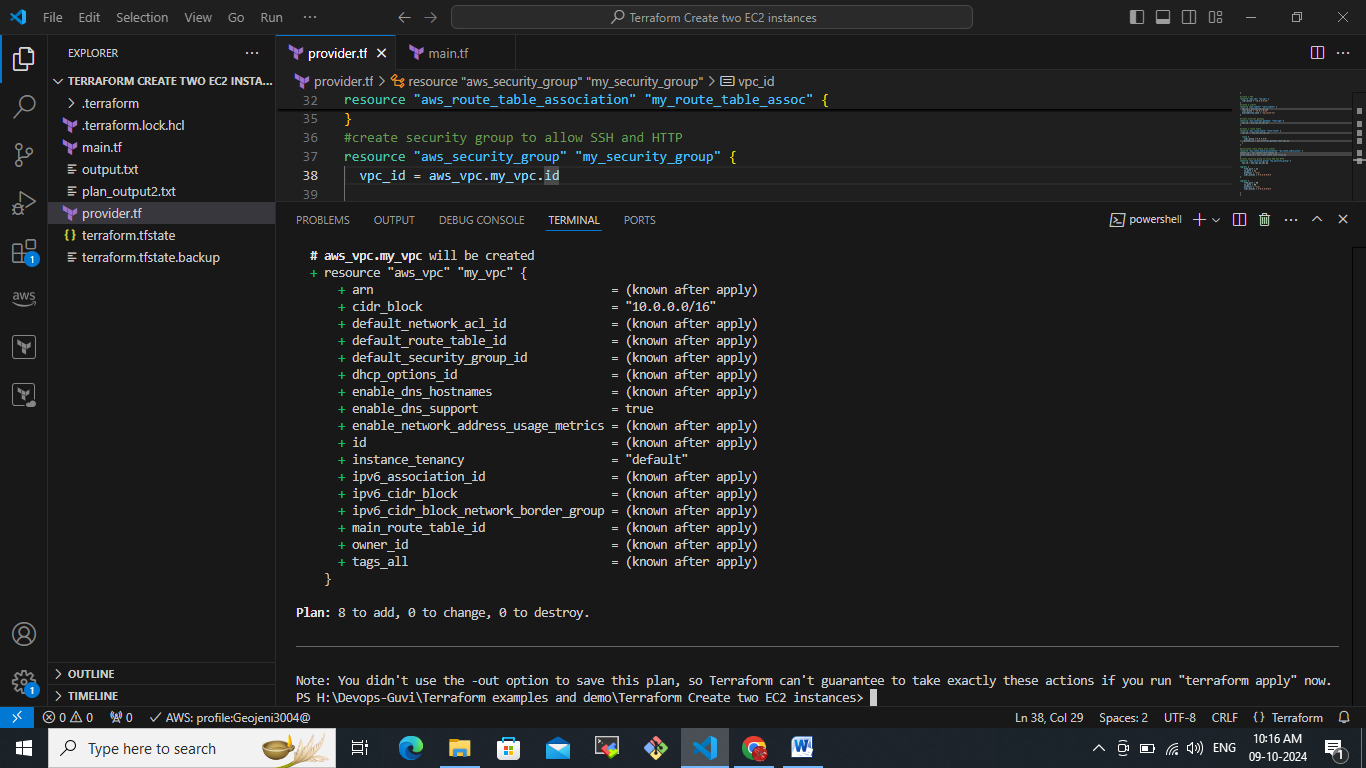
Need to check whether terraform install on your machine.  


