**Using Terraform Module to Deploy Multiple Webservers**

1. Create a new directory for your Terraform project and create a main.tf file.

mkdir terraform-module-ec2 && cd terraform-module-ec2

touch main.tf

1. Create a new directory named modules in your project directory, and inside it, create a folder named ec2\_instance

mkdir -p modules/ec2\_instance

1. Create the following files inside the modules/ec2\_instance directory.

variables.tf, output.tf, and main.tf

touch modules/ec2\_instance/variables.tf modules/ec2\_instance/outputs.tf modules/ec2\_instance/main.tf

1. Edit modules/ec2\_instance/variable.tf to define input variables for the module.

variable "instance\_name" {

description = "Name of the EC2 instance"

type = string

}

variable "ami\_id" {

description = "AMI ID for the EC2 instance"

type = string

}

variable "instance\_type" {

description = "EC2 instance type"

type = string

default = "t2.micro"

}

variable "instance\_count" {

description = "Number of EC2 instances to create"

type = number

}

1. Edit modules/ec2\_instance/main.ft to create the EC2 instances

resource "aws\_instance" "webserver" {

count = var.instance\_count

ami = var.ami\_id

instance\_type = var.instance\_type

tags = {

Name = "${var.instance\_name}-${count.index + 1}"

}

}

1. Edit modules/ec2\_instance/output.ft to define the module’s output variables

output "instance\_ids" {

value = aws\_instance.webserver[\*].id

}

output "public\_ips" {

value = aws\_instance.webserver[\*].public\_ip

}

1. Edit the main.tf file in your project directory to use the EC2 module

provider "aws" {

region = "us-east-1"

}

module "ec2\_instances" {

source = "./modules/ec2\_instance"

ami\_id = "ami-007855ac798b5175e" # Amazon Linux 2 AMI

instance\_count = 10

instance\_name = "webserver"

instance\_type = "t2.micro"

}

output "instance\_ids" {

value = module.ec2\_instances.instance\_ids

}

output "public\_ips" {

value = module.ec2\_instances.public\_ips

}

1. Initialize Terraform

Terraform init

1. Apply the Terraform configuration

Terraform apply

(view the results)

1. End the session

Terraform destroy