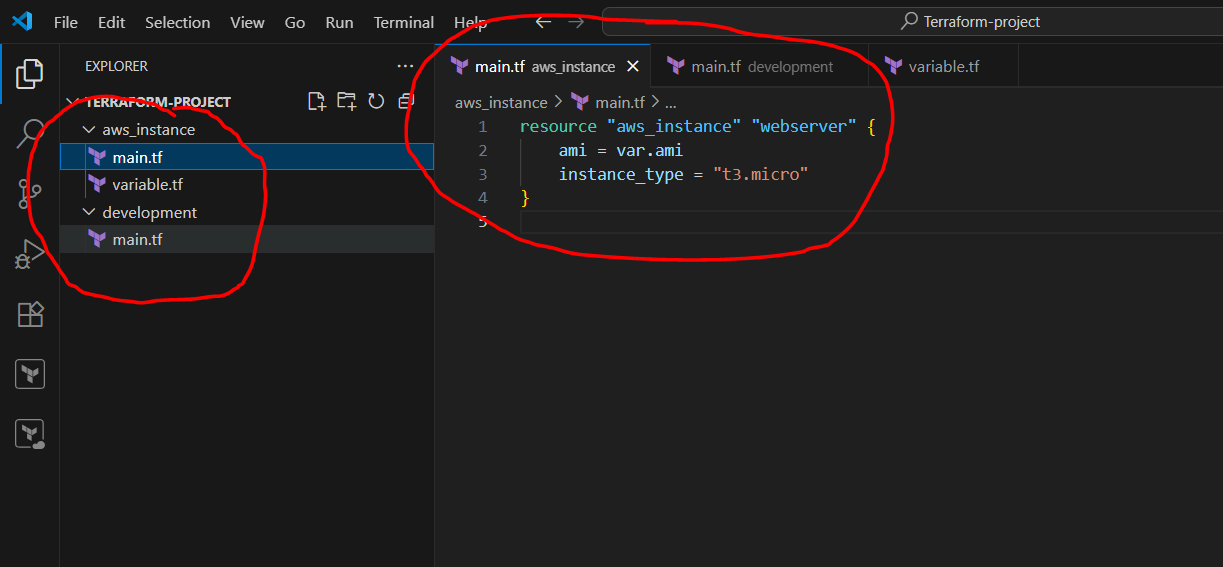
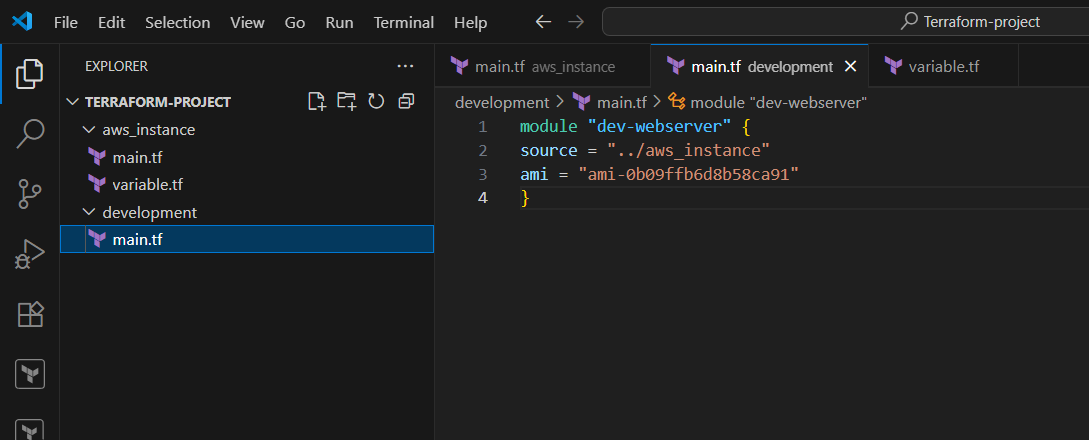
1. **Watch the Terraform-06 video.**

