Lab Exercise 15 – Terraform Variables

Name:-Vansh Bhatt
SapId:- 500125395
R.No:- R2142231689
Batch:- DevOps B1
To:- Hitesh Kumar Sharma Sir

Objective:

Learn how to define and use variables in Terraform configuration.

Prerequisites:

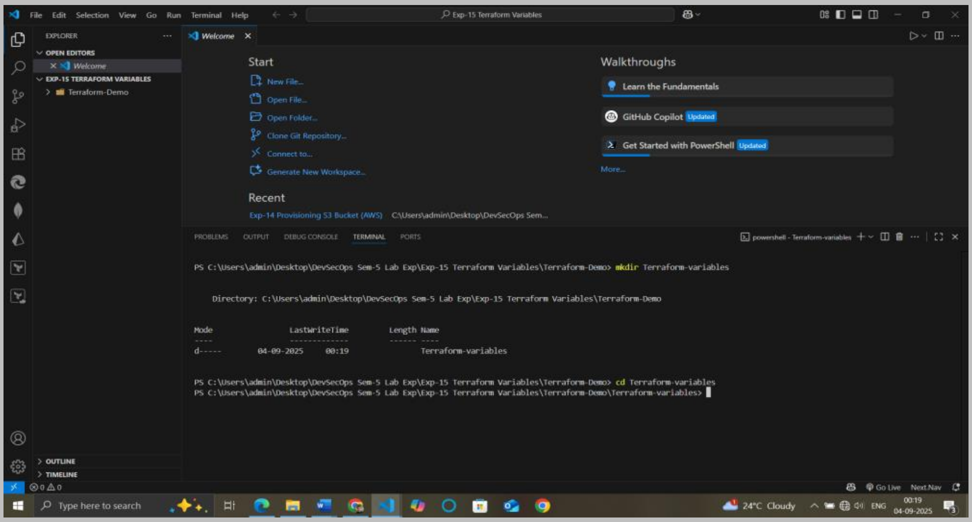
- Install Terraform on your machine.

Steps:

1. Create a Terraform Directory:

- Create a new directory for your Terraform project.

```
mkdir terraform-variables
cd terraform-variables
```



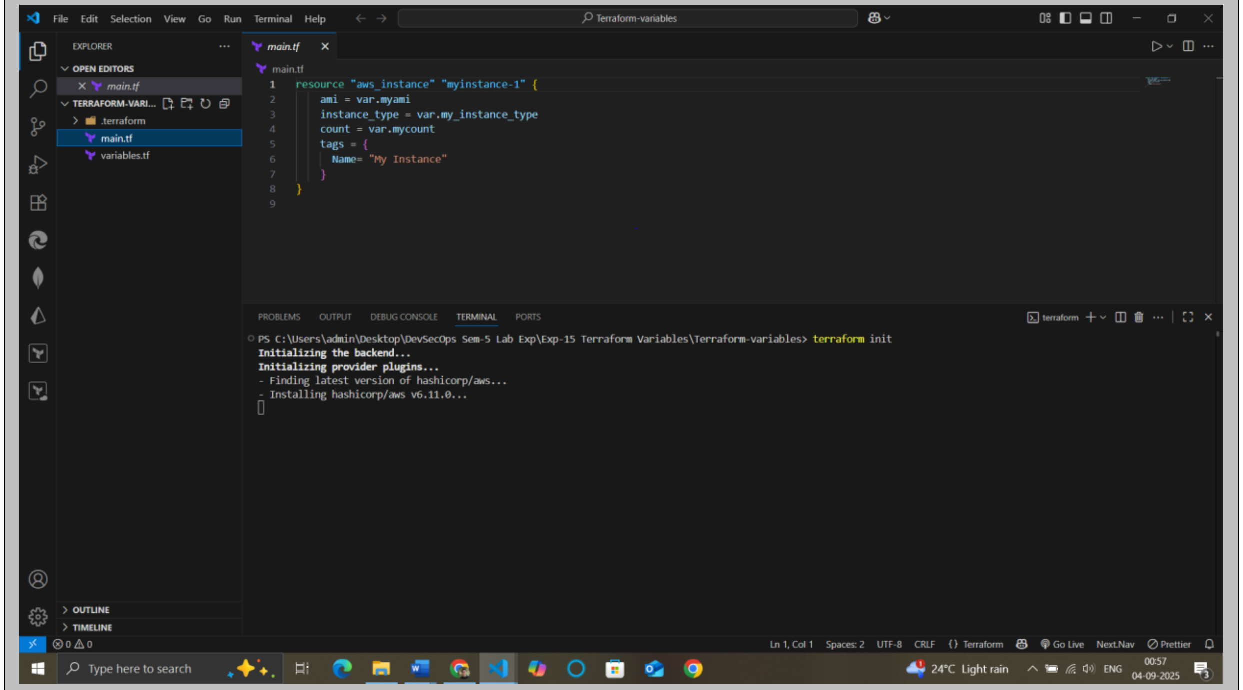
```
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-Demo> mkdir Terraform-variables


    Directory: C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-Demo


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        04-09-2025     00:19                Terraform-variables


PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-Demo> cd Terraform-variables
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-Demo\Terraform-variables>
```

