**1.create an instance using data block in a vpc which is created manually outside of terraform**

**main.tf file**

# Configure the AWS provider

provider "aws" {

    region = "ap-south-1"

}

# Create a VPC

resource "aws\_vpc" "terran\_vpc" {

    cidr\_block = "10.0.0.0/16"

    tags = {

        Name = "terran-vpc"

    }

}

# Create a Subnet

resource "aws\_subnet" "terran\_subnet" {

    vpc\_id                  = aws\_vpc.terran\_vpc.id

    cidr\_block              = "10.0.1.0/24"

    availability\_zone       = "ap-south-1b"

    map\_public\_ip\_on\_launch = true

    tags = {

        Name = "terran-subnet"

    }

}

# Create an Internet Gateway

resource "aws\_internet\_gateway" "terran\_igw" {

    vpc\_id = aws\_vpc.terran\_vpc.id

    tags = {

        Name = "terran-igw"

    }

}

# Create a Route Table

resource "aws\_route\_table" "terran\_rt" {

    vpc\_id = aws\_vpc.terran\_vpc.id

    route {

        cidr\_block = "0.0.0.0/0"

        gateway\_id = aws\_internet\_gateway.terran\_igw.id

    }

    tags = {

        Name = "terran-rt"

    }

}

# Associate Route Table with Subnet

resource "aws\_route\_table\_association" "terran\_rt\_association" {

    subnet\_id      = aws\_subnet.terran\_subnet.id

    route\_table\_id = aws\_route\_table.terran\_rt.id

}

# Create a Security Group

resource "aws\_security\_group" "terran\_sg" {

    description = "Allow SSH traffic"

    vpc\_id      = aws\_vpc.terran\_vpc.id

ingress {

        description = "Allow SSH"

        from\_port   = 22

        to\_port     = 22

        protocol    = "tcp"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    egress {

        from\_port   = 0

        to\_port     = 0

        protocol    = "-1"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    tags = {

        Name = "terran-sg"

    }

}

# Create an EC2 Instance

resource "aws\_instance" "terran\_instance" {

    subnet\_id       = aws\_subnet.terran\_subnet.id

    ami             = "ami-0ad21ae1d0696ad58"

    instance\_type   = "t2.micro"

    key\_name        = "terraform"

    security\_groups  = [aws\_security\_group.terran\_sg.id]  # Use security\_group\_ids instead of security\_group\_id