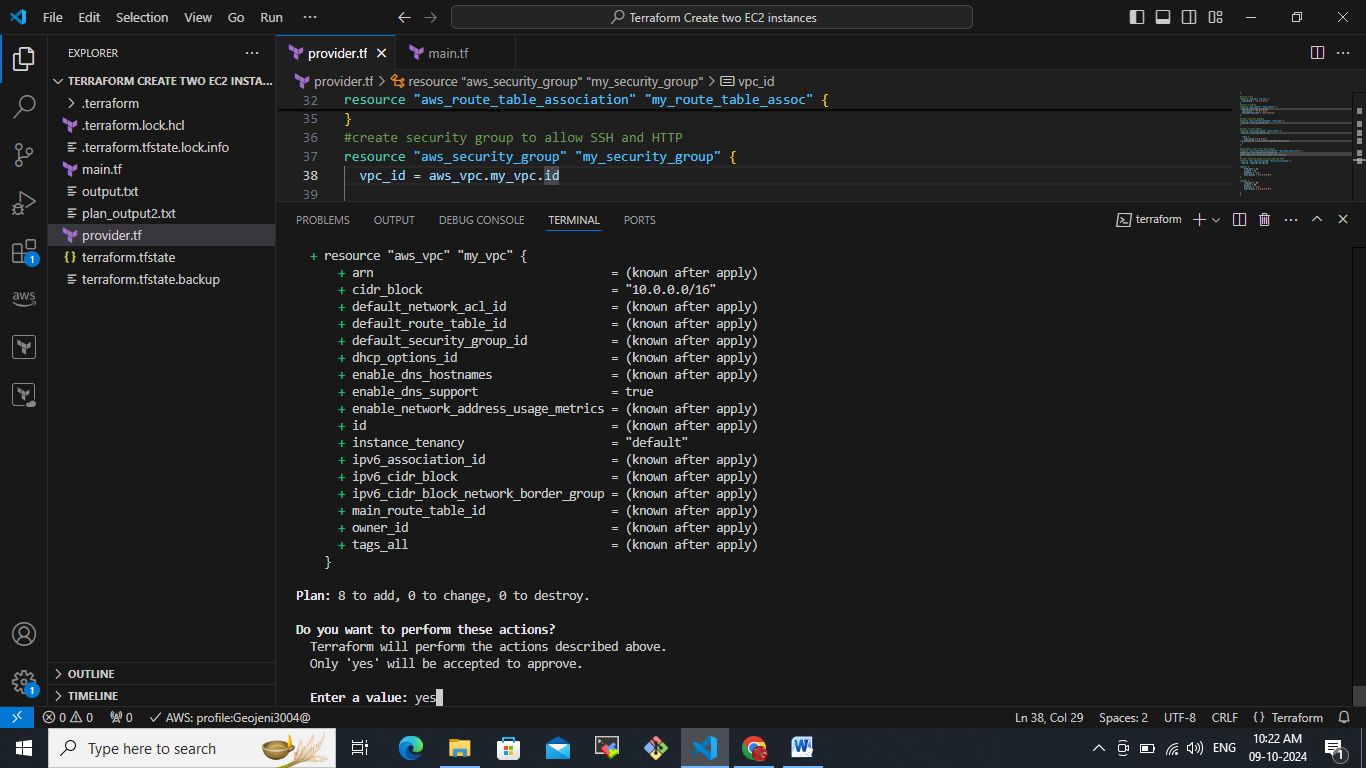
We have configure this create 2 ec2 instance by using the Amazon (CLI- command Line interface). I already downloaded and installed it in my machine.

