

School of Computer Science, UPES, Dehradun.

Lab 17

DevSecOps Lab

B.TECH. - V Semester

**Submitted to:**

Dr. Hitesh Kumar Sharma

**Submitted by:**

Name: Ishika Mehrotra

Batch: Devops B1

Sap Id: 500124411

Roll No.: R2142230281

**Lab Exercise 8– Terraform Multiple tfvars Files**

**Objective:**

Learn how to use multiple tfvars files in Terraform for different environments.

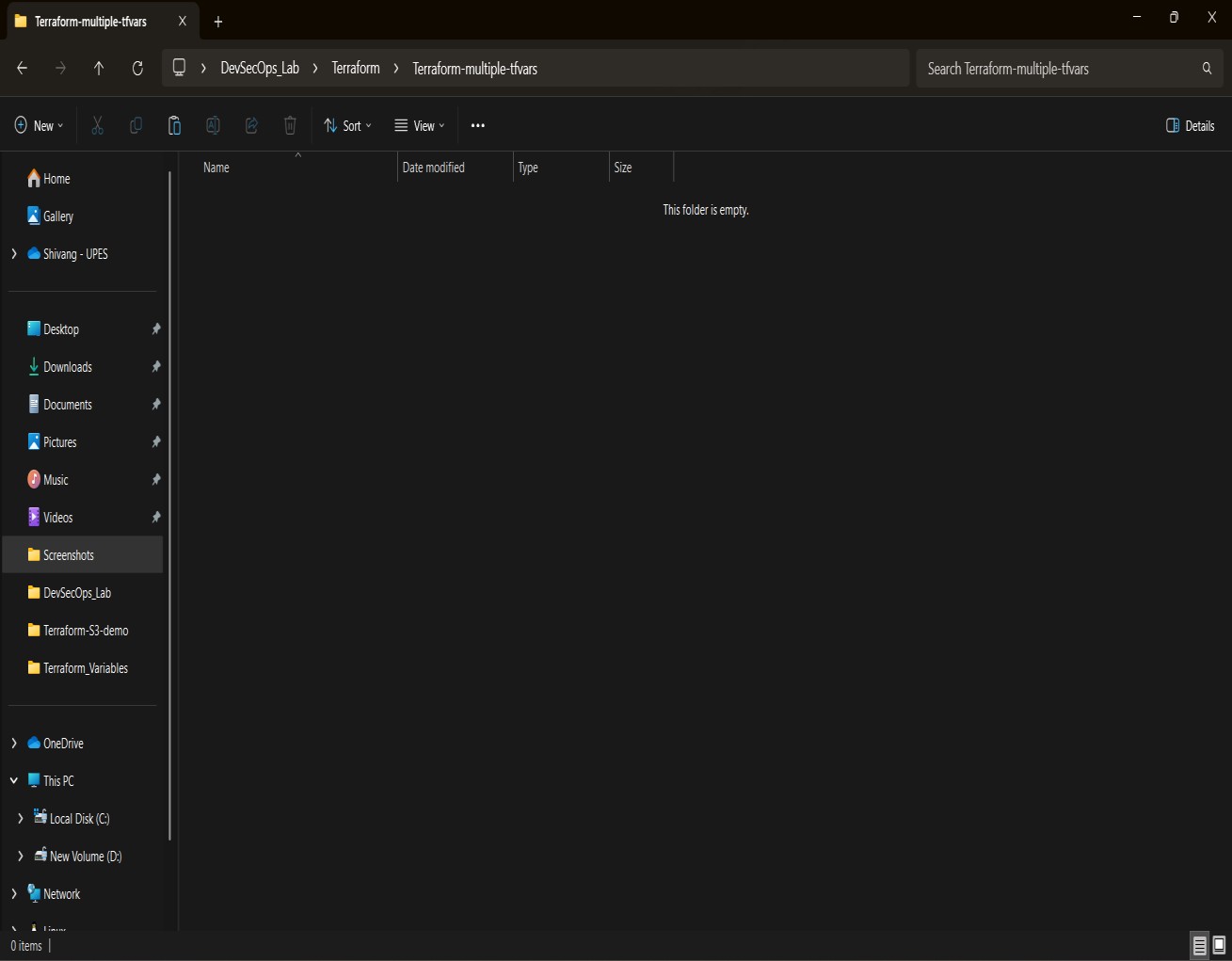
**Prerequisites:**

* Terraform installed on your machine.
* Basic knowledge of Terraform configuration and variables.