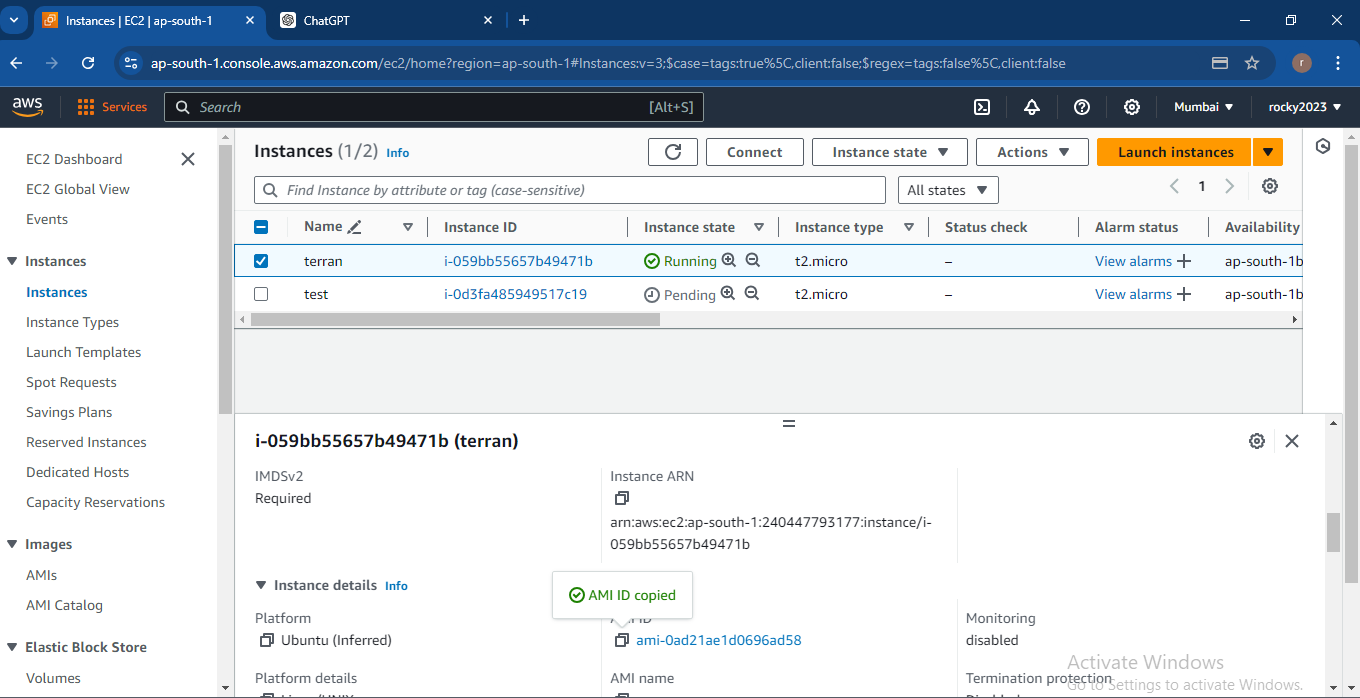
    tags = {

        Name = "terraform-instance"