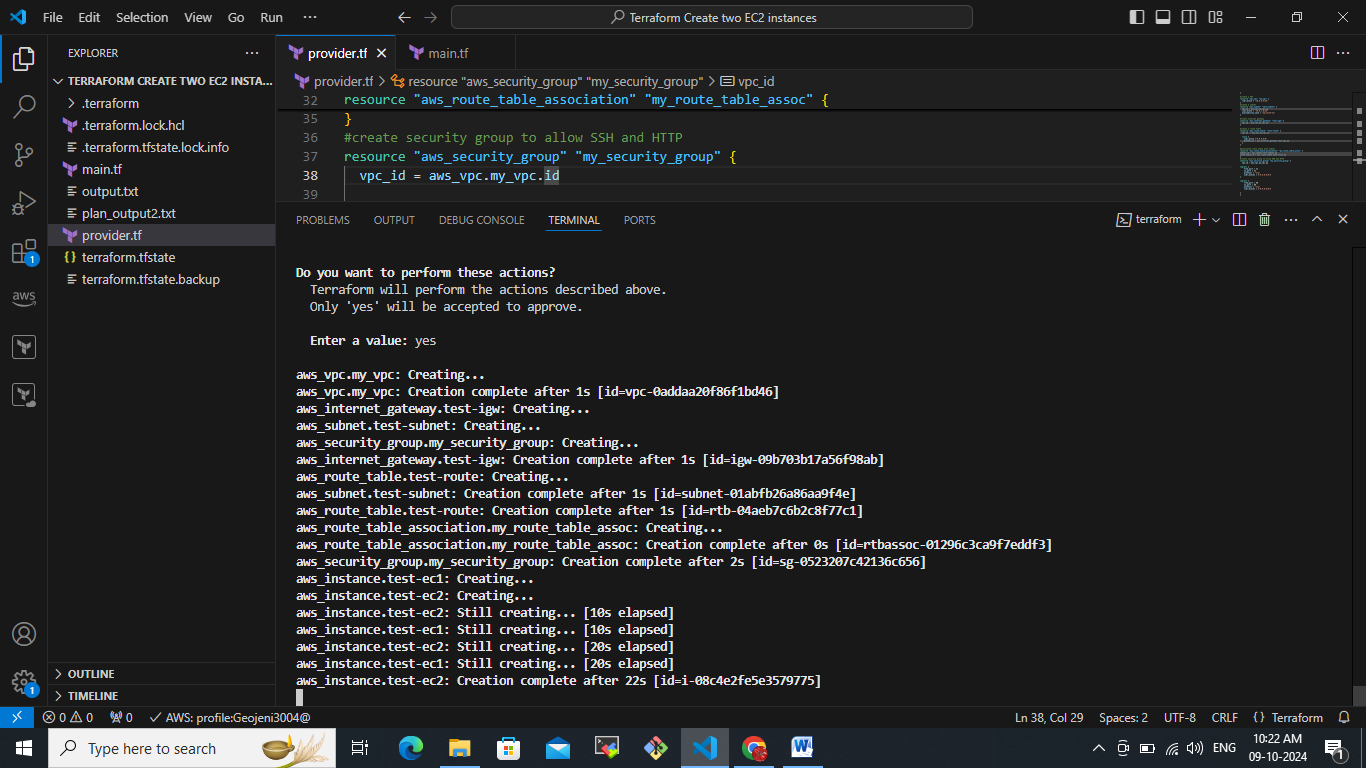


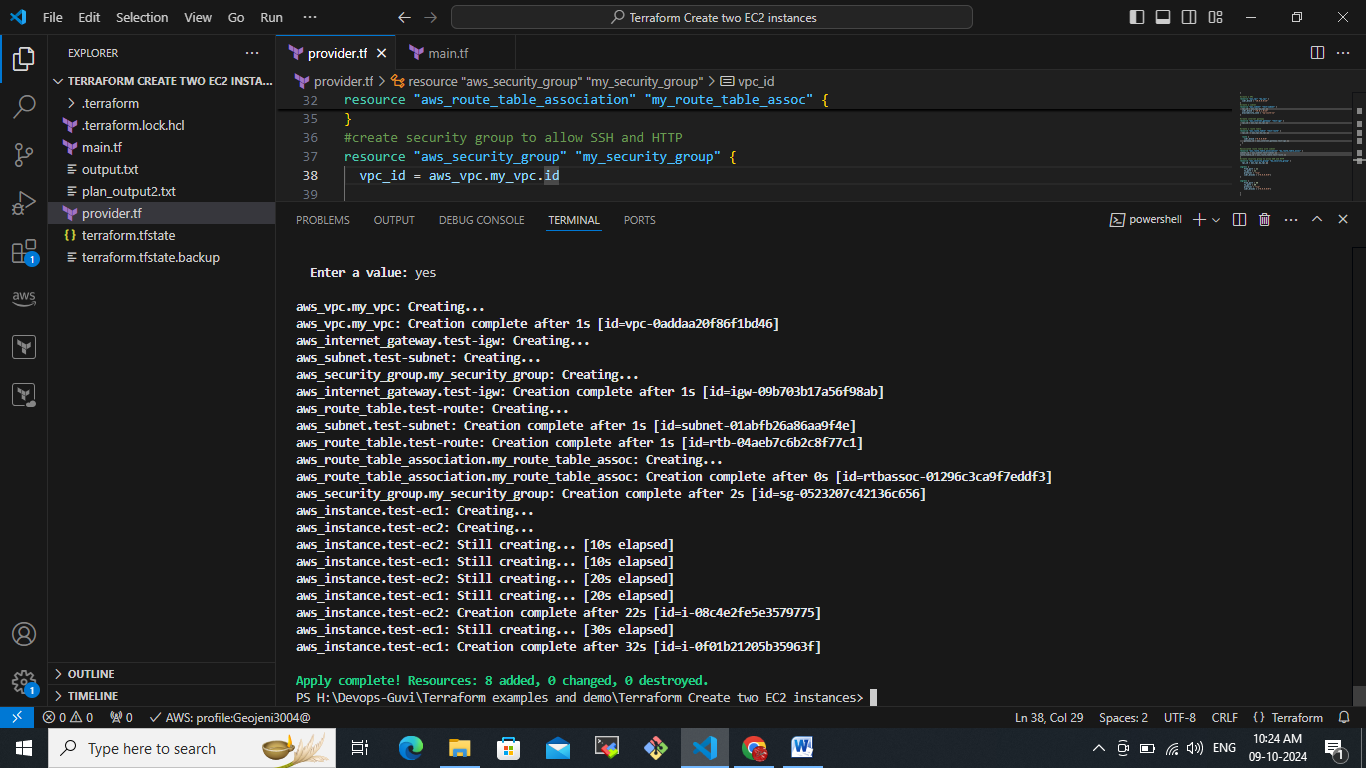
Initiate terraform  
terraform init  


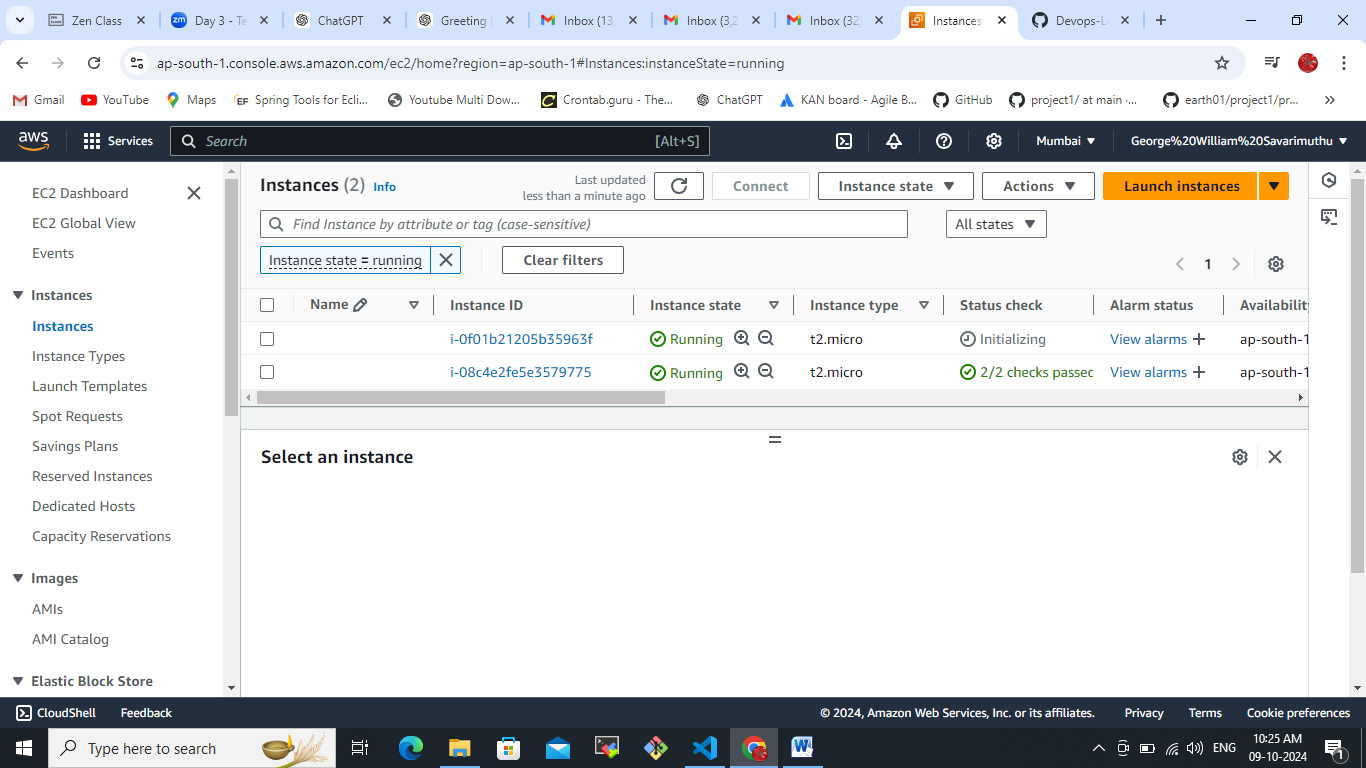
Terraform initiated successfully and I have provided my region and subnet inter gateway all the providing stuff from AWS I have provided the provider.tf file.  
  