**Steps:**

1. **Create a Terraform Directory:**

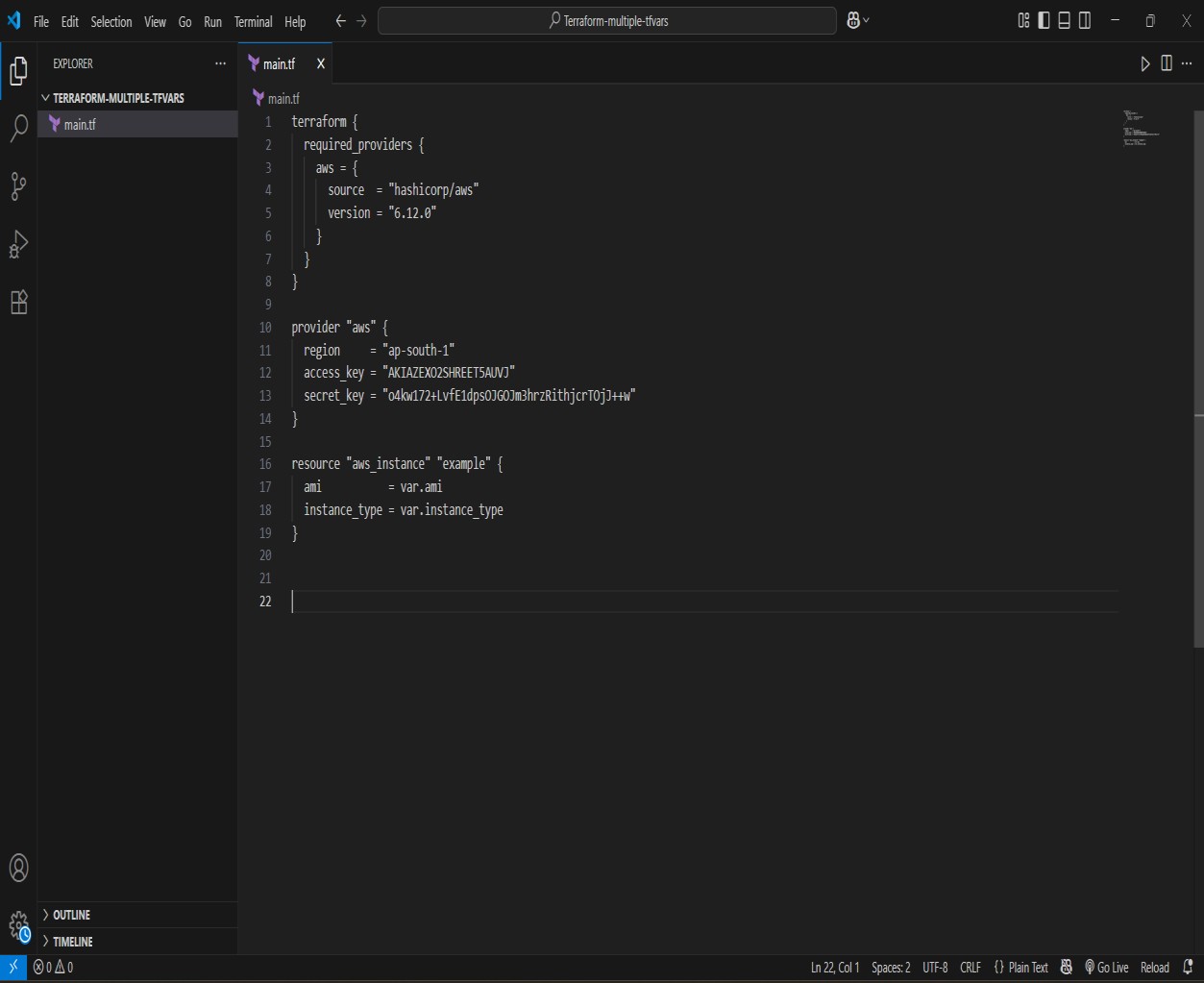
|  |  |
| --- | --- |
|  | **mkdir terraform-multiple-tfvars cd terraform-multiple-tfvars** |



* + Create Terraform Configuration Files:
  + Create a file named main.tf:

**# main.tf**

|  |
| --- |
| **provider "aws" { region = var.region**  **}**    **resource "aws\_instance" "example" {**  **ami = var.ami**  **instance\_type = var.instance\_type**  **}** |



* + Create a file named variables.tf:

**# variables.tf**

**variable "ami" {**

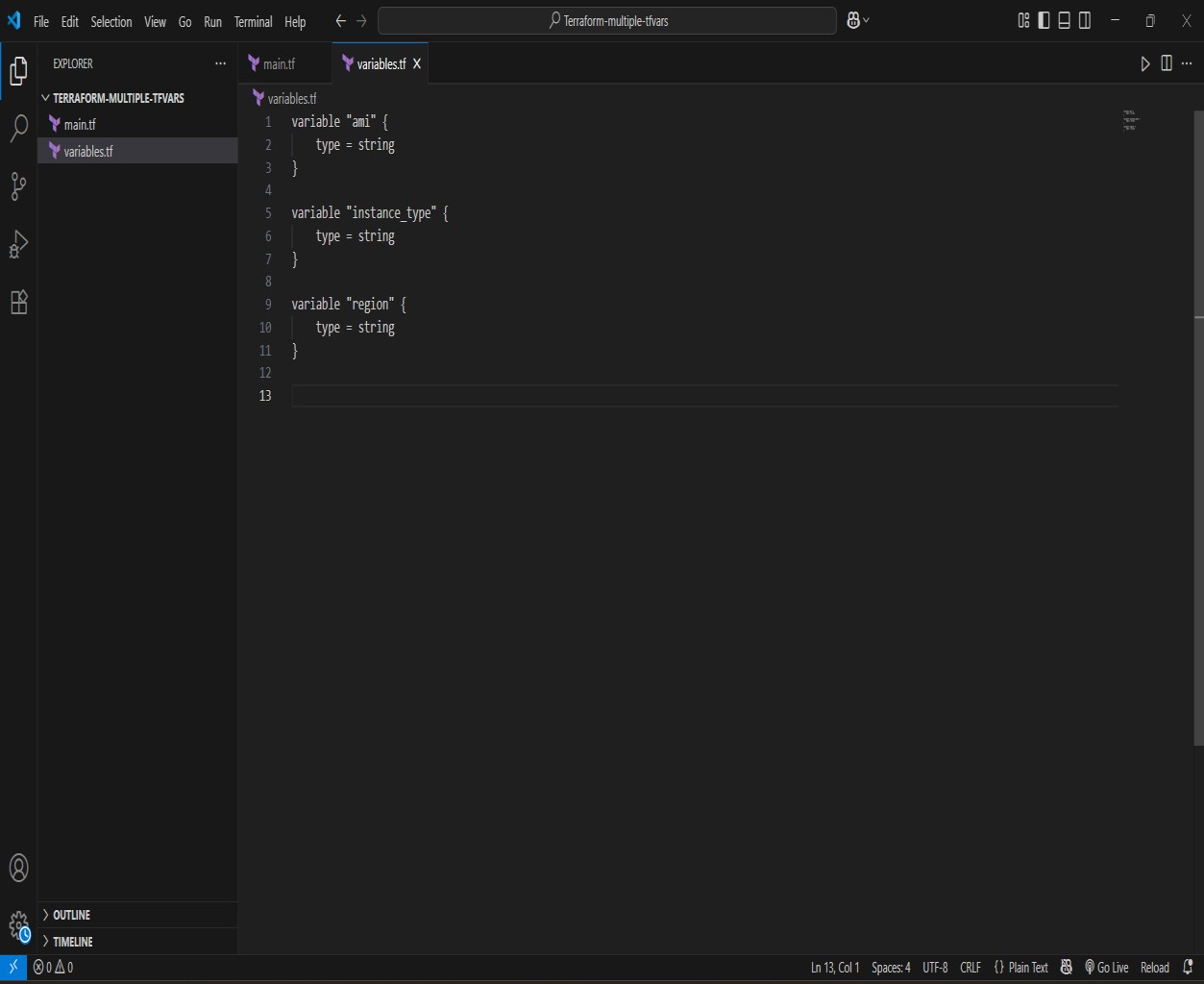
**type = string**

**}**

**variable "instance\_ty" {**

**type = string**