2. Create a Terraform Configuration File:

- Create a file named main.tf within your project directory.

# main.tf

```
resource "aws_instance" "myinstance-1" {
  ami = var.myami
  instance_type = var.my_instance_type
  count = var.mycount
  tags = {
   Name= "My Instance"
  }
}
```



```
resource "aws_instance" "myinstance-1" {
  ami = var.myami
  instance_type = var.my_instance_type
  count = var.mycount
  tags = {
    Name= "My Instance"
  }
}
```

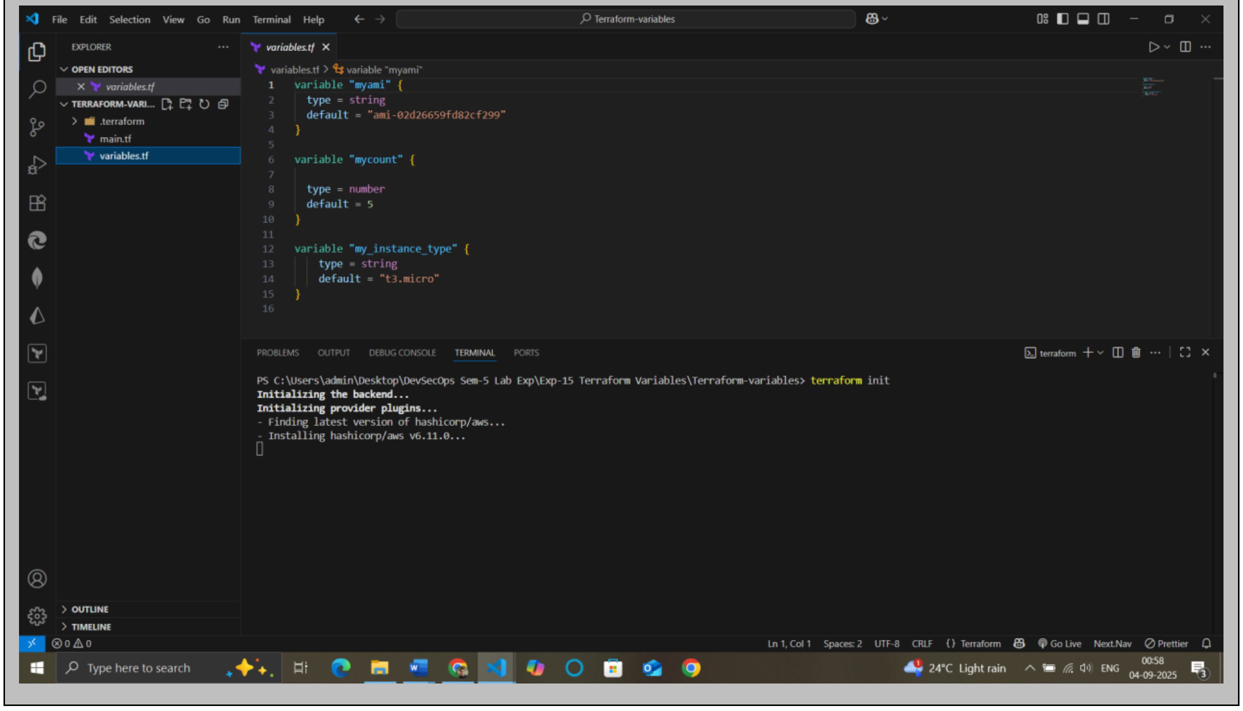
```
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.11.0...
```

3. Define Variables:

- Open a new file named variables.tf. Define variables for region, ami, and instance_type.

