Lab Exercise 15– 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 = "your IAM access key"
  secret_key = "your secret access key"
}
```

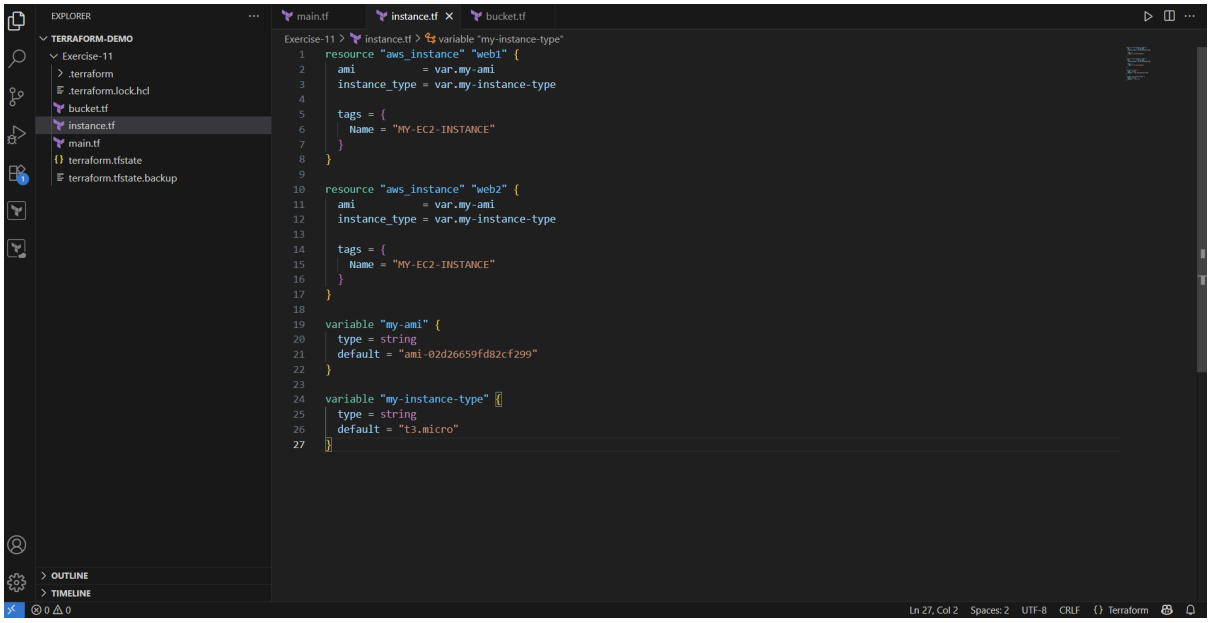
 # instance.tf

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

3. Define Variables:

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

 # v a r i a b l e s . t f



4. Initialize and Apply:

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

terraform init

terraform plan

terraform apply -auto-approve

```
C:\Me\DevSecOps\Terraform-demo\Exercise-11>terraform validate
Success! The configuration is valid.


C:\Me\DevSecOps\Terraform-demo\Exercise-11>
```

```
C:\Me\DevSecOps\Terraform-demo\Exercise-11>terraform plan

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)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + force_destroy                        = false
      + 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                        = "t3.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)
```

```
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = "ap-south-1"
      + request_payer               = (known after apply)
      + tags                        = {
          + "Name" = "My bucket"
        }
      + tags_all                    = {
          + "Name" = "My 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.

─────────────────────────────────────────────────────────────────────────────

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.

C:\Me\DevSecOps\Terraform-demo\Exercise-11>
```

```
Plan: 3 to add, 0 to change, 0 to destroy.
aws_s3_bucket.example: Creating...
aws_instance.web1: Creating...
aws_instance.web2: Creating...
aws_s3_bucket.example: Creation complete after 5s [id=demo456-my-tf-test-bucket]
aws_instance.web1: Still creating... [00m10s elapsed]
aws_instance.web2: Still creating... [00m10s elapsed]
aws_instance.web2: Creation complete after 14s [id=i-08dec419cefa85b86]
aws_instance.web1: Creation complete after 14s [id=i-0e4a118724c8b9a9e]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

C:\Me\DevSecOps\Terraform-demo\Exercise-11>
```

Observe how the region changes based on the variable override.

5. Clean Up:

After testing, you can clean up resources.

terraform destroy

Confirm the destruction by typing yes.

```
Plan: 0 to add, 0 to change, 3 to destroy.
aws_s3_bucket.example: Destroying... [id=demo456-my-tf-test-bucket]
aws_instance.web2: Destroying... [id=i-08dec419cefa85b86]
aws_instance.web1: Destroying... [id=i-0e4a118724c8b9a9e]
aws_s3_bucket.example: Destruction complete after 1s
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 00m10s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 00m10s elapsed]
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 00m20s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 00m20s elapsed]
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 00m30s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 00m30s elapsed]
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 00m40s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 00m40s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 00m50s elapsed]
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 00m50s elapsed]
aws_instance.web2: Still destroying... [id=i-08dec419cefa85b86, 01m00s elapsed]
aws_instance.web1: Still destroying... [id=i-0e4a118724c8b9a9e, 01m00s elapsed]
aws_instance.web1: Destruction complete after 1m1s
aws_instance.web2: Destruction complete after 1m1s

Destroy complete! Resources: 3 destroyed.

C:\Me\DevSecOps\Terraform-demo\Exercise-11>
```

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.