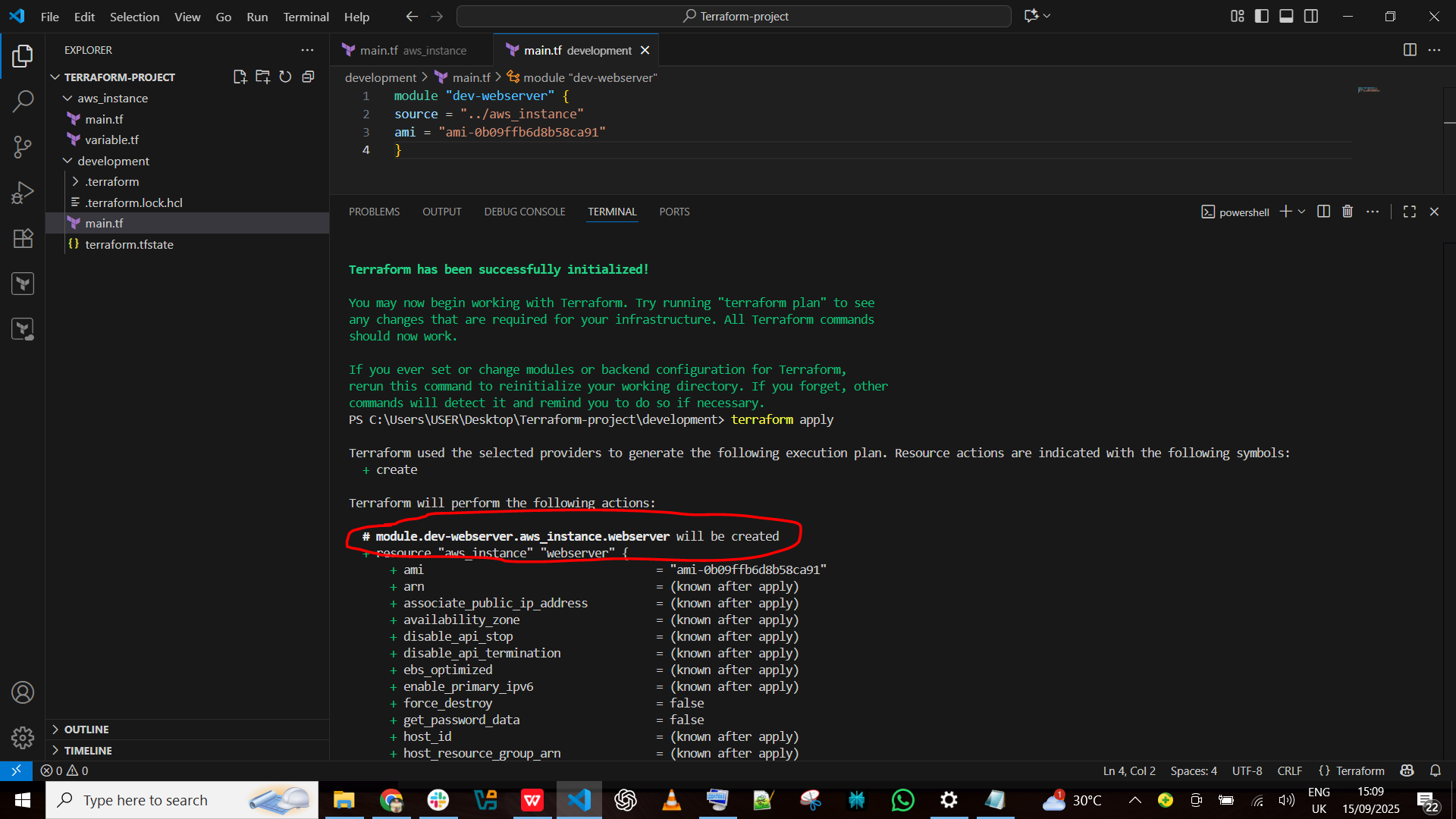
Completed the video

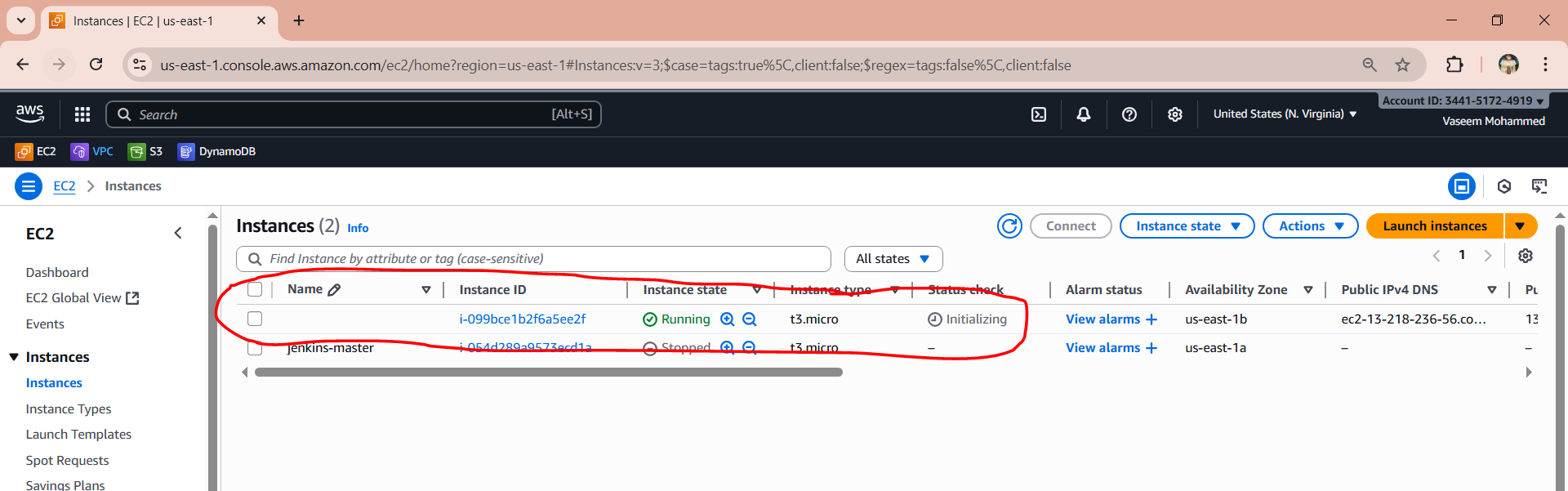
1. **Execute the script shown in the video.**
2. Here we created 2 directories and files in that with main.tf and variable.tf



