Lab Exercise 6– Terraform Variables

Objective:

Learn how to define and use variables in Terraform configuration.

Prerequisites:

- Install Terraform on your machine.

Steps:

1. Create a Terraform Directory:

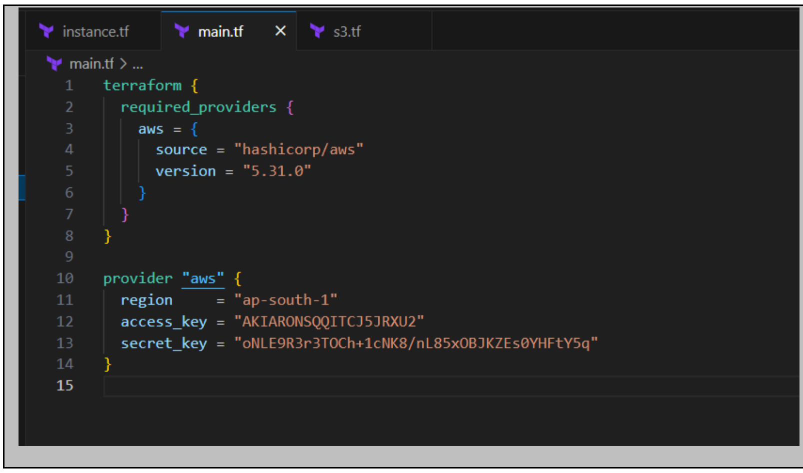
- Create a new directory for your Terraform project.

```
mkdir terraform-variables

cd terraform-variables
```

2. Create a Terraform Configuration File:

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

# main.tf

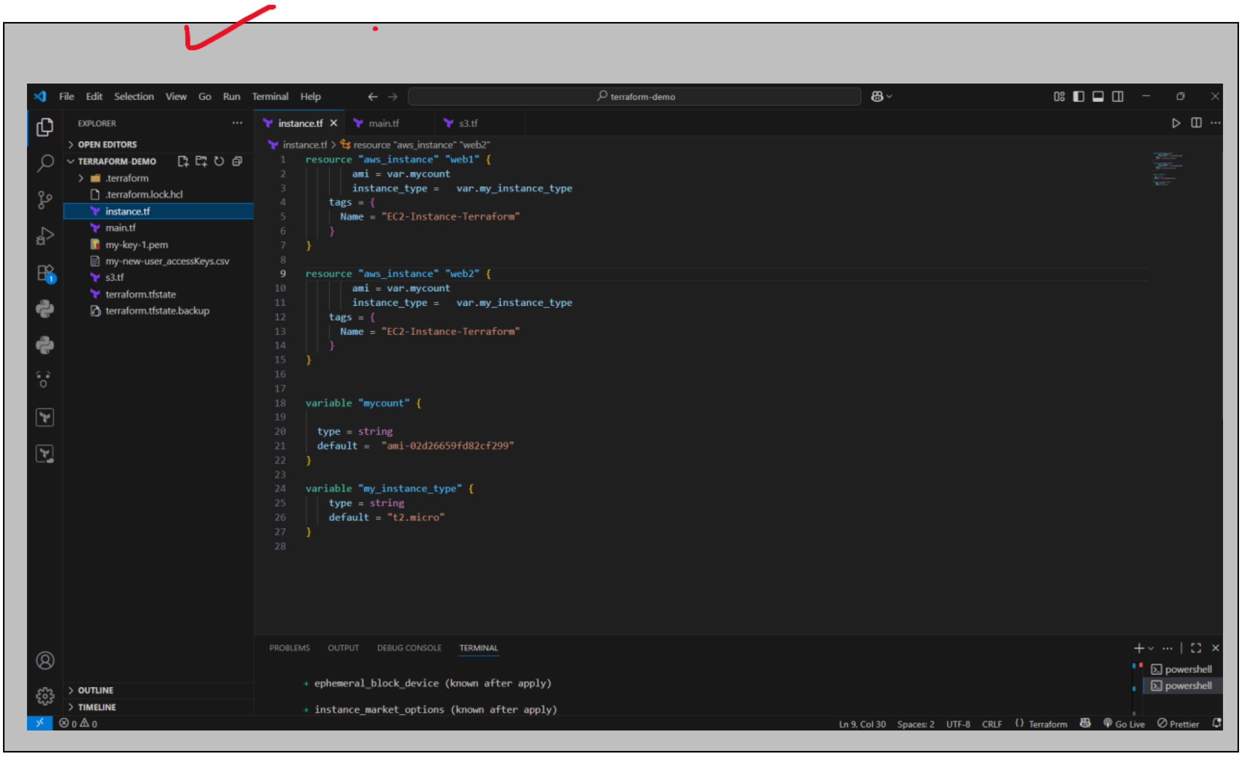


```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.31.0"
    }
  }
}

provider "aws" {
  region     = "ap-south-1"
  access_key = "AKIARONSQQITCJ5JRXU2"
  secret_key = "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"
}