# variables.tf

```
variable "myami" {
  type = string
  default = "ami-08718895af4dfa033"
}

variable "mycount" {

  type = number
  default = 5
}

variable "my_instance_type" {
    type = string
    default = "t2.micro"
}
```



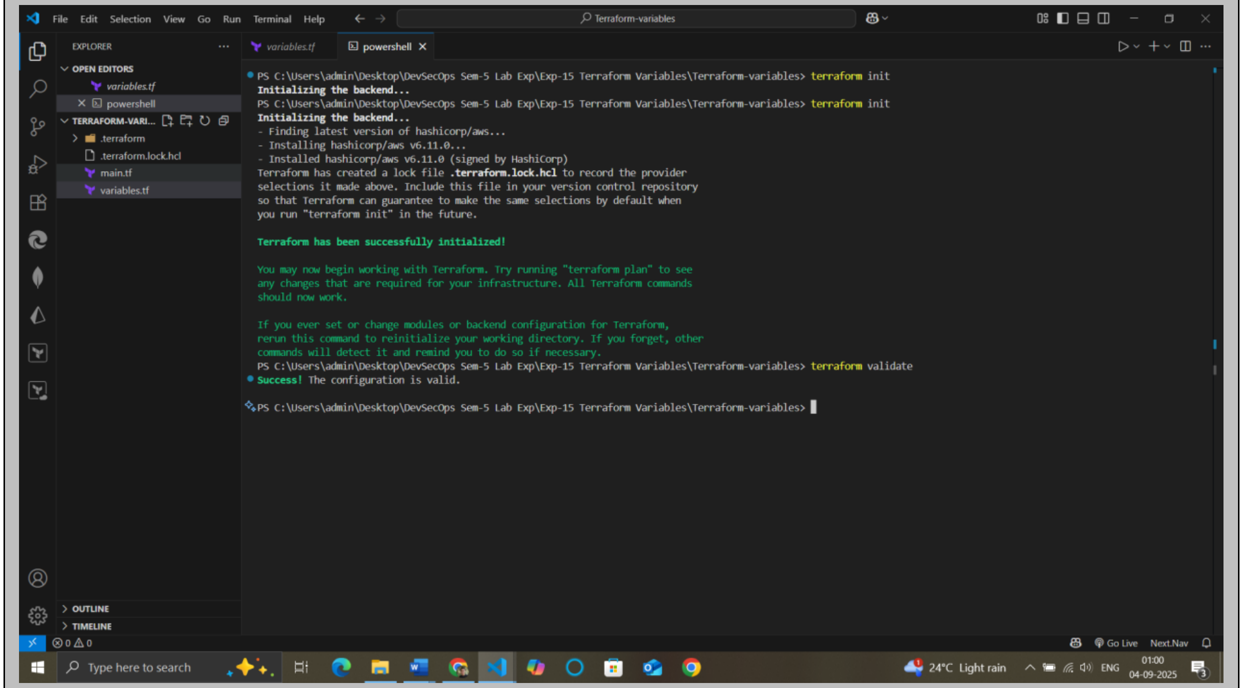
```
variable "myami" {
  type = string
  default = "ami-02d26659fd82cf299"
}

variable "mycount" {

  type = number
  default = 5
}

variable "my_instance_type" {
    type = string
    default = "t3.micro"
}
```

```
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.11.0...
```