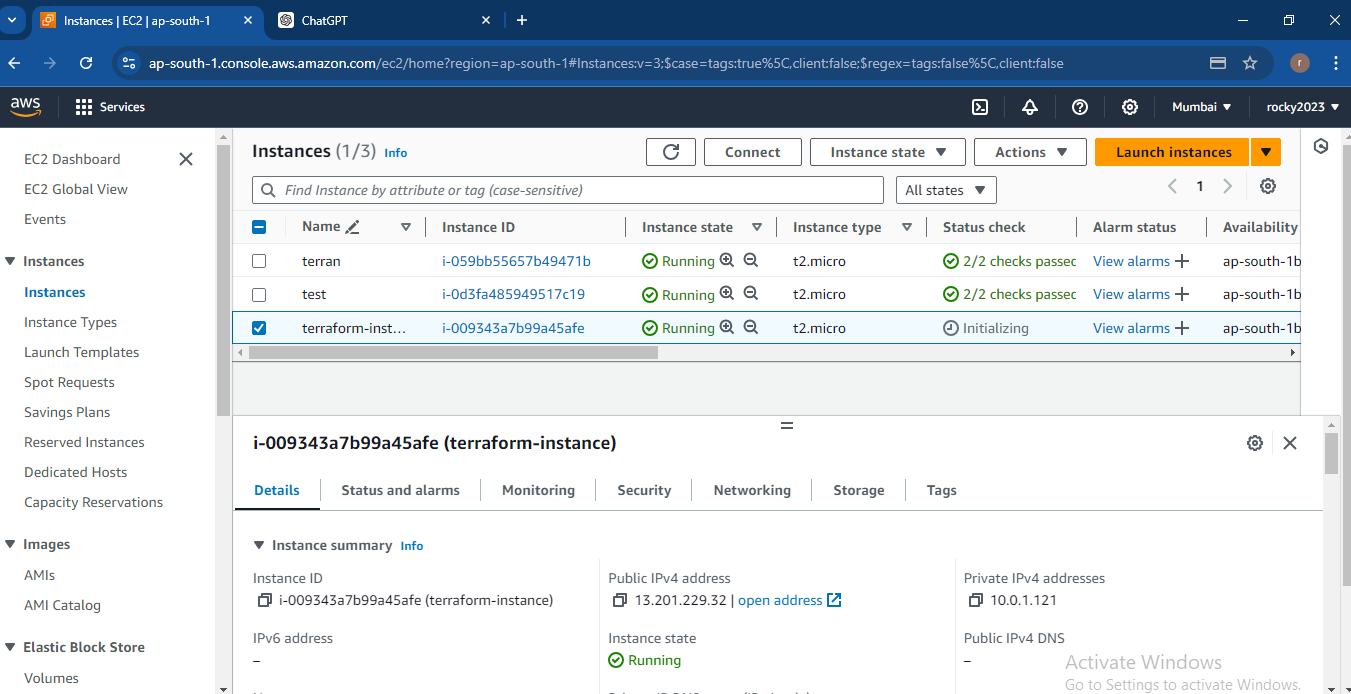
    }

}

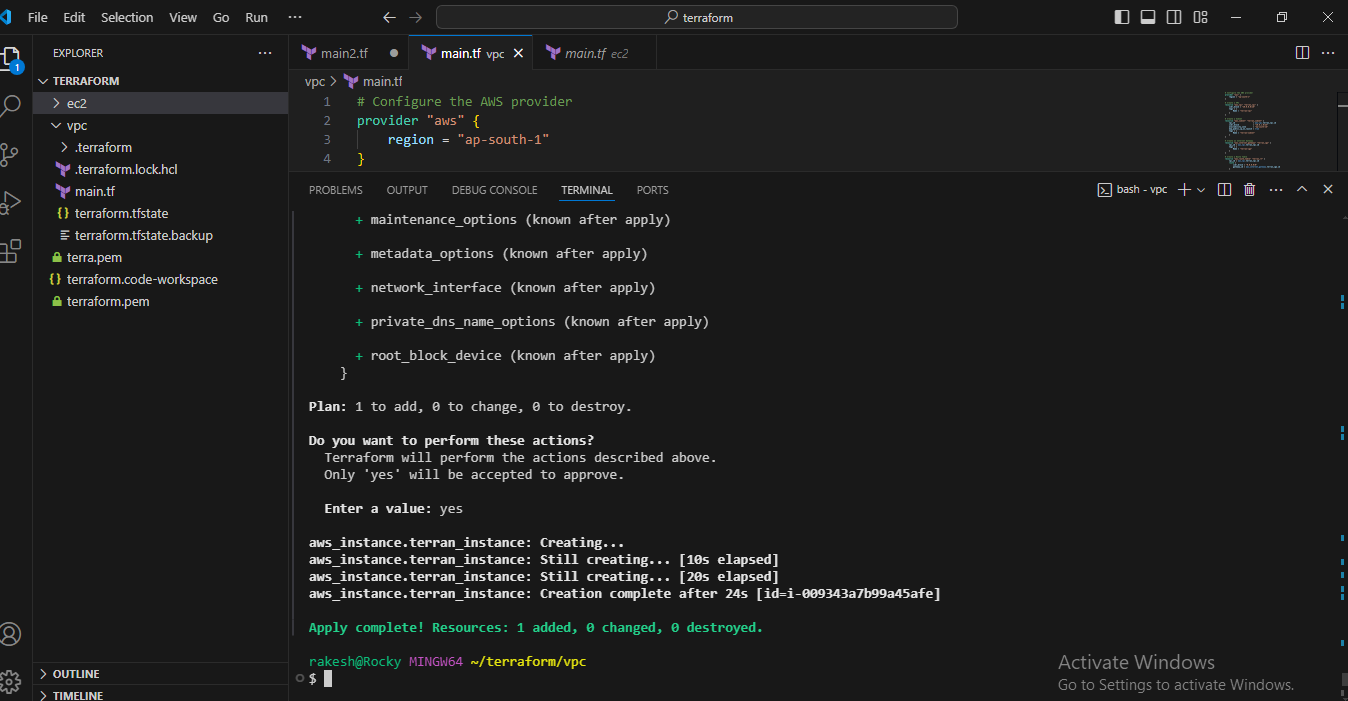
**Before initializing tf file**

****

**After initializing tf file terraform-instance created…**

****

**Executed result with instance id**

****

**2)**  **create instances in multiple regions with same config file using an alias**

**Main.tf file**

# Configure the AWS provider for ap-south-1

provider "aws" {

    alias  = "ap\_south"

    region = "ap-south-1"

}

# Configure the AWS provider for us-west-1

provider "aws" {

    alias  = "us\_west"

    region = "us-west-1"

}

# Define resources in ap-south-1

# Create a VPC in ap-south-1

resource "aws\_vpc" "terran\_vpc\_ap\_south" {

    provider = aws.ap\_south

    cidr\_block = "10.0.0.0/16"

    tags = {

        Name = "terran-vpc-ap-south"

    }

}

# Create a Subnet in ap-south-1

resource "aws\_subnet" "terran\_subnet\_ap\_south" {

    provider               = aws.ap\_south

    vpc\_id                 = aws\_vpc.terran\_vpc\_ap\_south.id

    cidr\_block             = "10.0.1.0/24"

    availability\_zone      = "ap-south-1b"

    map\_public\_ip\_on\_launch = true

    tags = {

        Name = "terran-subnet-ap-south"