```

3. Define Variables:

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

# variables.tf



```
resource "aws_instance" "web1" {
        ami = var.mycount
        instance_type =   var.my_instance_type
    tags = {
      Name = "EC2-Instance-Terraform"
    }
}

resource "aws_instance" "web2" {
        ami = var.mycount
        instance_type =   var.my_instance_type
    tags = {
      Name = "EC2-Instance-Terraform"
    }
}


variable "mycount" {

  type = string
  default =  "ami-02d26659fd82cf299"
}

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

```

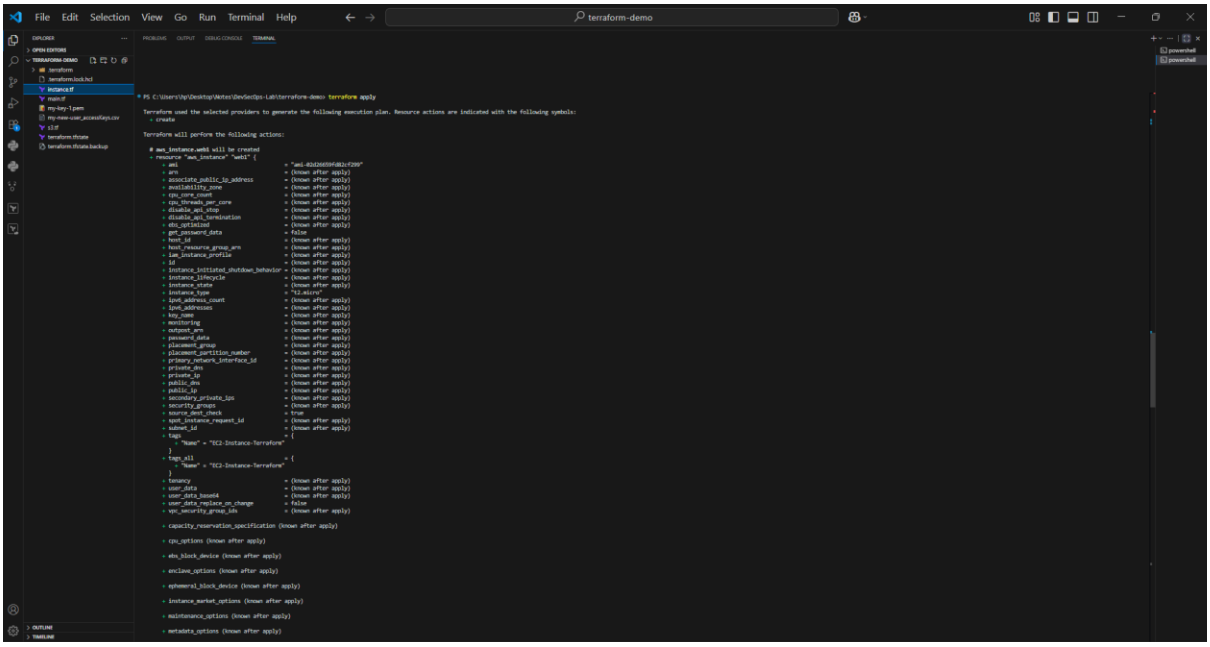
```
      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)
```

