# Week 2 Assignment 1

File Path - [https://github.com/shubhamsharma11/TerraformProjects](https://github.com/shubhamsharma11/TerraformProjects/raw/main/Week%202%20-%20Assignment%201.zip)

# main.tf

provider "azurerm" {

features {}

}

# Create a resource group

resource "azurerm\_resource\_group" "rg04" {

name = var.rg\_name

location = "East US"

}

# Create a virtual network within the resource group

resource "azurerm\_virtual\_network" "vnet02" {

name = var.vnet\_name

resource\_group\_name = azurerm\_resource\_group.rg04.name

location = azurerm\_resource\_group.rg04.location

address\_space = ["10.0.0.0/16"]

}

resource "azurerm\_public\_ip" "publicip01" {

name = "publiciptest01"

location = azurerm\_resource\_group.rg04.location

resource\_group\_name = azurerm\_resource\_group.rg04.name

allocation\_method = "Static"

}

resource "azurerm\_network\_interface" "nic01" {

name = "nic\_test\_01"

location = azurerm\_resource\_group.rg04.location

resource\_group\_name = azurerm\_resource\_group.rg04.name

ip\_configuration {

name = "internal"

subnet\_id = azurerm\_subnet.subnet01[0].id

private\_ip\_address\_allocation = "Dynamic"

public\_ip\_address\_id = azurerm\_public\_ip.publicip01.id

}

}

resource "azurerm\_subnet" "subnet01" {

count = length(var.subnet\_ip)

name = var.subnet\_name[count.index]

resource\_group\_name = azurerm\_resource\_group.rg04.name

virtual\_network\_name = azurerm\_virtual\_network.vnet02.name

address\_prefixes = [var.subnet\_ip[count.index]]

}

# ---------------------------------------------------------------------------

# This line is to follow company policy as boot diagnostics should be enabled

/\*Create a storage account to create blob storage for the boot diag output\*/

resource "azurerm\_storage\_account" "diagSA01" {

name = "bootdiagsa021220232"

resource\_group\_name = azurerm\_resource\_group.rg04.name

location = azurerm\_resource\_group.rg04.location

account\_tier = "${element(split("\_", var.boot\_diagnostics\_sa\_type),0)}"

account\_replication\_type = "${element(split("\_", var.boot\_diagnostics\_sa\_type),1)}"

}

# ---------------------------------------------------------------------------

resource "azurerm\_virtual\_machine" "vm01" {

name = "vm\_test\_01"

location = azurerm\_resource\_group.rg04.location

resource\_group\_name = azurerm\_resource\_group.rg04.name

network\_interface\_ids = [azurerm\_network\_interface.nic01.id]

vm\_size = "Standard\_DS1\_v2"

# ---------------------------------------------------------------------------

# This line is to follow company policy as boot diagnostics should be enabled

boot\_diagnostics {

enabled = "true"

storage\_uri = azurerm\_storage\_account.diagSA01.primary\_blob\_endpoint

}

# ---------------------------------------------------------------------------

# 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 = "Canonical"

offer = "0001-com-ubuntu-server-jammy"

sku = "22\_04-lts"

version = "latest"

}

storage\_os\_disk {

name = "myosdisk1"

caching = "ReadWrite"

create\_option = "FromImage"

managed\_disk\_type = "Standard\_LRS"

}

os\_profile {

computer\_name = "hostname"

admin\_username = "testadmin"

admin\_password = "Password1234!"

}

os\_profile\_linux\_config {

disable\_password\_authentication = true

ssh\_keys {

path = "/home/testadmin/.ssh/authorized\_keys"

key\_data = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCywo3fhW1eU1NCRRgTsHHbUlwJKQNK5RAl56cwK1YmySgt3ZKNd31q0C5UFV+z1bwGNHi0bANr1LTXxkngitiZ4mfitiT/dmhe24qVGdc9dgVXRmSH1wEeiJTQ6IN/dBRtLBbV5cNrZNGsksVqrGoEzgEHHDdQYpohPL0HBhqvBBNwCs/X6IagH1T0uLm4OihCWc1FmDTMxgPytj8t6VNzXNuH2ubq/s2fhY2yIo2AsGeJXXQc4o2yipwWC0Zg+h406zS/mKRTmGqXarTJos2e8hPB4FwRAI37yKb+lqgGUs3wPnb+m+MMMw1iRvPreZnCVDEh2jwCY0CXn/phGggz vmuser@CNative"

}

}

}

resource "null\_resource" "copy-file" {

provisioner "file" {

source = "text1.txt"

destination = "text1.txt"

connection {

type = "ssh"

user = "testadmin"

private\_key= "${file("C:\\Users\\VMUser\\.ssh\\id\_rsa1")}"

host = azurerm\_public\_ip.publicip01.ip\_address

}

}

}

resource "null\_resource" "remote-command" {

provisioner "remote-exec" {

inline = [

"cp text1.txt text2.txt"

]

connection {

type = "ssh"

user = "testadmin"

private\_key = "${file("C:\\Users\\VMUser\\.ssh\\id\_rsa1")}"

host = azurerm\_public\_ip.publicip01.ip\_address

}

}

}

# variable.tf

variable "vnet\_name" {

type = string

default = "vnet\_test2"

}

variable "rg\_name" {

type = string

default = "rg\_test4"

}

variable "subnet\_ip" {

type = list

default = ["10.0.0.0/24", "10.0.1.0/24", "10.0.2.0/24"]

}

variable "subnet\_name" {

type = list

default = ["subnet\_test\_1", "subnet\_test\_2", "subnet\_test\_3"]

}

variable "boot\_diagnostics\_sa\_type" {

default = "Standard\_LRS"

}

# text1.txt

This is text file.

This will be copied to the server