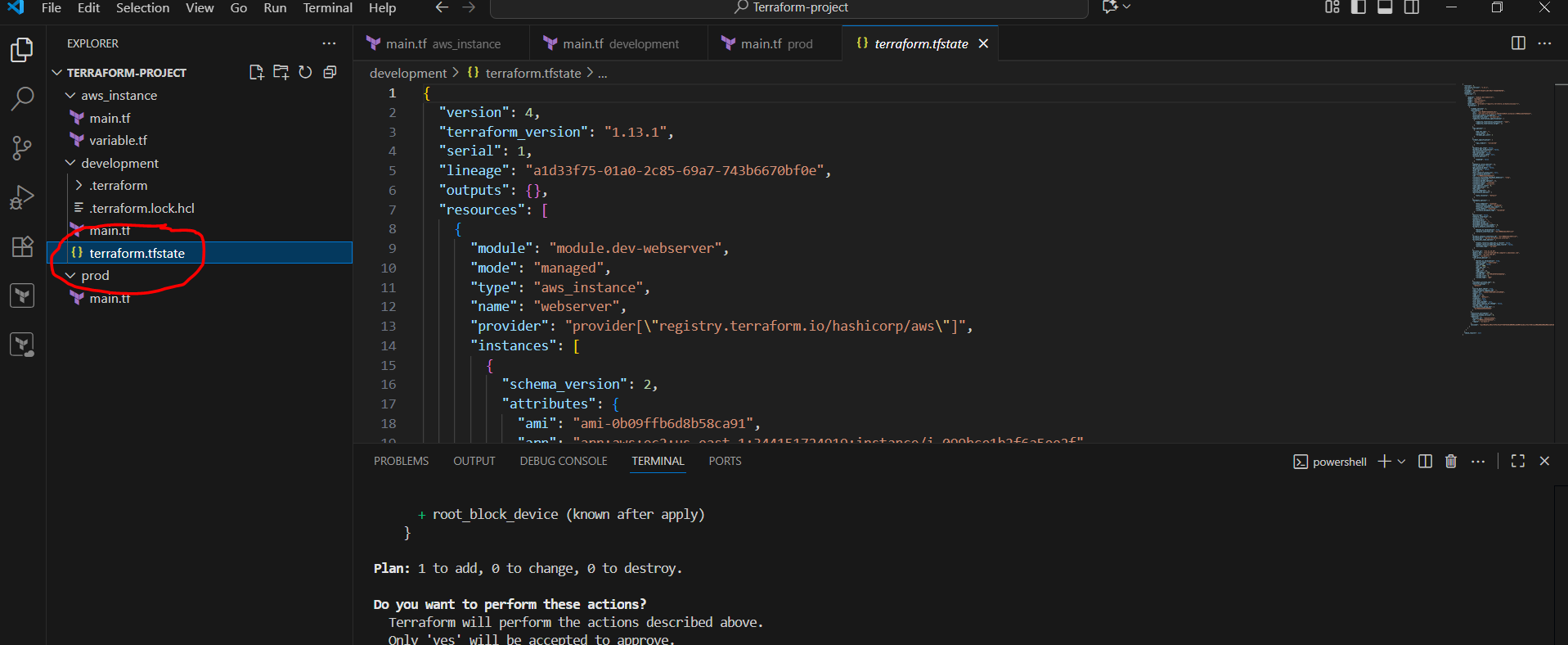


1. Now we have to go in development directory cd ./development --> terraform init --> terraform apply and from that particular module it is creating one instance

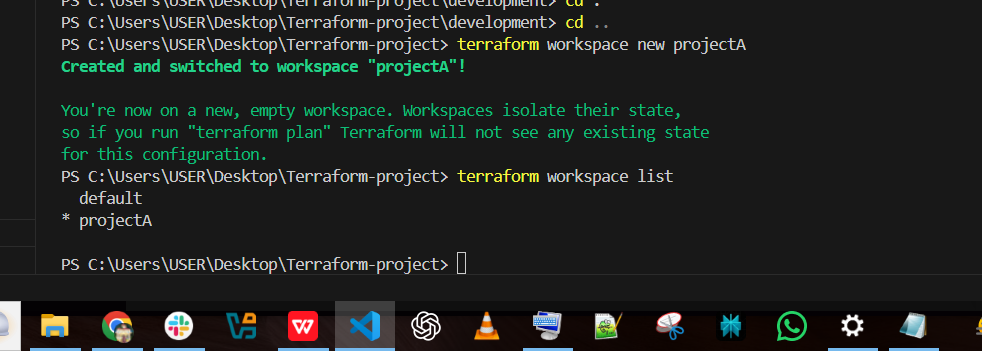




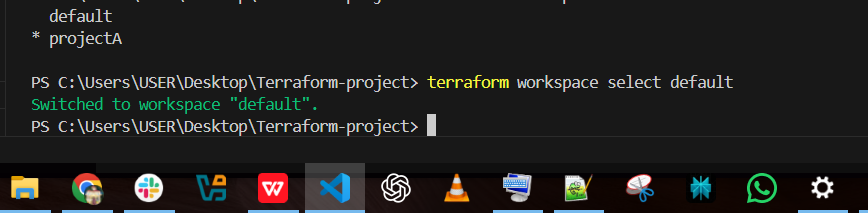
1. And the terraform.tfstate file will be only in development directory

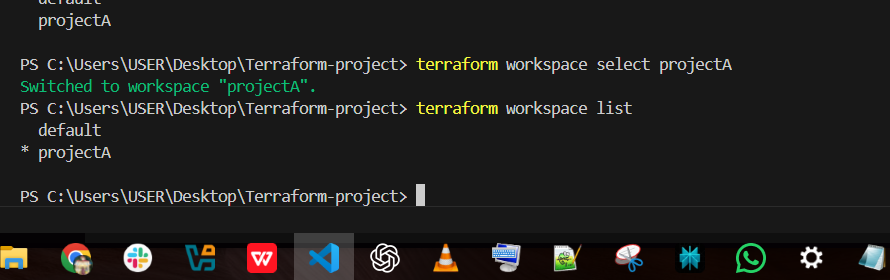


1. Now we have created the **Terraform Workspace**  using the command terraform workspace new projectA --> terraform workspace list \*projectA is our current created workspace