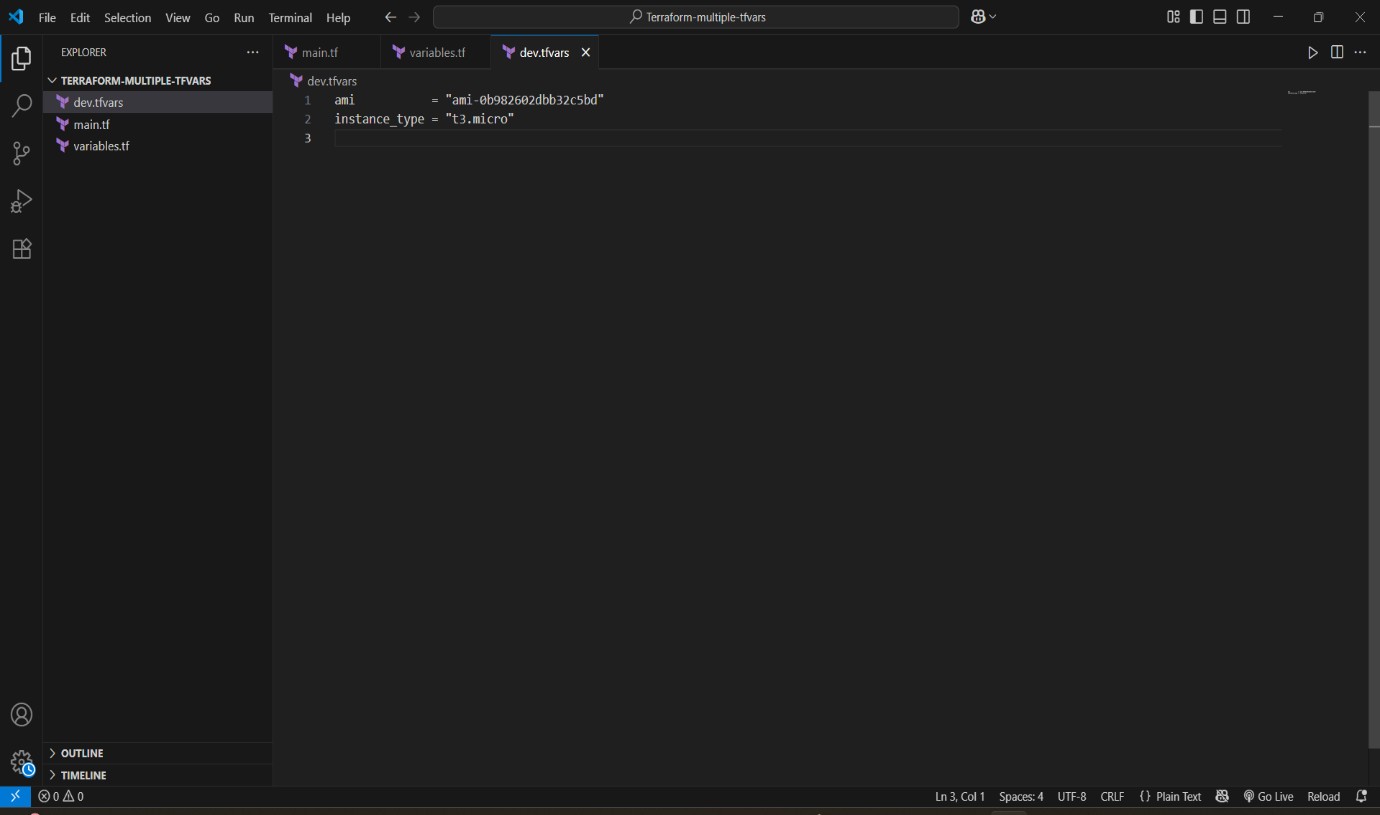
**}**



1. **Create Multiple tfvars Files:**

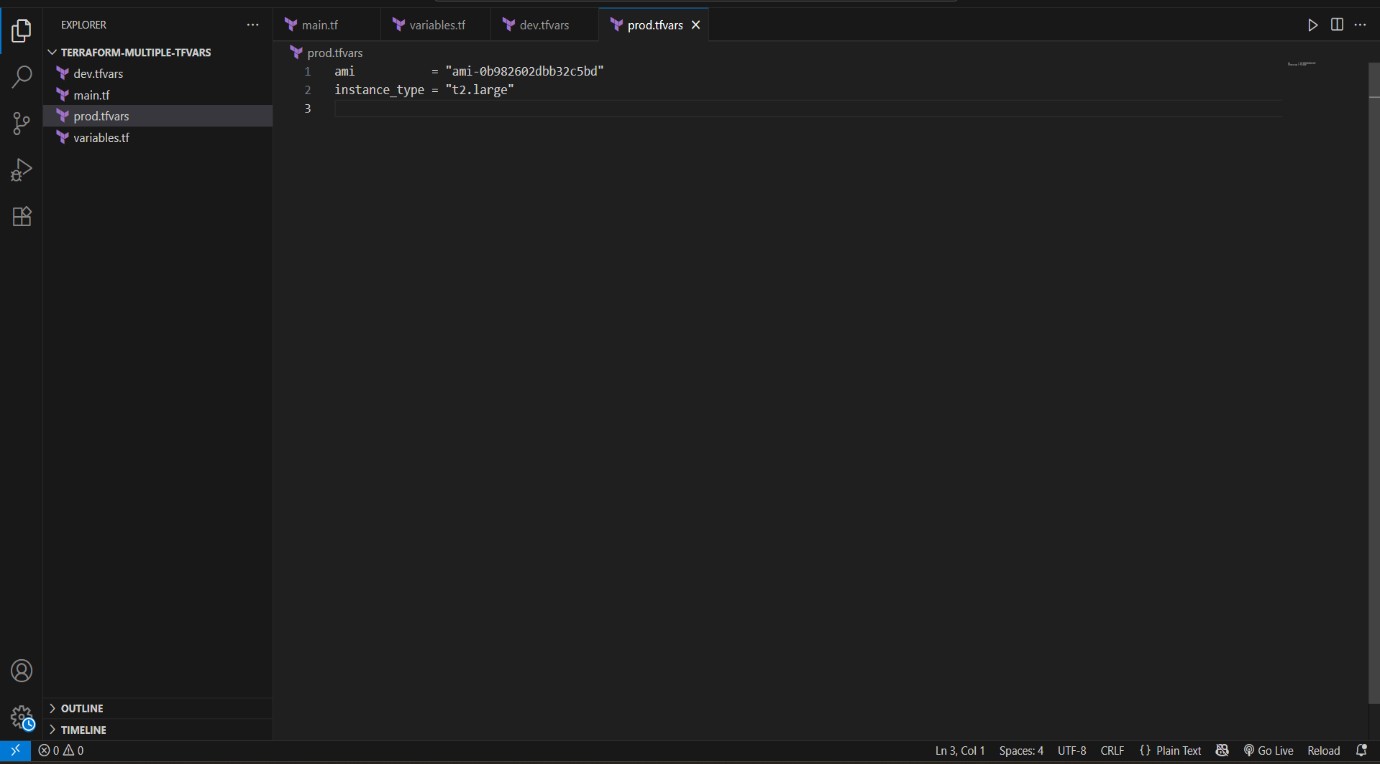
* Create a file named dev.tfvars: **# dev.tfvars**