    }

}

# Create a Security Group in ap-south-1

resource "aws\_security\_group" "terran\_sg\_ap\_south" {

    provider    = aws.ap\_south

    description = "Allow SSH traffic"

    vpc\_id     = aws\_vpc.terran\_vpc\_ap\_south.id

    ingress {

        description = "Allow SSH"

        from\_port   = 22

        to\_port     = 22

        protocol    = "tcp"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    egress {

        from\_port   = 0

        to\_port     = 0

        protocol    = "-1"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    tags = {

        Name = "terran-sg-ap-south"

    }

}

# Create an EC2 Instance in ap-south-1

resource "aws\_instance" "terran\_instance\_ap\_south" {

    provider            = aws.ap\_south

    ami                  = "ami-0ad21ae1d0696ad58"  # Verify the AMI ID is valid in ap-south-1

    instance\_type        = "t2.micro"

    subnet\_id            = aws\_subnet.terran\_subnet\_ap\_south.id

    key\_name             = "terraform"

    security\_groups   = [aws\_security\_group.terran\_sg\_ap\_south.id]

    tags = {

        Name = "terraform-instance-ap-south"

    }

}

# Define resources in us-west-1

# Create a VPC in us-west-1

resource "aws\_vpc" "terran\_vpc\_us\_west" {

    provider = aws.us\_west

    cidr\_block = "10.1.0.0/16"

    tags = {

        Name = "terran-vpc-us-west-1"

    }

}

# Create a Subnet in us-west-1

resource "aws\_subnet" "terran\_subnet\_us\_west" {

    provider               = aws.us\_west

    vpc\_id                 = aws\_vpc.terran\_vpc\_us\_west.id

    cidr\_block             = "10.1.1.0/24"

    availability\_zone      = "us-west-1a"

    map\_public\_ip\_on\_launch = true

    tags = {

        Name = "terran-subnet-us-west-1"

    }

}

# Create a Security Group in us-west-1

resource "aws\_security\_group" "terran\_sg\_us\_west" {

    provider    = aws.us\_west

    description = "Allow SSH traffic"

    vpc\_id     = aws\_vpc.terran\_vpc\_us\_west.id

    ingress {

        description = "Allow SSH"

        from\_port   = 22

        to\_port     = 22

        protocol    = "tcp"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    egress {

        from\_port   = 0

        to\_port     = 0

        protocol    = "-1"

        cidr\_blocks = ["0.0.0.0/0"]

    }

    tags = {

        Name = "terran-sg-us-west-1"

    }

}

# Create an EC2 Instance in us-west-1

resource "aws\_instance" "terran\_instance\_us\_west" {

    provider            = aws.us\_west

    ami                  = "ami-0ff591da048329e00"  # Verify the AMI ID is valid in us-west-1

    instance\_type        = "t2.micro"

    subnet\_id            = aws\_subnet.terran\_subnet\_us\_west.id

    key\_name             = "qpa-test"

    security\_groups  = [aws\_security\_group.terran\_sg\_us\_west.id]

    tags = {

        Name = "terraform-instance-us-west-1"