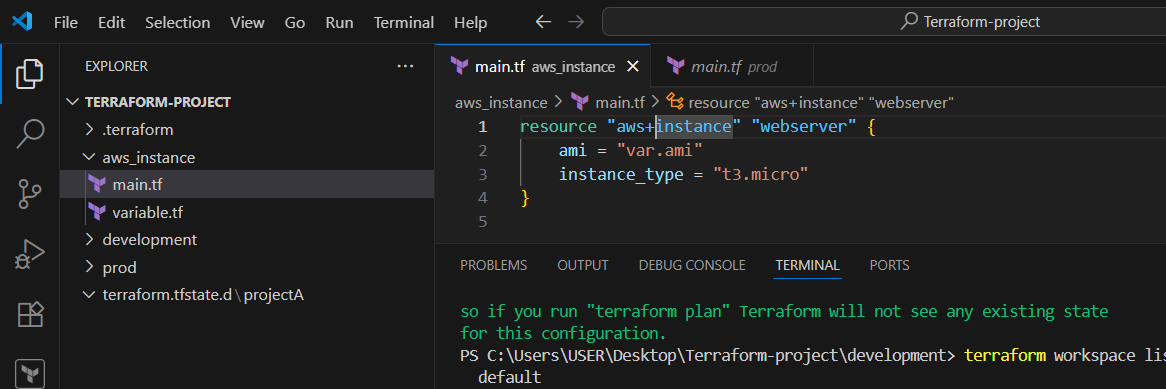


1. If we want to switch to default give command terraform workspace select default

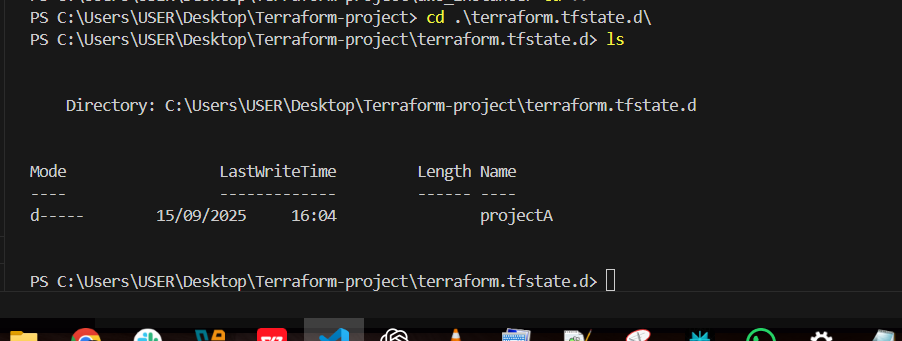




1. Now we have created the file main.tf and variable.tf



1. So here we have created the projectA and it will get in terraform.tfstate file cd .. --> cd .\terraform.tfstate.d\ --> ls



1. **Provision EC2, S3, and VPC using Terraform modules.**

Here we have created the aws\_instance folder, EC2, VPC and main.tf and variable.tf file, output.tf in every folder and updated the script to launch instance from using terraform script

**EC2:**

resource "aws\_instance" "web" {

ami = var.ami

instance\_type = var.instance\_type

subnet\_id = var.subnet\_id

tags = { Name = var.name }

}

variable "ami" {}

variable "instance\_type" {}

variable "subnet\_id" {}

variable "name" {}

**VPC:**

resource "aws\_vpc" "main" {

cidr\_block = var.cidr\_block

tags = { Name = var.name }

}

resource "aws\_subnet" "public" {

vpc\_id = aws\_vpc.main.id

cidr\_block = var.public\_subnet\_cidr

map\_public\_ip\_on\_launch = true

tags = { Name = "${var.name}-public-subnet" }

}

variable "cidr\_block" {}

variable "public\_subnet\_cidr" {}

variable "name" {}

**S3:**

resource "aws\_s3\_bucket" "bucket" {

bucket = var.bucket\_name

acl = "private"