now we have plan this terraform file  
  
  
  
  
  
  
  


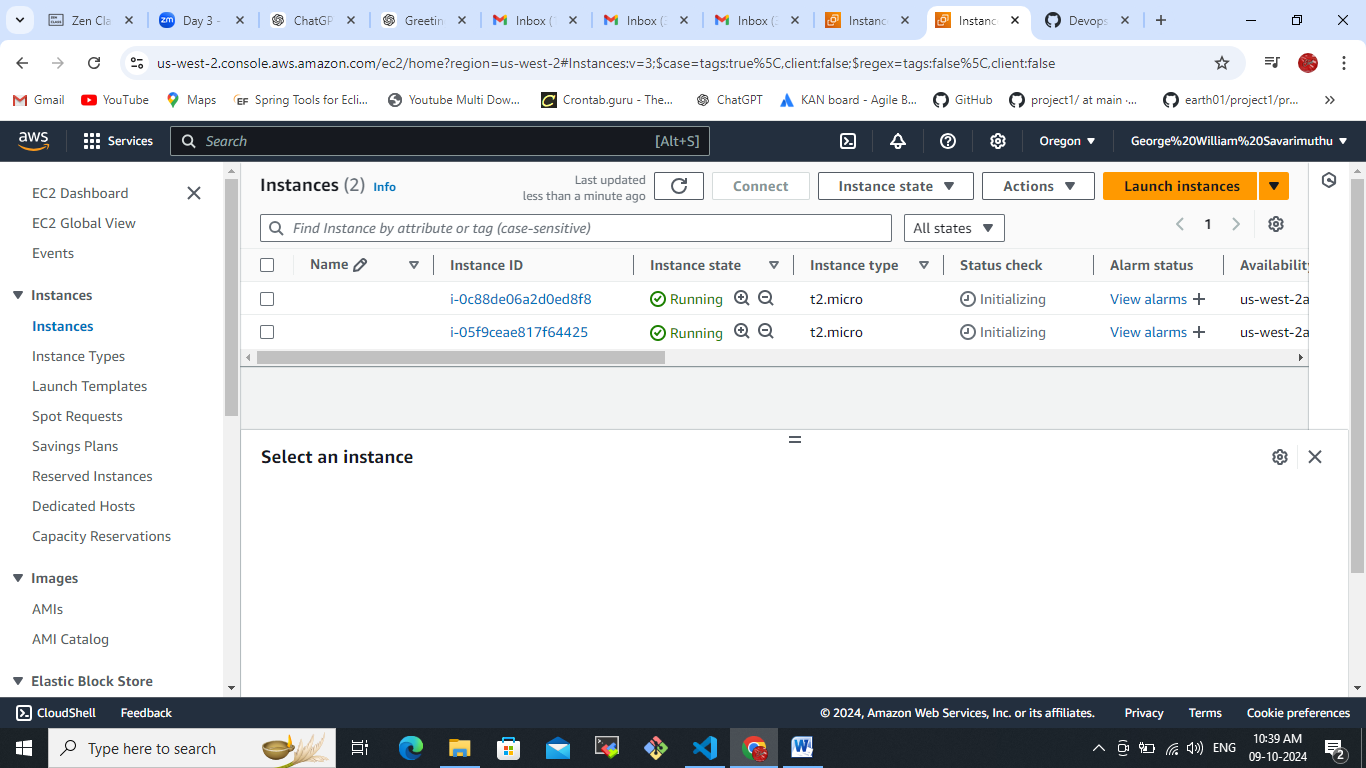


Now terraform apply to get all our requested service will mentioned in the main.tf  