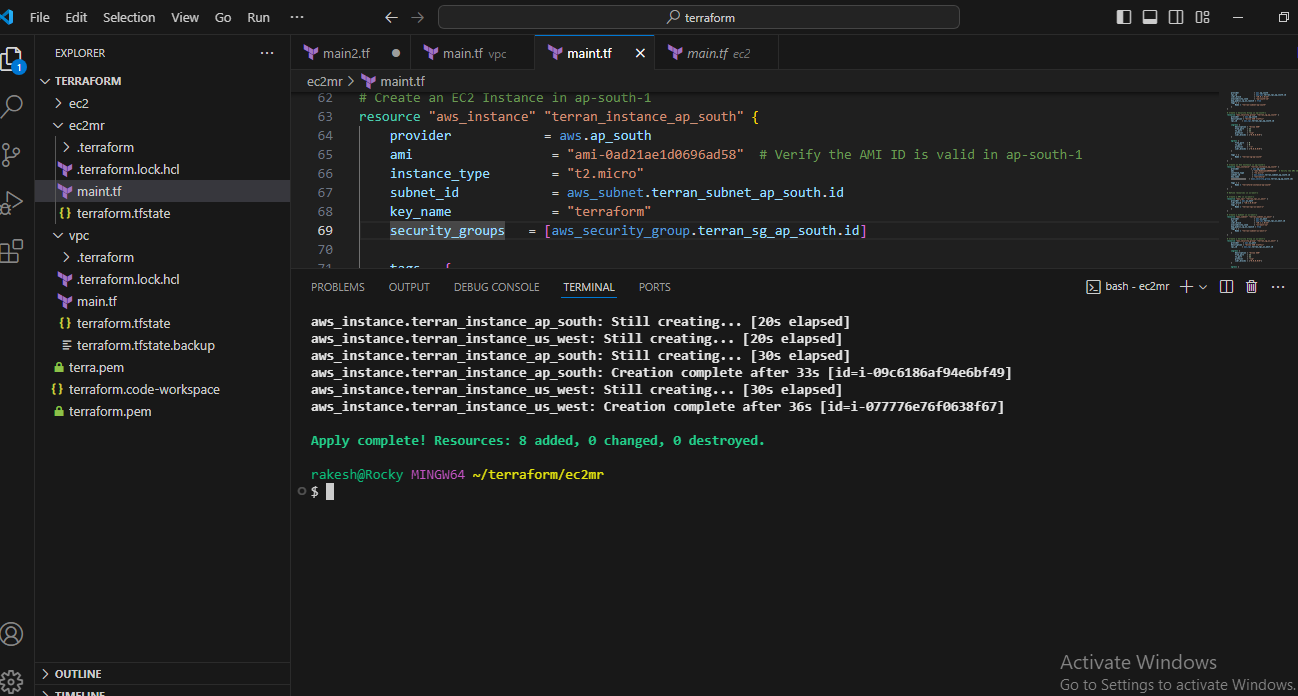
    }

}

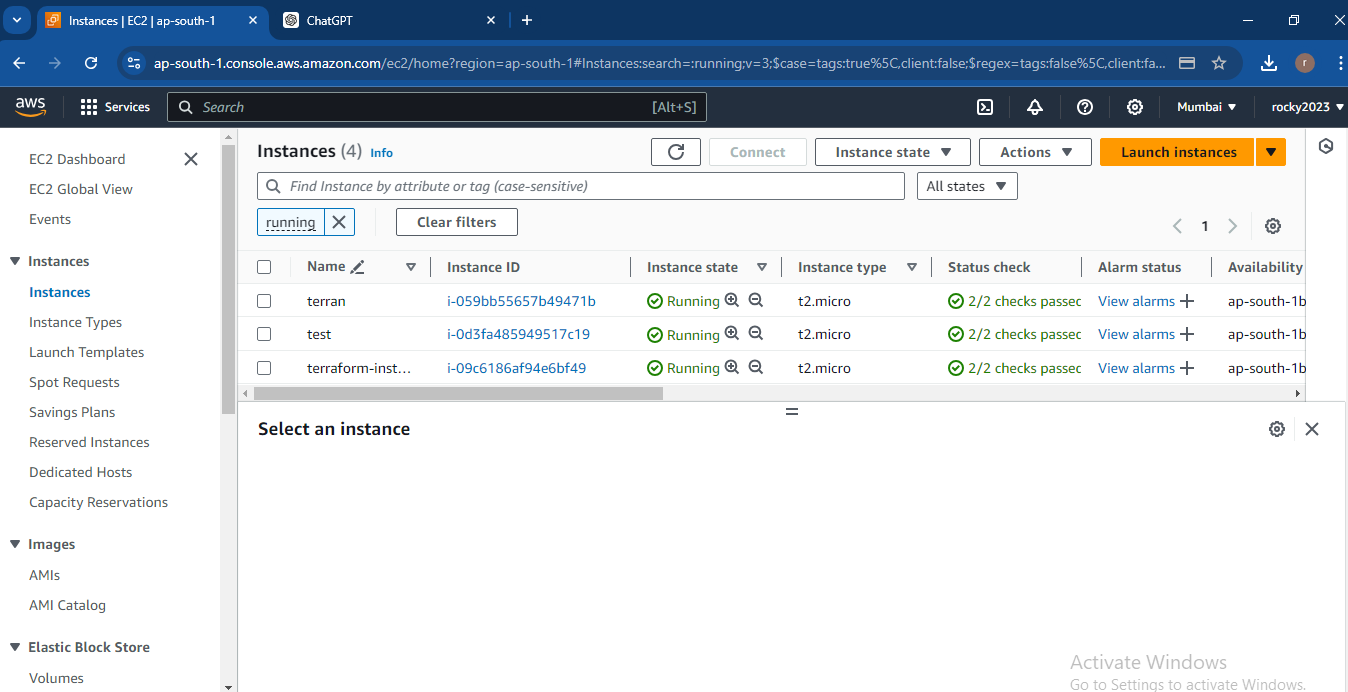
**Executed results for creating instances in multiple regions**