}

variable "bucket\_name" {}

**Module-Main.tf:**

provider "aws" {

region = "us-east-1"

}

module "vpc" {

source = "./modules/vpc"

cidr\_block = "10.0.0.0/16"

public\_subnet\_cidr = "10.0.1.0/24"

name = "my-vpc"

}

module "s3" {

source = "./modules/s3"

bucket\_name = "**new-vaseem-bucket-000**"

}

module "ec2" {

source = "./modules/ec2"

ami = "ami-0d85d4f07a62e2969"

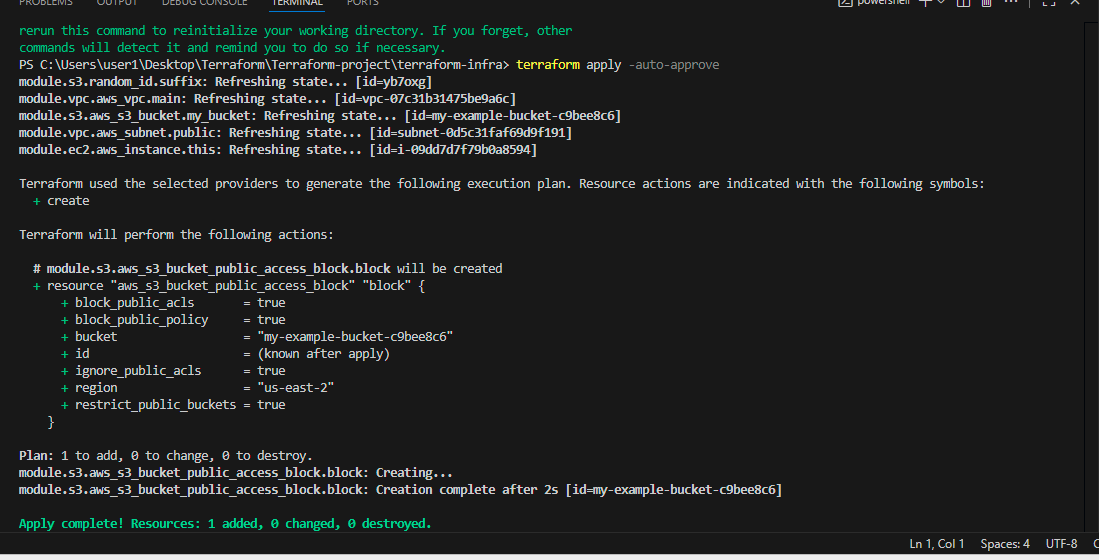
instance\_type = "t2.micro"