4. Initialize and Apply:

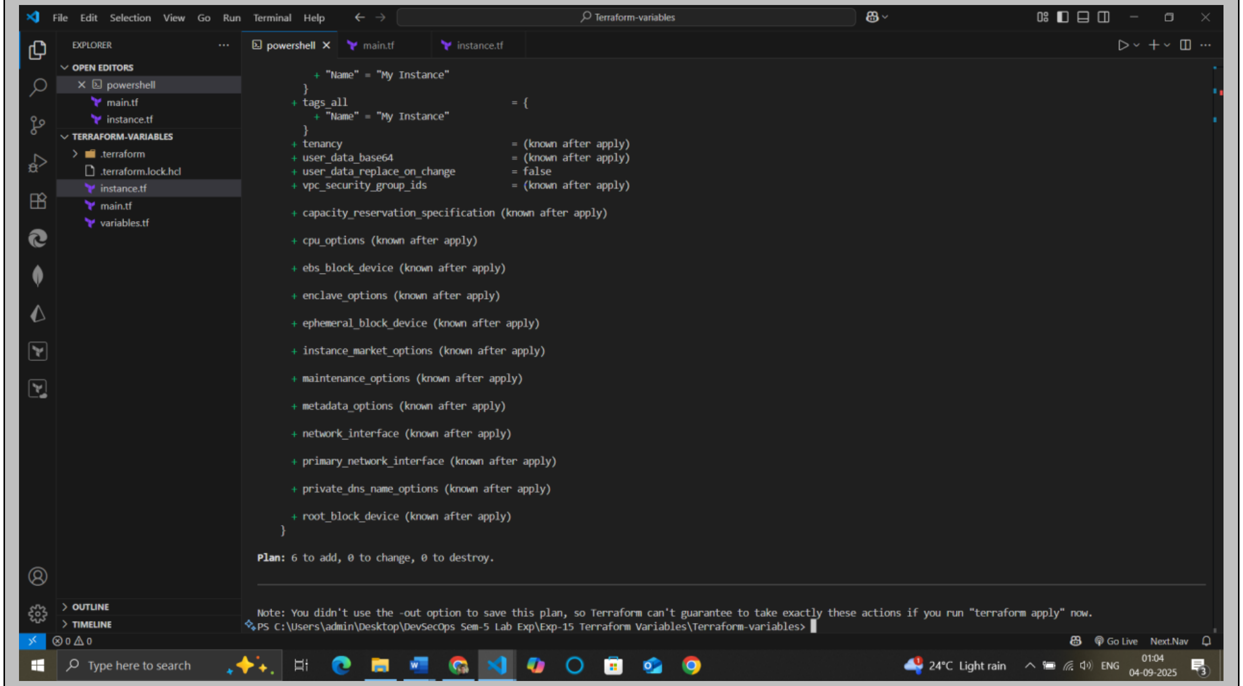
- Run the following Terraform commands to initialize and apply the configuration.

terraform init



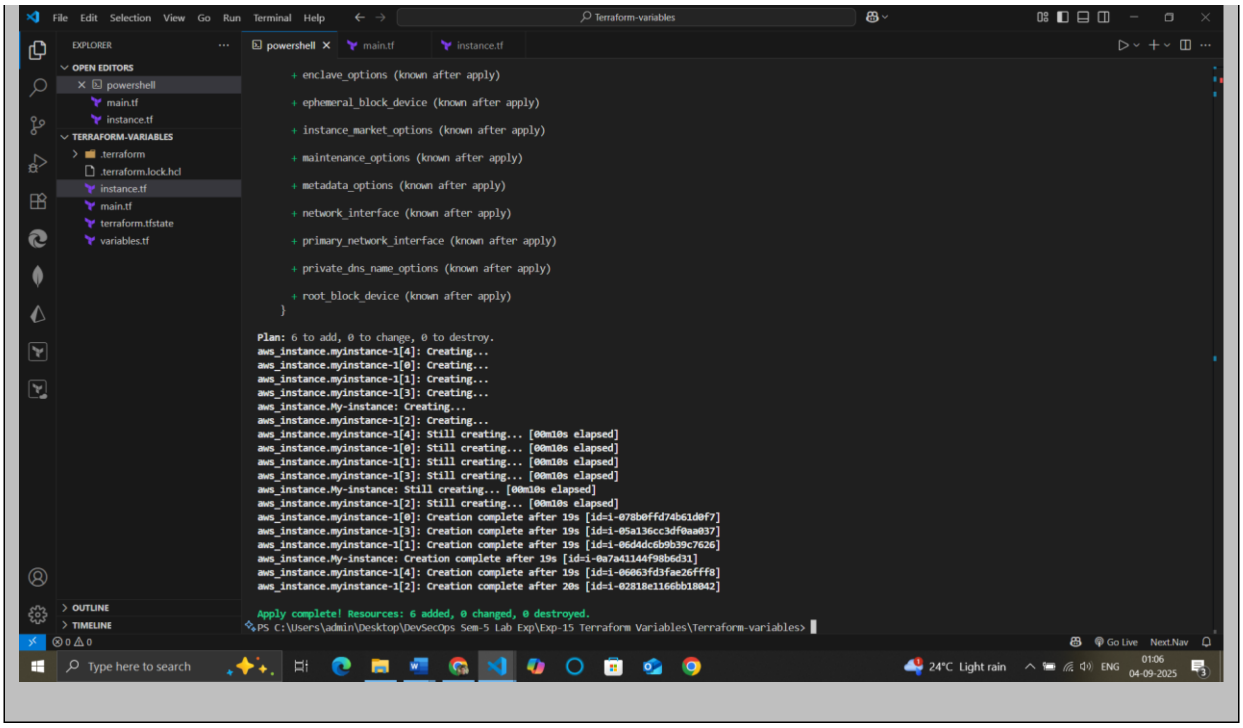
```
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables> terraform init
Initializing the backend...
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables> terraform init
Initializing the backend...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.11.0...
- Installed hashicorp/aws v6.11.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables> terraform validate
Success! The configuration is valid.

PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables>
```