**ami = "ami-0123456789abcdef0" instance\_type = "t2.micro"**



* Create a file named prod.tfvars: **# prod.tfvars**

**ami = "ami-9876543210fedcba0" instance\_type = "t2.large"**



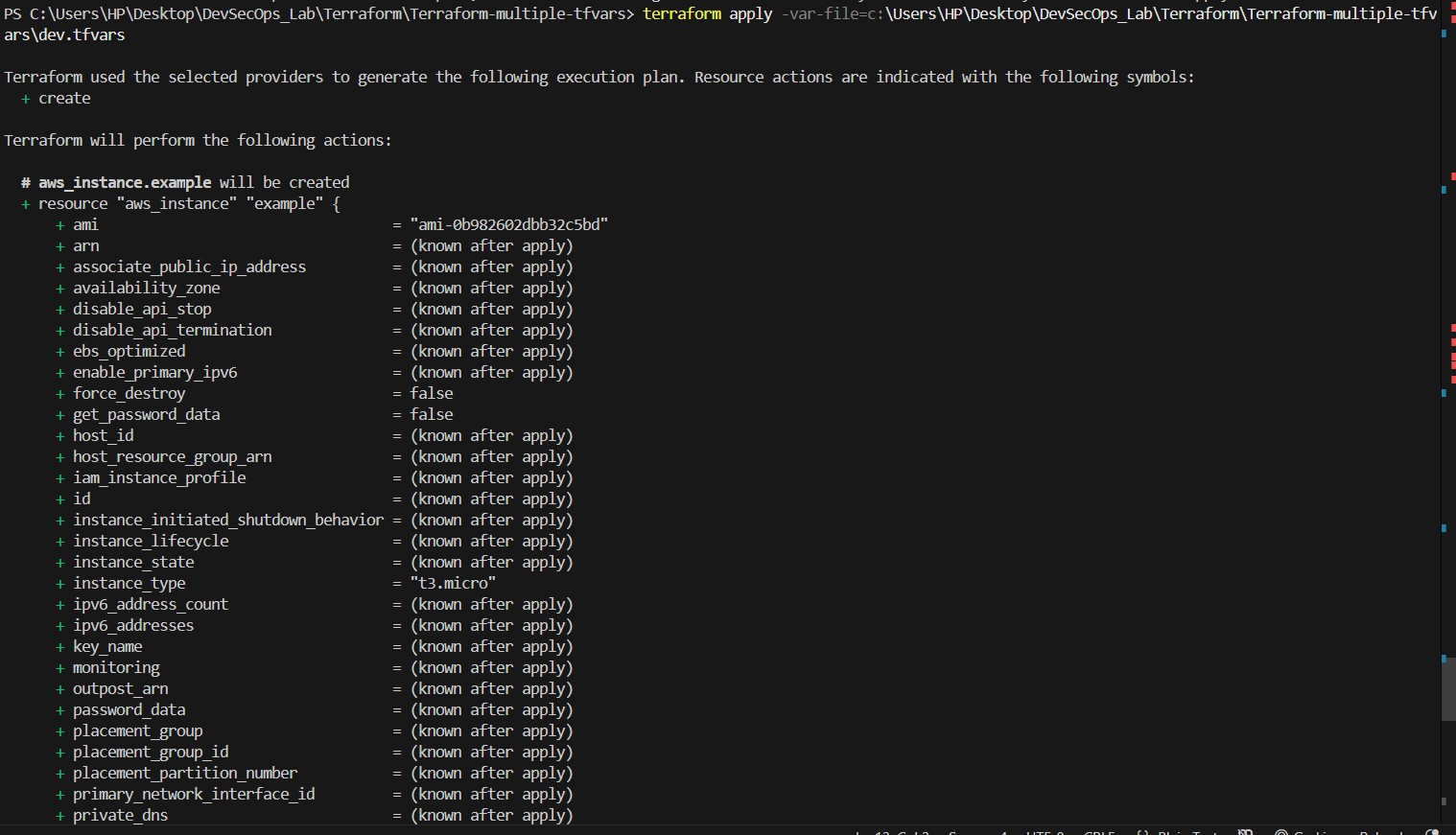