subnet\_id = module.vpc.public\_subnet\_id

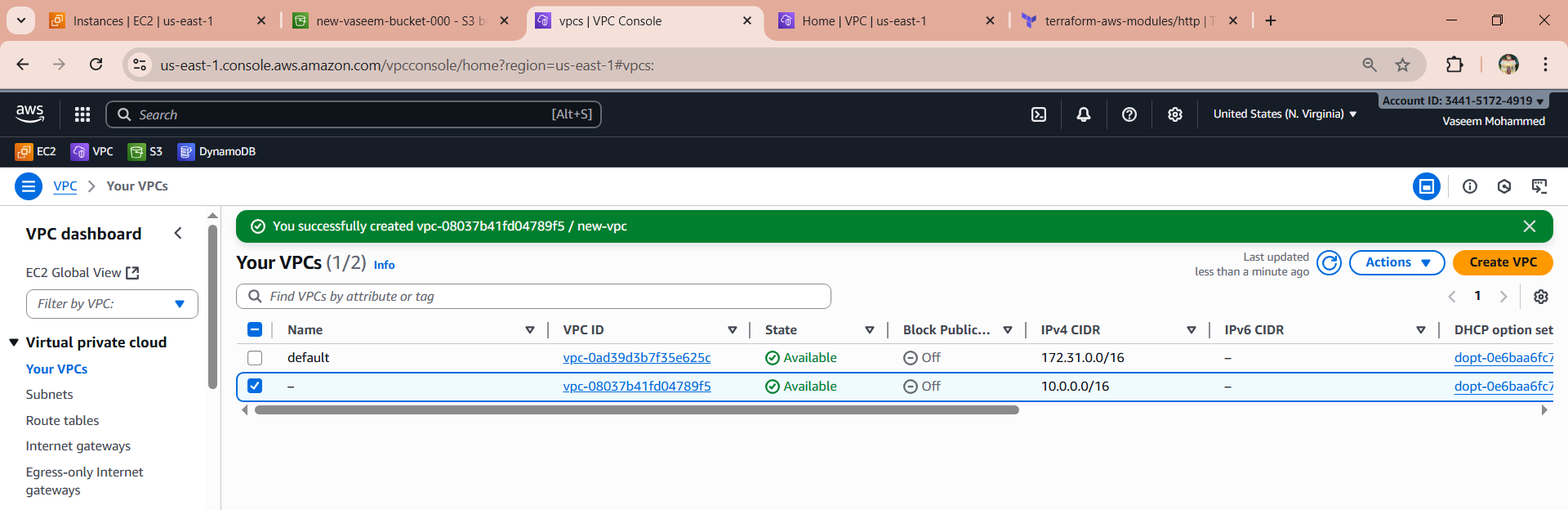
name = "my-ec2-instance"

}

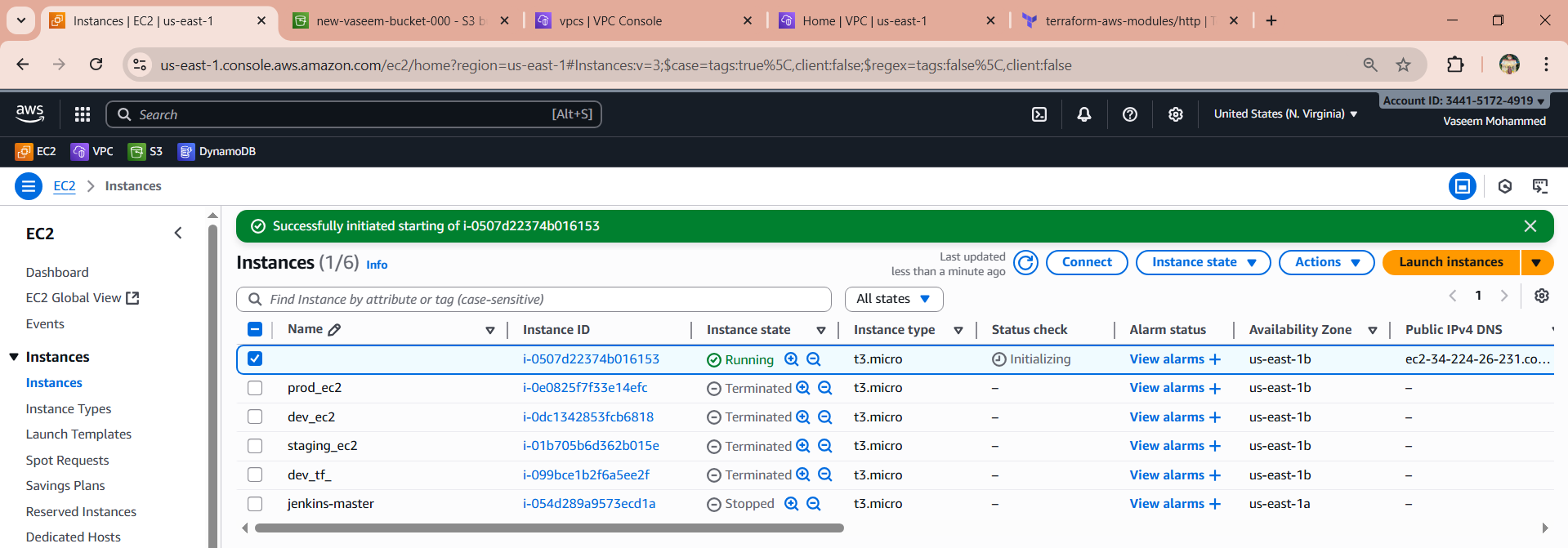
After tjis need to give command **terraform init --> terraform plan--> terraform apply**