**1)ap-south-1**

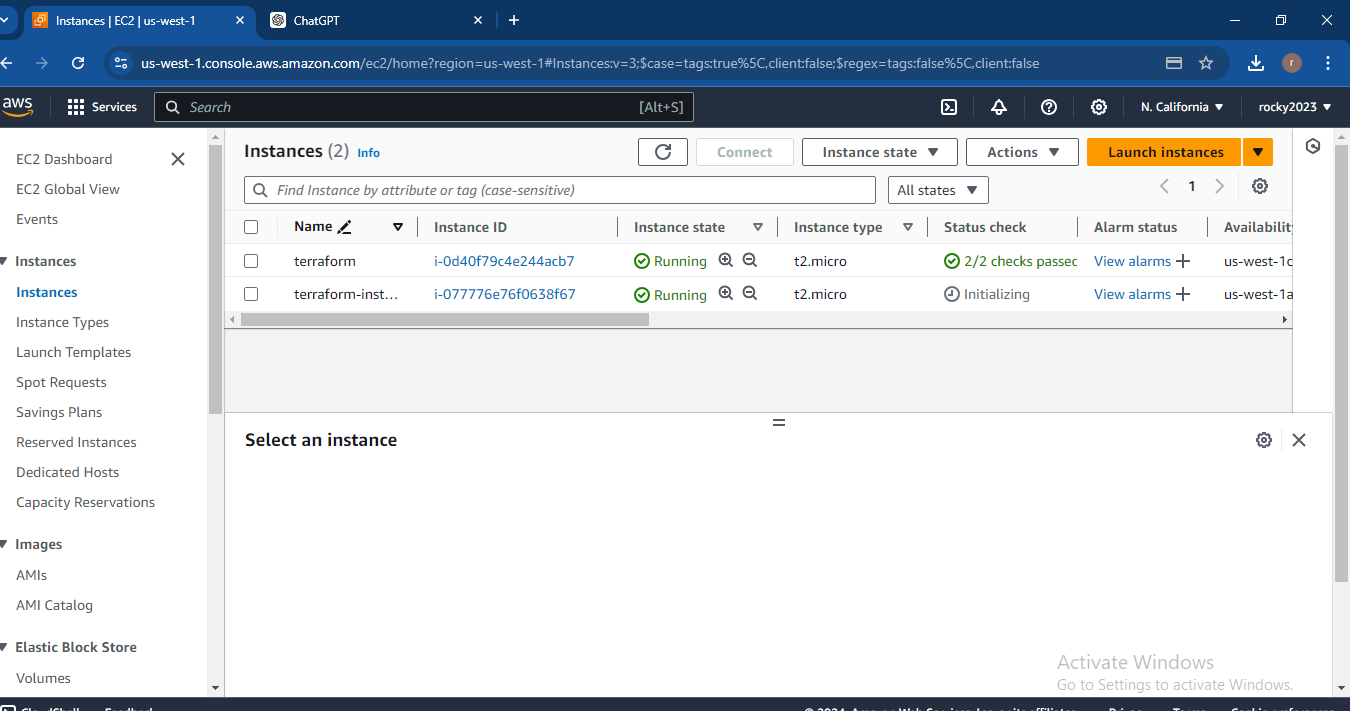
**2)us-west-1**

****

**Terraform instance at ap-south-1 region**

****

**Us-west-1 region**

****

**3) Create multiple instances using for\_each**

# Configure the AWS provider for ap-south-1

provider "aws" {

  alias  = "ap\_south"

  region = "ap-south-1"

}

# Define a map of instance configurations

locals {

  instance\_configs = {

    "instance1" = {

      ami           = "ami-0ad21ae1d0696ad58"  # Verify AMI ID for ap-south-1

      instance\_type = "t2.micro"

      key\_name      = "terraform"

      subnet\_id     = aws\_subnet.terran\_subnet\_ap\_south.id

      security\_groups = [aws\_security\_group.terran\_sg\_ap\_south.id]

      tags = {

        Name = "terraform-instance-1"