terraform plan



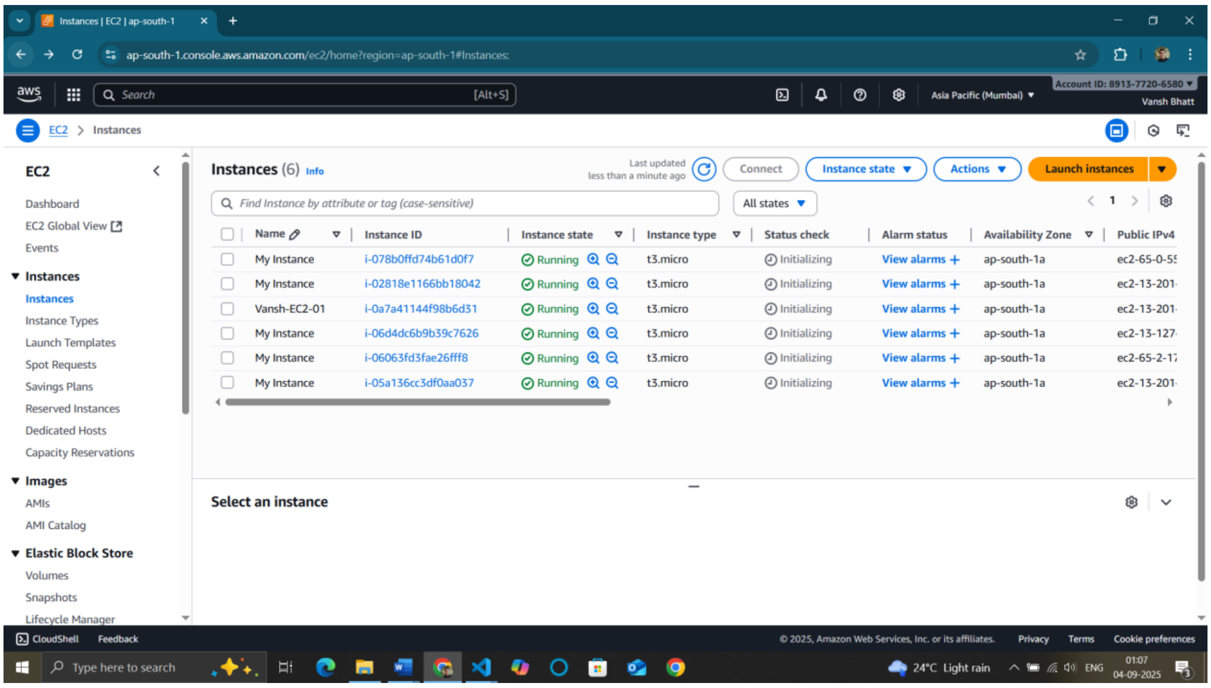
```
          + "Name" = "My Instance"
        }
      + tags_all                             = {
          + "Name" = "My Instance"
        }
      + tenancy                              = (known after apply)
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + primary_network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

Plan: 6 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables>
```