4. Initialize and Apply:

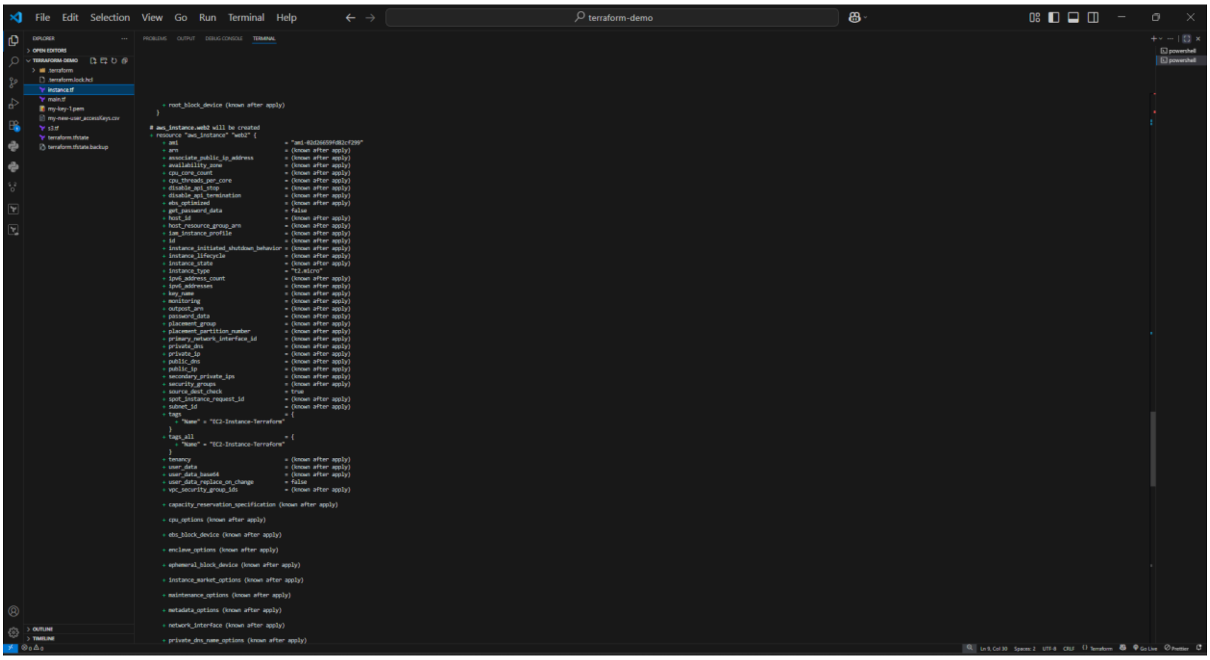
- Run the following Terraform commands to initialize and apply the configuration.

```
terraform init

terraform plan

terraform apply -auto-approve
```