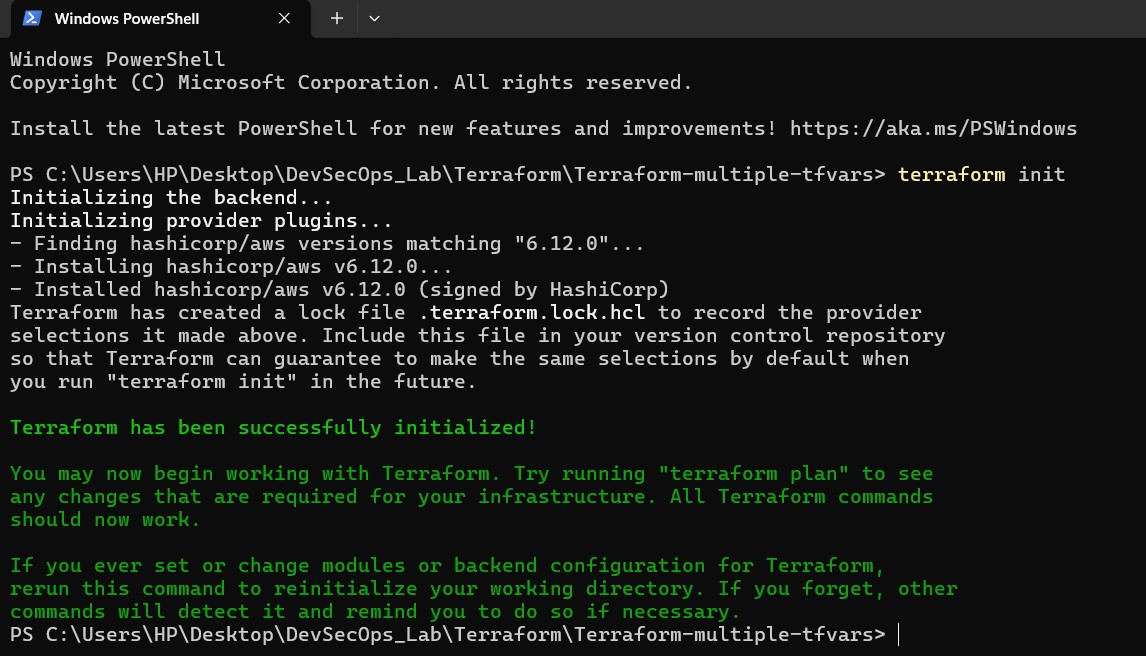
* In these files, provide values for the variables based on the environments.

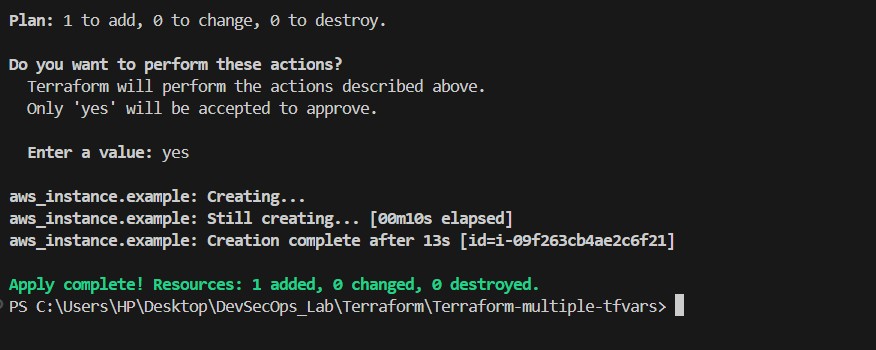
**3. Initialize and Apply for Dev Environment:**