terraform apply -auto-approve



```
      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + primary_network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

Plan: 6 to add, 0 to change, 0 to destroy.
aws_instance.myinstance-1[4]: Creating...
aws_instance.myinstance-1[0]: Creating...
aws_instance.myinstance-1[1]: Creating...
aws_instance.myinstance-1[3]: Creating...
aws_instance.My-instance: Creating...
aws_instance.myinstance-1[2]: Creating...
aws_instance.myinstance-1[4]: Still creating... [00m10s elapsed]
aws_instance.myinstance-1[0]: Still creating... [00m10s elapsed]
aws_instance.myinstance-1[1]: Still creating... [00m10s elapsed]
aws_instance.myinstance-1[3]: Still creating... [00m10s elapsed]
aws_instance.My-instance: Still creating... [00m10s elapsed]
aws_instance.myinstance-1[2]: Still creating... [00m10s elapsed]
aws_instance.myinstance-1[0]: Creation complete after 19s [id=i-078b0ffd74b61d0f7]
aws_instance.myinstance-1[3]: Creation complete after 19s [id=i-05a136cc3df0aa037]
aws_instance.myinstance-1[1]: Creation complete after 19s [id=i-06d4dc6b9b39c7626]
aws_instance.My-instance: Creation complete after 19s [id=i-0a7a41144f98b6d31]
aws_instance.myinstance-1[4]: Creation complete after 19s [id=i-06063fd3fae26fff8]
aws_instance.myinstance-1[2]: Creation complete after 20s [id=i-02818e1166bb18042]

Apply complete! Resources: 6 added, 0 changed, 0 destroyed.
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables>
```

Observe how the region changes based on the variable override.



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:

Asia Pacific (Mumbai) · Account ID: 8913-7720-6580 · Vansh Bhatt

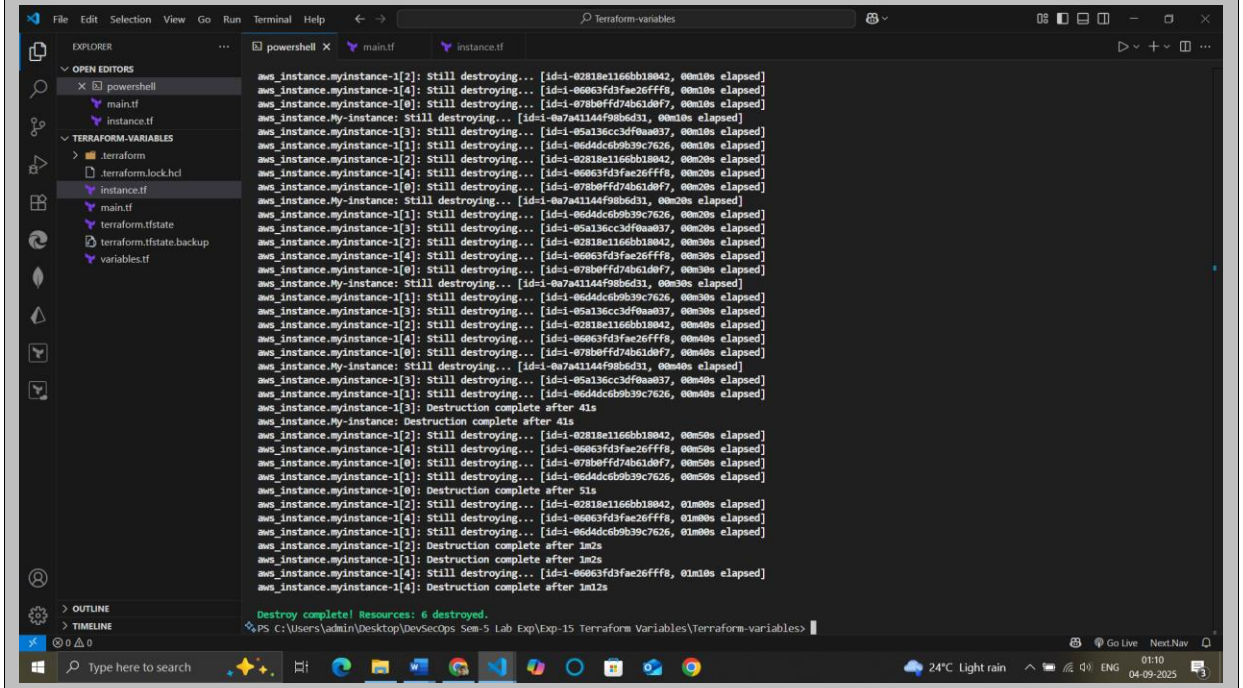
Instances (6)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
My Instance	i-078b0ffd74b61d0f7	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-65-0-5
My Instance	i-02818e1166bb18042	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-13-201
Vansh-EC2-01	i-0a7a41144f98b6d31	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-13-201
My Instance	i-06d4dc6b9b39c7626	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-13-127
My Instance	i-06063fd3fae26fff8	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-65-2-1
My Instance	i-05a136cc3df0aa037	Running	t3.micro	Initializing	View alarms +	ap-south-1a	ec2-13-201