      }

    },

    "instance2" = {

      ami           = "ami-0ad21ae1d0696ad58"  # Verify AMI ID for ap-south-1

      instance\_type = "t2.micro"

      key\_name      = "terraform"

      subnet\_id     = aws\_subnet.terran\_subnet\_ap\_south.id

      security\_groups = [aws\_security\_group.terran\_sg\_ap\_south.id]

      tags = {

        Name = "terraform-instance-2"

      }

    },

    "instance3" = {

      ami           = "ami-0ad21ae1d0696ad58"  # Verify AMI ID for ap-south-1

      instance\_type = "t2.micro"

      key\_name      = "terraform"

      subnet\_id     = aws\_subnet.terran\_subnet\_ap\_south.id

      security\_groups = [aws\_security\_group.terran\_sg\_ap\_south.id]

      tags = {

        Name = "terraform-instance-3"

      }

    }

  }

}

# Create a VPC in ap-south-1

resource "aws\_vpc" "terran\_vpc\_ap\_south" {

  provider  = aws.ap\_south

  cidr\_block = "10.0.0.0/16"

  tags = {

    Name = "terran-vpc-ap-south"

  }

}

# Create a Subnet in ap-south-1

resource "aws\_subnet" "terran\_subnet\_ap\_south" {

  provider               = aws.ap\_south

  vpc\_id                 = aws\_vpc.terran\_vpc\_ap\_south.id

  cidr\_block             = "10.0.1.0/24"

  availability\_zone      = "ap-south-1b"

  map\_public\_ip\_on\_launch = true

  tags = {

    Name = "terran-subnet-ap-south"

  }

}

# Create a Security Group in ap-south-1

resource "aws\_security\_group" "terran\_sg\_ap\_south" {

  provider    = aws.ap\_south

  description = "Allow SSH traffic"

  vpc\_id      = aws\_vpc.terran\_vpc\_ap\_south.id

  ingress {

    description = "Allow SSH"

    from\_port   = 22

    to\_port     = 22

    protocol    = "tcp"

    cidr\_blocks = ["0.0.0.0/0"]

  }

  egress {

    from\_port   = 0

    to\_port     = 0

    protocol    = "-1"

    cidr\_blocks = ["0.0.0.0/0"]

  }

  tags = {

    Name = "terran-sg-ap-south"

  }

}

# Create EC2 Instances in ap-south-1 using for\_each

resource "aws\_instance" "terran\_instance\_ap\_south" {

  for\_each = local.instance\_configs

  provider            = aws.ap\_south

  ami                  = each.value.ami

  instance\_type        = each.value.instance\_type

  subnet\_id            = each.value.subnet\_id

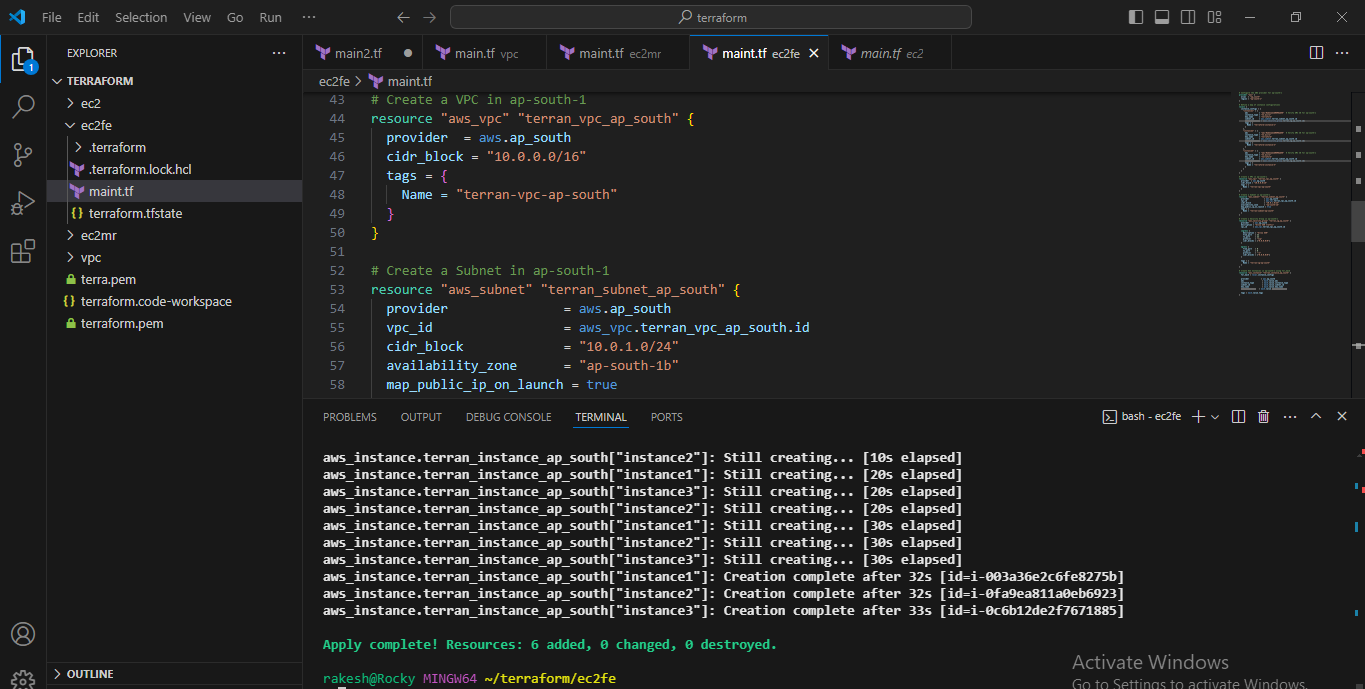
  key\_name             = each.value.key\_name

