Week 2 Assignment 2

Project Link – https://github.com/shubhamsharma11/TerraformProjects

Main.tf

```
provider "azurerm" {
      features {}
}

# Create a resource group
resource "azurerm_resource_group" "rg01" {
  name     = var.rg["name"]
  location = var.rg["location"]
}

resource "azurerm_virtual_network" "vnet02" {
  name                = var.vnet_name
  resource_group_name = azurerm_resource_group.rg01.name
  location            = azurerm_resource_group.rg01.location
  address_space       = ["10.0.0.0/16"]
}


resource "azurerm_subnet" "example" {
  name                 = var.subnet["name"]
  resource_group_name  = azurerm_resource_group.rg01.name
  virtual_network_name = azurerm_virtual_network.vnet02.name
  address_prefixes     = [var.subnet["CIDR"]]
}

resource "azurerm_public_ip" "example" {
  name                = var.public_ip["name"]
  resource_group_name = azurerm_resource_group.rg01.name
  location            = azurerm_resource_group.rg01.location
  allocation_method   = var.public_ip["allocation_method"]

  tags = {
    environment = var.env
  }
}

resource "azurerm_network_interface" "main" {
  name                = var.nic_name
  location            = azurerm_resource_group.rg01.location
  resource_group_name = azurerm_resource_group.rg01.name

  ip_configuration {
    name                          = var.nic_ip_config["name"]
    subnet_id                     = azurerm_subnet.example.id
    private_ip_address_allocation = var.nic_ip_config["ip_allocation_type"]
    public_ip_address_id          = azurerm_public_ip.example.id
  }
}

resource "azurerm_virtual_machine" "main" {
  name                  = var.vm["name"]
  location              = azurerm_resource_group.rg01.location
  resource_group_name   = azurerm_resource_group.rg01.name
  network_interface_ids = [azurerm_network_interface.main.id]
  vm_size               = var.vm["size"]

  # Uncomment this line to delete the OS disk automatically when deleting the VM
  # delete_os_disk_on_termination = true

  # Uncomment this line to delete the data disks automatically when deleting the VM
  # delete_data_disks_on_termination = true

  storage_image_reference {
    publisher = var.storage_image_reference["publisher"]
    offer     = var.storage_image_reference["offer"]
    sku       = var.storage_image_reference["sku"]
    version   = var.storage_image_reference["version"]
  }
  storage_os_disk {
    name              = var.storage_os_disk["name"]
    caching           = var.storage_os_disk["caching"]
    create_option     = var.storage_os_disk["create_option"]
    managed_disk_type = var.storage_os_disk["managed_disk_type"]
  }
  os_profile {
    computer_name  = var.connection["hostname"]
    admin_username = var.connection["username"]
    admin_password = var.connection["password"]
  }
  os_profile_linux_config {
    disable_password_authentication = false
  }
  tags = {
    environment = "staging"
  }
}

resource "null_resource" "copy-file" {

  triggers = {
    always_run = timestamp()
  }
  provisioner "file" {
    source = var.file_copy["srource"]
    destination = var.file_copy["destination"]

    connection {
      type = var.connection["type"]
      user = var.connection["username"]
      password = var.connection["password"]
      host = azurerm_public_ip.example.ip_address
    }
  }
}

resource "null_resource" "remote-command" {

  triggers = {
    always_run = timestamp()
  }

  provisioner "remote-exec" {
    #scripts = "[cp test1.txt test2.txt]"
    inline = [
      "${var.remote_exec_cmd}"
     ]

    connection {
      type = var.connection["type"]
      user = var.connection["username"]
      password = var.connection["password"]
      host = azurerm_public_ip.example.ip_address
    }
  }
  depends_on = [ null_resource.copy-file ]
}
```

Variable.tf

```
variable "region" {
  default = "East US"
}

variable "env" {
  type = string
  default = "Production"
}

variable "rg" {
  type = map
  default = {
    "name" = "rg_test1"
    "location" = "East US"
  }
}
variable "subnet" {
  type = map
  default =  {
    "name" = "sub1"
    "CIDR" = "10.0.1.0/24"
  }
}

variable "public_ip" {
  type         = map
  default      = {
    "name" = "acceptanceTestPublicIp1"
    "allocation_method" = "Static"
  }
}

variable "nic_name" {
  type         = string
  default      = "nic1"
}

variable "vm" {
  type = map
  default =  {
  "name" = "vm01"
  "size" = "Standard_DS1_v2"
}

variable "nic_ip_config" {
  type         = map
  default      = {
    "name" = "testconfiguration1"
    "ip_allocation_type" = "Dynamic"
  }
}

variable "connection" {
  type         = map
  default      = {
    "hostname" = "hostname"
    "type" = "ssh"
    "username" = "testadmin"
    "password" = "Password1234!"
  }
}

variable "file_copy" {
  type         = map
  default      = {
    "source" = "test1.txt"
    "destination" = "test1.txt"
  }
}

variable "remote_exec_cmd" {
  type        = string
  default     = "cp test1.txt test2.txt"
}

variable "storage_image_reference" {
  type        = map
  default     = {
    "publisher" = "Canonical"
    "offer" = "0001-com-ubuntu-server-jammy"
    "sku" = "22_04-lts"
    "version"   = "latest"
  }
}

variable "storage_os_disk" {
  type        = map
  default     = {
    "name" = "myosdisk1"
    "caching" = "ReadWrite"
    "create_option" = "FromImage"
    "managed_disk_type" = "Standard_LRS"
  }
}
```