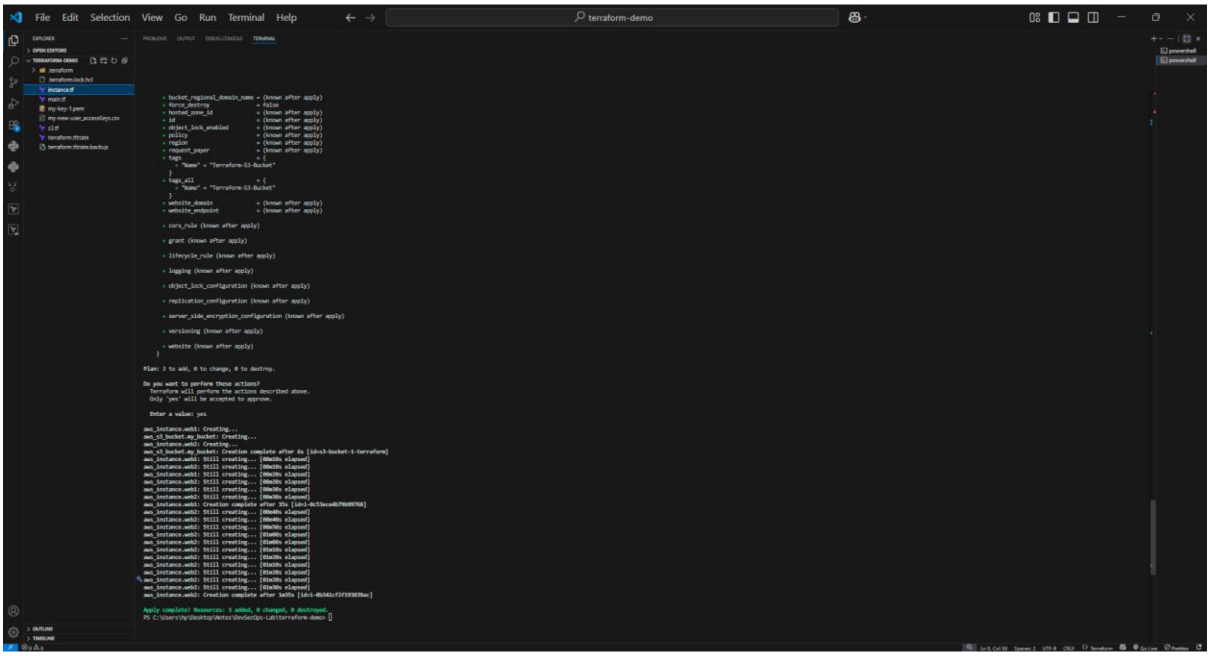
Observe how the region changes based on the variable override.



```
PS C:\Users\hp\Desktop\Notes\DevSecOps-Lab\terraform-demo> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.web1 will be created
  + resource "aws_instance" "web1" {
      + ami                                  = "ami-02d26659fd82cf299"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "EC2-Instance-Terraform"
        }
      + tags_all                             = {
          + "Name" = "EC2-Instance-Terraform"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (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)
```



```
      + root_block_device (known after apply)
    }

  # aws_instance.web2 will be created
  + resource "aws_instance" "web2" {
      + ami                                  = "ami-02d26659fd82cf299"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "EC2-Instance-Terraform"
        }
      + tags_all                             = {
          + "Name" = "EC2-Instance-Terraform"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (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)

      + private_dns_name_options (known after apply)
```



```
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags                        = {
          + "Name" = "Terraform-S3-Bucket"
        }
      + tags_all                    = {
          + "Name" = "Terraform-S3-Bucket"
        }
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)

      + cors_rule (known after apply)

      + grant (known after apply)

      + lifecycle_rule (known after apply)

      + logging (known after apply)

      + object_lock_configuration (known after apply)

      + replication_configuration (known after apply)

      + server_side_encryption_configuration (known after apply)

      + versioning (known after apply)

      + website (known after apply)
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.web1: Creating...
aws_s3_bucket.my_bucket: Creating...
aws_instance.web2: Creating...
aws_s3_bucket.my_bucket: Creation complete after 6s [id=s3-bucket-1-terraform]
aws_instance.web1: Still creating... [00m10s elapsed]
aws_instance.web2: Still creating... [00m10s elapsed]
aws_instance.web1: Still creating... [00m20s elapsed]
aws_instance.web2: Still creating... [00m20s elapsed]
aws_instance.web1: Still creating... [00m30s elapsed]
aws_instance.web2: Still creating... [00m30s elapsed]
aws_instance.web1: Creation complete after 35s [id=i-0c53ece4b79b99768]
aws_instance.web2: Still creating... [00m40s elapsed]
aws_instance.web2: Still creating... [00m40s elapsed]
aws_instance.web2: Still creating... [00m50s elapsed]
aws_instance.web2: Still creating... [01m00s elapsed]
aws_instance.web2: Still creating... [01m00s elapsed]
aws_instance.web2: Still creating... [01m10s elapsed]
aws_instance.web2: Still creating... [01m20s elapsed]
aws_instance.web2: Still creating... [01m10s elapsed]
aws_instance.web2: Still creating... [01m20s elapsed]
aws_instance.web2: Still creating... [01m20s elapsed]
aws_instance.web2: Still creating... [01m30s elapsed]
aws_instance.web2: Creation complete after 1m35s [id=i-0b341cf2f193839ac]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
PS C:\Users\hp\Desktop\Notes\DevSecOps-Lab\terraform-demo> []
```

5. Clean Up:

After testing, you can clean up resources.

```
terraform destroy
```

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.