  security\_groups   = each.value.security\_groups

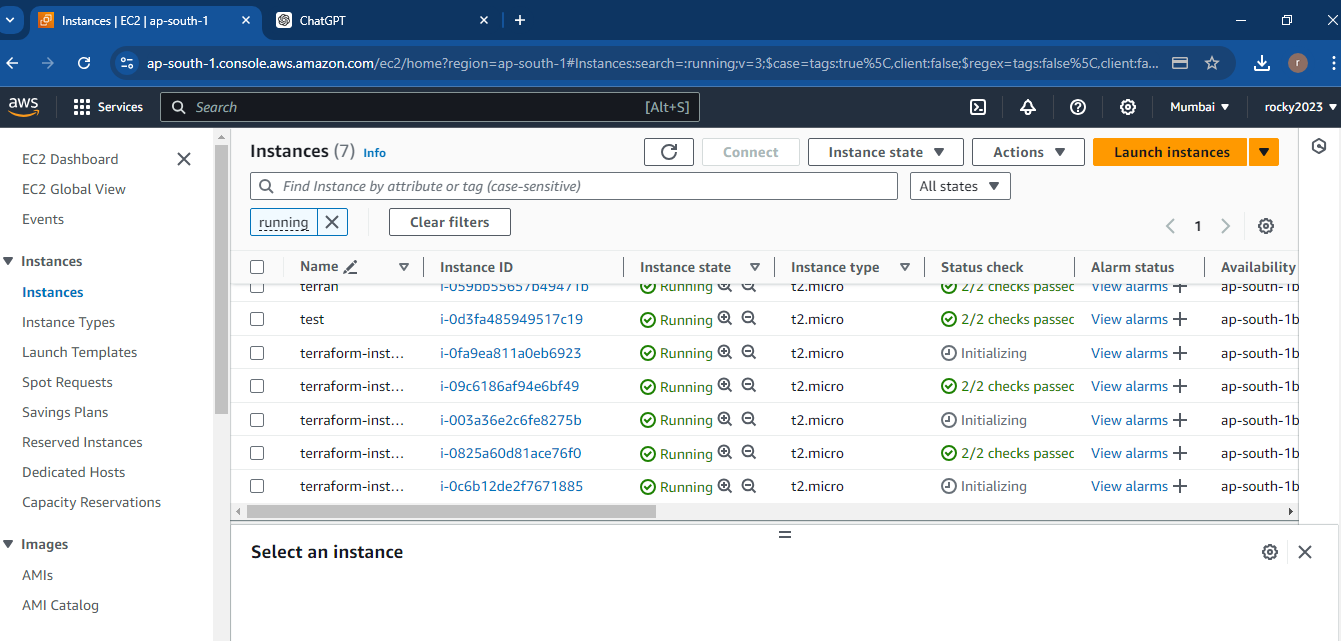
  tags = each.value.tags

}

**Executed result**

****

**Created multiple instances in aws that initializing instances**

****