• Run the following Terraform commands to initialize and apply the configuration for the dev environment:

**terraform init**

**terraform apply -var-file=dev.tfvars**



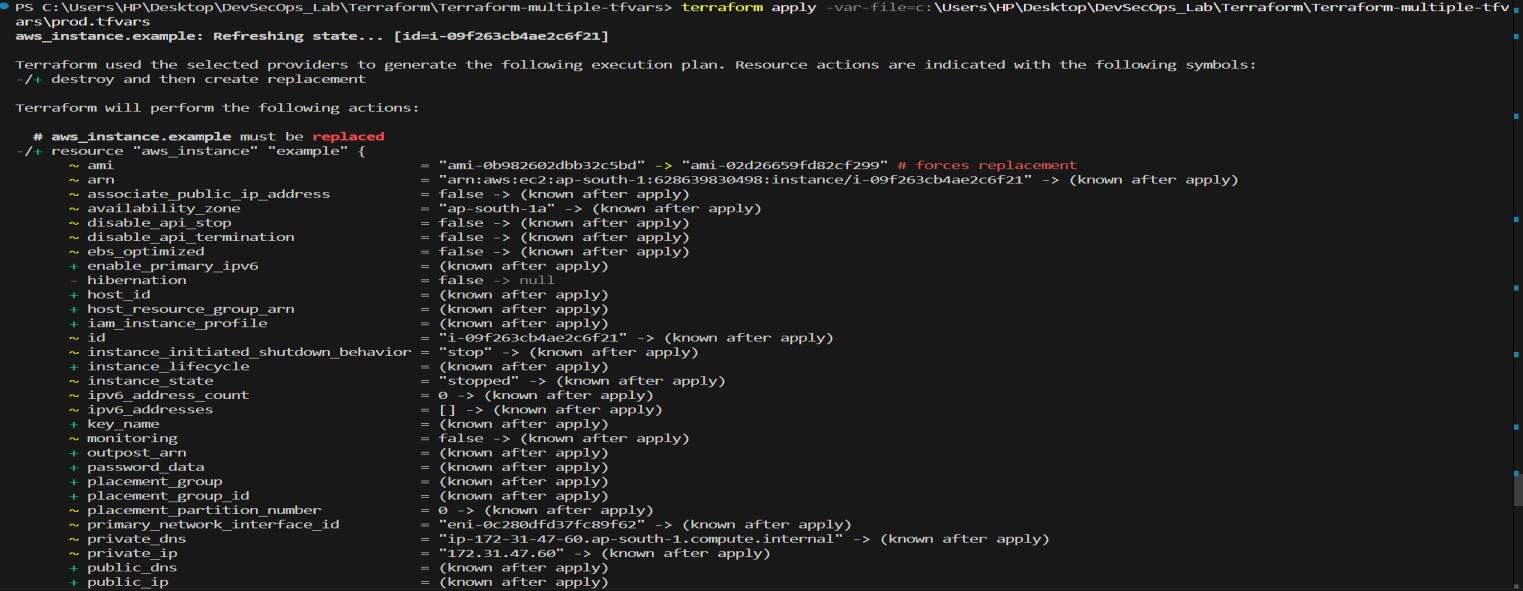
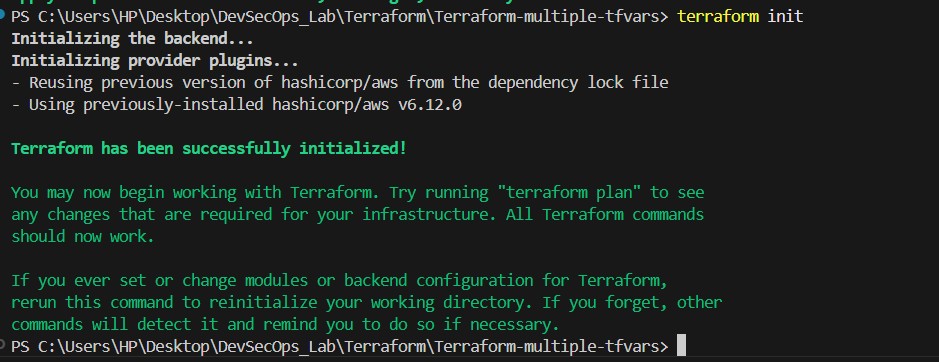