Now validate in Amazon EC2 – ap-south-1a  
  


Validate another region us-west-2a  


Provider.tf file for south zone

provider "aws" {

  region  ="ap-south-1"

}

#create a VPC

resource "aws\_vpc" "my\_vpc" {

  cidr\_block = "10.0.0.0/16"

}

#create a subnet

resource "aws\_subnet" "test-subnet" {

  vpc\_id = aws\_vpc.my\_vpc.id

  cidr\_block = "10.0.1.0/24"

  availability\_zone = "ap-south-1a"

}

#create internet gateway

resource "aws\_internet\_gateway" "test-igw" {

  vpc\_id = aws\_vpc.my\_vpc.id

}

#create a route table

resource "aws\_route\_table" "test-route" {

  vpc\_id  = aws\_vpc.my\_vpc.id

  route {

    cidr\_block ="0.0.0.0/0"

    gateway\_id = aws\_internet\_gateway.test-igw.id

  }

}

#associated route table with subnet

resource "aws\_route\_table\_association" "my\_route\_table\_assoc" {

subnet\_id = aws\_subnet.test-subnet.id

route\_table\_id = aws\_route\_table.test-route.id

}

#create security group to allow SSH and HTTP

resource "aws\_security\_group" "my\_security\_group" {

  vpc\_id = aws\_vpc.my\_vpc.id

ingress {

    from\_port = 22

    to\_port = 22

    protocol = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

}

ingress {

    from\_port = 80

    to\_port = 80

    protocol = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

}

}

Main.tf  
#create two ec2 instances here the first one

resource "aws\_instance" "test-ec1" {

  ami = "ami-08718895af4dfa033"

  instance\_type = "t2.micro"

  subnet\_id = aws\_subnet.test-subnet.id

  vpc\_security\_group\_ids = [aws\_security\_group.my\_security\_group.id]

}

#create second ec2 instance

resource "aws\_instance" "test-ec2" {

  ami = "ami-08718895af4dfa033"

  instance\_type = "t2.micro"

  subnet\_id = aws\_subnet.test-subnet.id

  vpc\_security\_group\_ids = [aws\_security\_group.my\_security\_group.id]

}

Provider.tf for us-west-2  
provider "aws" {

  region  ="us-west-2"

}

#create a VPC

resource "aws\_vpc" "my\_vpc" {

  cidr\_block = "10.0.0.0/16"

}

#create a subnet

resource "aws\_subnet" "test-subnet" {

  vpc\_id = aws\_vpc.my\_vpc.id

  cidr\_block = "10.0.1.0/24"

  availability\_zone = "us-west-2a"

}

#create internet gateway

resource "aws\_internet\_gateway" "test-igw" {

  vpc\_id = aws\_vpc.my\_vpc.id

}

#create a route table

resource "aws\_route\_table" "test-route" {

  vpc\_id  = aws\_vpc.my\_vpc.id

  route {

    cidr\_block ="0.0.0.0/0"

    gateway\_id = aws\_internet\_gateway.test-igw.id

  }

}

#associated route table with subnet

resource "aws\_route\_table\_association" "my\_route\_table\_assoc" {

subnet\_id = aws\_subnet.test-subnet.id

route\_table\_id = aws\_route\_table.test-route.id

}

#create security group to allow SSH and HTTP

resource "aws\_security\_group" "my\_security\_group" {

  vpc\_id = aws\_vpc.my\_vpc.id

ingress {

    from\_port = 22

    to\_port = 22

    protocol = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

}

ingress {

    from\_port = 80

    to\_port = 80

    protocol = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

}

}

Main.tf  
#create two ec2 instances here the first one

resource "aws\_instance" "test-ec1" {

  ami = "ami-0d081196e3df05f4d"

  instance\_type = "t2.micro"

  subnet\_id = aws\_subnet.test-subnet.id

  vpc\_security\_group\_ids = [aws\_security\_group.my\_security\_group.id]

}

#create second ec2 instance

resource "aws\_instance" "test-ec2" {

  ami = "ami-0d081196e3df05f4d"

  instance\_type = "t2.micro"

  subnet\_id = aws\_subnet.test-subnet.id

  vpc\_security\_group\_ids = [aws\_security\_group.my\_security\_group.id]

}