5. Clean Up:

After testing, you can clean up resources.

terraform destroy



```
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 00m10s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 00m10s elapsed]
aws_instance.myinstance-1[0]: Still destroying... [id=i-078b0ffd74b61d0f7, 00m10s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0a7a41144f98b6d31, 00m10s elapsed]
aws_instance.myinstance-1[3]: Still destroying... [id=i-05a136cc3df0aa037, 00m10s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 00m10s elapsed]
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 00m20s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 00m20s elapsed]
aws_instance.myinstance-1[0]: Still destroying... [id=i-078b0ffd74b61d0f7, 00m20s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0a7a41144f98b6d31, 00m20s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 00m20s elapsed]
aws_instance.myinstance-1[3]: Still destroying... [id=i-05a136cc3df0aa037, 00m20s elapsed]
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 00m30s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 00m30s elapsed]
aws_instance.myinstance-1[0]: Still destroying... [id=i-078b0ffd74b61d0f7, 00m30s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0a7a41144f98b6d31, 00m30s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 00m30s elapsed]
aws_instance.myinstance-1[3]: Still destroying... [id=i-05a136cc3df0aa037, 00m30s elapsed]
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 00m40s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 00m40s elapsed]
aws_instance.myinstance-1[0]: Still destroying... [id=i-078b0ffd74b61d0f7, 00m40s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0a7a41144f98b6d31, 00m40s elapsed]
aws_instance.myinstance-1[3]: Still destroying... [id=i-05a136cc3df0aa037, 00m40s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 00m40s elapsed]
aws_instance.myinstance-1[3]: Destruction complete after 41s
aws_instance.My-instance: Destruction complete after 41s
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 00m50s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 00m50s elapsed]
aws_instance.myinstance-1[0]: Still destroying... [id=i-078b0ffd74b61d0f7, 00m50s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 00m50s elapsed]
aws_instance.myinstance-1[0]: Destruction complete after 51s
aws_instance.myinstance-1[2]: Still destroying... [id=i-02818e1166bb18042, 01m00s elapsed]
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 01m00s elapsed]
aws_instance.myinstance-1[1]: Still destroying... [id=i-06d4dc6b9b39c7626, 01m00s elapsed]
aws_instance.myinstance-1[2]: Destruction complete after 1m2s
aws_instance.myinstance-1[1]: Destruction complete after 1m2s
aws_instance.myinstance-1[4]: Still destroying... [id=i-06063fd3fae26fff8, 01m10s elapsed]
aws_instance.myinstance-1[4]: Destruction complete after 1m12s

Destroy complete! Resources: 6 destroyed.
PS C:\Users\admin\Desktop\DevSecOps Sem-5 Lab Exp\Exp-15 Terraform Variables\Terraform-variables>
```

Confirm the destruction by typing yes.

6. Conclusion:

This lab exercise introduces you to Terraform variables and demonstrates how to use them in your configurations. Experiment with different variable values and overrides to understand their impact on the infrastructure provisioning process.