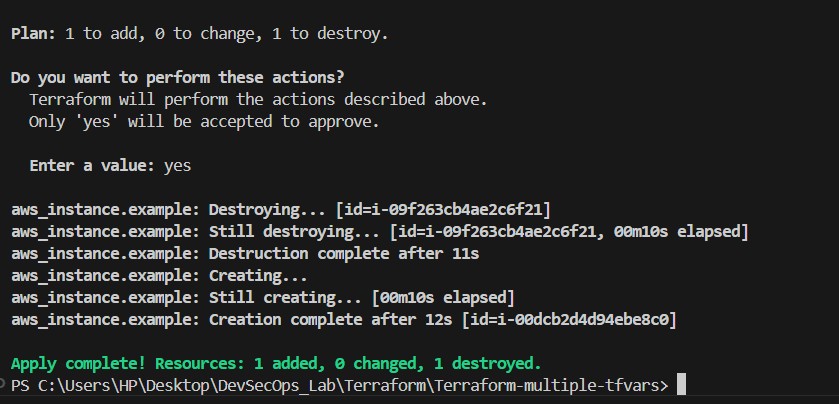
**4. Initialize and Apply for Prod Environment:**

• Run the following Terraform commands to initialize and apply the configuration for the prod environment:

**terraform init**

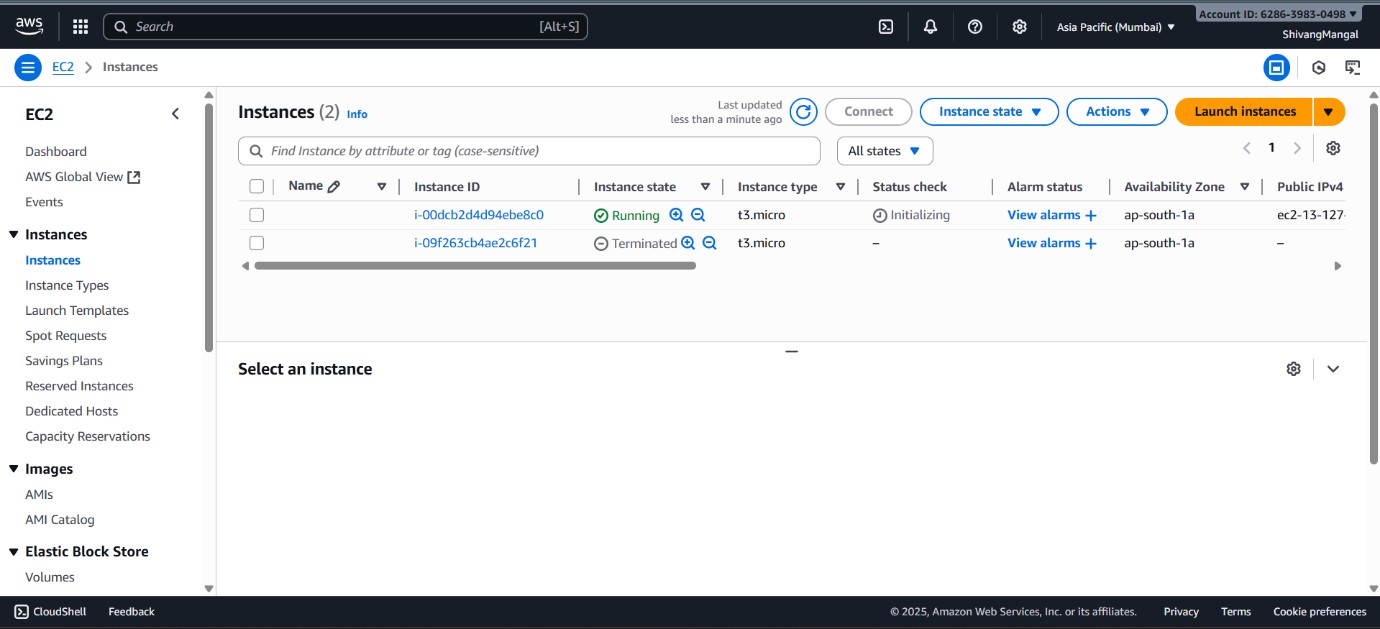
**terraform apply -var-file=prod.tfvars**





**5. Test and Verify:**

* Observe how different tfvars files are used to set variable values for different environments during the apply process.
* Access the AWS Management Console or use the AWS CLI to verify the creation of resources in the specified regions and instance types.



1. **Clean Up:** 
   * After testing, you can clean up resources:

**terraform destroy -var-file=dev.tfvars terraform destroy -var-file=prod.tfvars**

* + Confirm the destruction by typing yes.