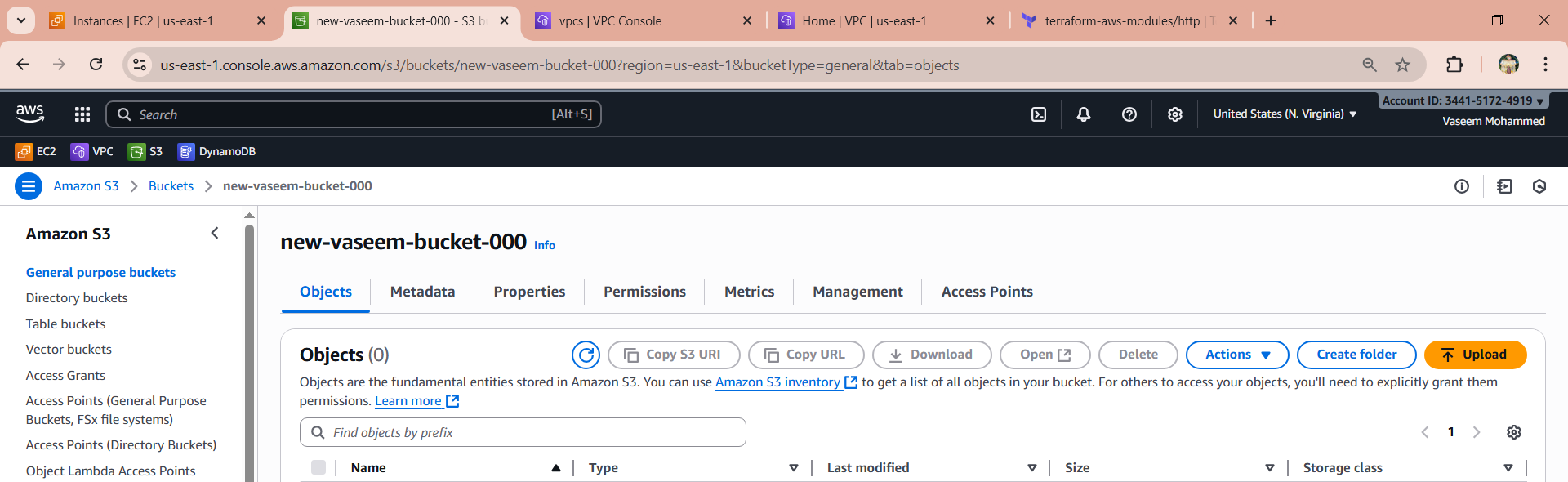
Now VPC created



And instance also created



And s3 bucket created



1. **Provision EC2 for 3 different environments (Dev, Staging, and Prod) using Terraform workspaces.**

Created the terraform.tf folder and main.tf, varaible.tf, output.tf files in the folder

**Main.tf :**

# Use AWS provider in us-east-1

provider "aws" {

  region = "us-east-1"

}

# Fetch the latest Amazon Linux 2 AMI in us-east-1

data "aws\_ami" "latest\_amazon\_linux" {

  most\_recent = true

  owners      = ["amazon"]

  filter {

    name   = "name"

    values = ["amzn2-ami-hvm-2.0.\*-x86\_64-gp2"]

  }

}

# Create EC2 instance

resource "aws\_instance" "app" {

  ami           = data.aws\_ami.latest\_amazon\_linux.id

  instance\_type = "t3.micro"

  tags = {

    Name = "MyAppInstance"

  }

}

**Varaible.tf:**

variable "ami" {

  default = "ami-0d85d4f07a62e2969"  # Replace with your region-specific AMI

}

variable "instance\_type" {

  default = "t3.micro"

}

**Output.tf**

output "instance\_id" {

  value = aws\_instance.app.id

}

Now need to give command

**terraform workspace select dev --> terraform apply**

**terraform workspace select prod--> terraform apply**

**terraform workspace select staging--> terraform apply**

after this the instances will get created

